

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第5733318号
(P5733318)

(45) 発行日 平成27年6月10日(2015.6.10)

(24) 登録日 平成27年4月24日(2015.4.24)

(51) Int.Cl. F I
G06F 11/20 (2006.01) G O 6 F 11/20 3 1 0 A
G06F 9/46 (2006.01) G O 6 F 9/46 3 5 0

請求項の数 12 (全 29 頁)

<p>(21) 出願番号 特願2012-542705 (P2012-542705)</p> <p>(86) (22) 出願日 平成22年11月12日(2010.11.12)</p> <p>(86) 国際出願番号 PCT/JP2010/006654</p> <p>(87) 国際公開番号 W02012/063294</p> <p>(87) 国際公開日 平成24年5月18日(2012.5.18)</p> <p>審査請求日 平成25年3月15日(2013.3.15)</p> <p>前置審査</p>	<p>(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</p> <p>(74) 代理人 100100310 弁理士 井上 学</p> <p>(72) 発明者 金 成昊 神奈川県横浜市戸塚区吉田町292番地 株式会社日立製作所システム開発研究所内</p> <p>(72) 発明者 西島 英児 神奈川県横浜市戸塚区吉田町292番地 株式会社日立製作所システム開発研究所内</p> <p>審査官 ▲高▼橋 正▲徳▼</p> <p style="text-align: right;">最終頁に続く</p>
---	--

(54) 【発明の名称】 計算機システム

(57) 【特許請求の範囲】

【請求項1】

複数の計算機ノードを有する計算機システムであって、
 前記複数の計算機ノードは各々、複数の仮想計算機と、当該複数の仮想計算機を制御する制御基盤部とを有しており、

各仮想計算機は、当該仮想計算機が動作している自計算機ノードとは別の他の計算機ノード上で動作している他の仮想計算機と多重化グループを構成し、当該仮想計算機若しくは当該他の仮想計算機のいずれか一方が主系仮想計算機として動作し他方が従系仮想計算機として動作しており、

前記制御基盤部は、当該制御基盤部が備えられている自計算機ノードの複数の仮想計算機各々について、主系仮想計算機若しくは従系仮想計算機のいずれとして動作しているのかを管理し、当該複数の仮想計算機各々の状態を監視しており、

前記制御基盤部は、自計算機ノード上で主系仮想計算機として動作する仮想計算機の障害を検知した場合に、前記自計算機ノード上で主系仮想計算機として動作している仮想計算機の数と従系仮想計算機として動作している仮想計算機の数との比率に応じて、障害が生じた仮想計算機とともに自計算機ノード上で動作する他の仮想計算機も主系仮想計算機から従系仮想計算機へ切り替えるかどうか判定することを特徴とする計算機システム。

【請求項2】

複数の計算機ノードを有する計算機システムであって、
 前記複数の計算機ノードは各々、複数の仮想計算機と、当該複数の仮想計算機を制御す

10

20

る制御基盤部とを有しており、

各仮想計算機は、当該仮想計算機が動作している自計算機ノードとは別の他の計算機ノード上で動作している他の仮想計算機と多重化グループを構成し、当該仮想計算機若しくは当該他の仮想計算機のいずれか一方が主系仮想計算機として動作し他方が従系仮想計算機として動作しており、

前記制御基盤部は、当該制御基盤部が備えられている自計算機ノードの複数の仮想計算機各々について、主系仮想計算機若しくは従系仮想計算機のいずれとして動作しているのかを管理し、当該複数の仮想計算機各々の状態を監視しており、

前記制御基盤部は、自計算機ノード上で主系仮想計算機として動作する仮想計算機の障害を検知した場合に、前記自計算機ノード上で主系仮想計算機として動作している仮想計算機の数、当該自計算機ノード上で動作している仮想計算機の総数の半数未満である場合に、障害が生じた仮想計算機とともに当該自計算機ノード上で主系仮想計算機として動作している他の仮想計算機も、主系仮想計算機から従系仮想計算機へ切り替えるよう制御を行うことを特徴とする計算機システム。

10

【請求項3】

複数の計算機ノードを有する計算機システムであって、

前記複数の計算機ノードは各々、複数の仮想計算機と、当該複数の仮想計算機を制御する制御基盤部とを有しており、

各仮想計算機は、当該仮想計算機が動作している自計算機ノードとは別の他の計算機ノード上で動作している他の仮想計算機と多重化グループを構成し、当該仮想計算機若しくは当該他の仮想計算機のいずれか一方が主系仮想計算機として動作し他方が従系仮想計算機として動作しており、

20

前記制御基盤部は、当該制御基盤部が備えられている自計算機ノードの複数の仮想計算機各々について、主系仮想計算機若しくは従系仮想計算機のいずれとして動作しているのかを管理し、当該複数の仮想計算機各々の状態を監視しており、

前記制御基盤部は、自計算機ノード上で主系仮想計算機として動作する仮想計算機の障害を検知した場合に、前記自計算機ノード上で主系仮想計算機として動作している仮想計算機の数、当該自計算機ノード上で動作している仮想計算機の総数の半数以上である場合には、障害が生じた仮想計算機を主系仮想計算機から従系仮想計算機へ切り替えるよう制御し、

30

前記制御基盤部は、前記障害が回復した場合に、前記仮想計算機を従系仮想計算機から主系仮想計算機へ戻すよう制御することを特徴とする計算機システム。

【請求項4】

請求項1から請求項3の何れか一つの請求項に記載の計算機システムであって、

前記制御基盤部は、自計算機ノード上で動作する仮想計算機を主系仮想計算機から従系仮想計算機へ切り替える場合に、当該仮想計算機へ従系仮想計算機への切替要求を通知すると共に、当該仮想計算機と多重化グループを構成している他の仮想計算機を従系仮想計算機から主系仮想計算機へ切り替えるために、当該他の仮想計算機が動作する他の計算機ノードへ通知を送信することを特徴とする計算機システム。

【請求項5】

請求項1から請求項3の何れか一つの請求項に記載の計算機システムであって、

前記制御基盤部は、前記自計算機ノード上で動作する各仮想計算機が使用するハードウェアリソースを管理しており、主系仮想計算機として動作する複数の仮想計算機に共有されるハードウェアリソースの障害を検知した場合には、当該複数の仮想計算機を共に主系仮想計算機から従系仮想計算機へ切り替えるよう制御することを特徴とする計算機システム。

40

【請求項6】

請求項1から請求項3の何れか一つの請求項に記載の計算機システムであって、

前記制御基盤部は、前記自計算機ノード上で動作する複数の仮想計算機が論理的なグループを構成している場合に、当該複数の仮想計算機の一に障害が生じた場合は、当該論理

50

的なグループを構成している他の仮想計算機も共に主系仮想計算機から従系仮想計算機へ切り替えるよう制御することを特徴とする計算機システム。

【請求項7】

複数の計算機ノードを有する計算機システムにおいて実行される方法であって、

前記複数の計算機ノードは各々、複数の仮想計算機と、当該複数の仮想計算機を制御する制御基盤部とを有しており、

各仮想計算機は、当該仮想計算機が動作している自計算機ノードとは別の他の計算機ノード上で動作している他の仮想計算機と多重化グループを構成し、当該仮想計算機若しくは当該他の仮想計算機のいずれか一方が主系仮想計算機として動作し他方が従系仮想計算機として動作しており、

前記制御基盤部は、当該制御基盤部が備えられている自計算機ノードの複数の仮想計算機各々について、主系仮想計算機若しくは従系仮想計算機のいずれとして動作しているのかを管理し、当該複数の仮想計算機各々の状態を監視しており、

前記制御基盤部は、自計算機ノード上で主系仮想計算機として動作する仮想計算機の障害を検知した場合に、前記自計算機ノード上で主系仮想計算機として動作している仮想計算機の数と従系仮想計算機として動作している仮想計算機の数との比率に応じて、障害が生じた仮想計算機とともに自計算機ノード上で動作する他の仮想計算機も主系仮想計算機から従系仮想計算機へ切り替えるかどうか判定することを特徴とする方法。

10

【請求項8】

複数の計算機ノードを有する計算機システムにおいて実行される方法であって、

前記複数の計算機ノードは各々、複数の仮想計算機と、当該複数の仮想計算機を制御する制御基盤部とを有しており、

各仮想計算機は、当該仮想計算機が動作している自計算機ノードとは別の他の計算機ノード上で動作している他の仮想計算機と多重化グループを構成し、当該仮想計算機若しくは当該他の仮想計算機のいずれか一方が主系仮想計算機として動作し他方が従系仮想計算機として動作しており、

前記制御基盤部は、当該制御基盤部が備えられている自計算機ノードの複数の仮想計算機各々について、主系仮想計算機若しくは従系仮想計算機のいずれとして動作しているのかを管理し、当該複数の仮想計算機各々の状態を監視しており、

前記制御基盤部は、自計算機ノード上で主系仮想計算機として動作する仮想計算機の障害を検知した場合に、前記自計算機ノード上で主系仮想計算機として動作している仮想計算機の数が、当該自計算機ノード上で動作している仮想計算機の総数の半数未満である場合に、障害が生じた仮想計算機とともに当該自計算機ノード上で主系仮想計算機として動作している他の仮想計算機も、主系仮想計算機から従系仮想計算機へ切り替えるよう制御を行うことを特徴とする計算機方法。

20

30

【請求項9】

複数の計算機ノードを有する計算機システムにおいて実行される方法であって、

前記複数の計算機ノードは各々、複数の仮想計算機と、当該複数の仮想計算機を制御する制御基盤部とを有しており、

各仮想計算機は、当該仮想計算機が動作している自計算機ノードとは別の他の計算機ノード上で動作している他の仮想計算機と多重化グループを構成し、当該仮想計算機若しくは当該他の仮想計算機のいずれか一方が主系仮想計算機として動作し他方が従系仮想計算機として動作しており、

前記制御基盤部は、当該制御基盤部が備えられている自計算機ノードの複数の仮想計算機各々について、主系仮想計算機若しくは従系仮想計算機のいずれとして動作しているのかを管理し、当該複数の仮想計算機各々の状態を監視しており、

前記制御基盤部は、自計算機ノード上で主系仮想計算機として動作する仮想計算機の障害を検知した場合に、前記自計算機ノード上で主系仮想計算機として動作している仮想計算機の数が、当該自計算機ノード上で動作している仮想計算機の総数の半数以上である場合には、障害が生じた仮想計算機を主系仮想計算機から従系仮想計算機へ切り替えるよう

40

50

制御し、

前記制御基盤部は、前記障害が回復した場合に、前記仮想計算機を従系仮想計算機から主系仮想計算機へ戻すよう制御することを特徴とする方法。

【請求項 10】

請求項 7 から請求項 9 の何れか一つの請求項に記載の方法であって、

前記制御基盤部は、自計算機ノード上で動作する仮想計算機を主系仮想計算機から従系仮想計算機へ切り替える場合に、当該仮想計算機へ従系仮想計算機への切替要求を通知すると共に、当該仮想計算機と多重化グループを構成している他の仮想計算機を従系仮想計算機から主系仮想計算機へ切り替えるために、当該他の仮想計算機が動作する他の計算機ノードへ通知を送信することを特徴とする方法。

10

【請求項 11】

請求項 7 から請求項 9 の何れか一つの請求項に記載の方法であって、

前記制御基盤部は、前記自計算機ノード上で動作する各仮想計算機が使用するハードウェアリソースを管理しており、主系仮想計算機として動作する複数の仮想計算機に共有されるハードウェアリソースの障害を検知した場合には、当該複数の仮想計算機を共に主系仮想計算機から従系仮想計算機へ切り替えるよう制御することを特徴とする方法。

【請求項 12】

請求項 7 から請求項 9 の何れか一つの請求項に記載の方法であって、

前記制御基盤部は、前記自計算機ノード上で動作する複数の仮想計算機が論理的なグループを構成している場合に、当該複数の仮想計算機の一に障害が生じた場合は、当該論理的なグループを構成している他の仮想計算機も共に主系仮想計算機から従系仮想計算機へ切り替えるよう制御することを特徴とする方法。

20

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、サーバ仮想化環境において可用性を上げるために複数の計算機ノードを多重化して構成された計算機システムに関し、特に計算機ノード上のホストマシンおよび各仮想マシンの障害監視と計算機ノードの系切り替えを行う方法に関する。

【背景技術】

【0002】

サーバ仮想化技術は、1台の物理計算機上で複数の仮想マシンを実行し、物理計算機の台数を削減することで、運用コスト、計算機の設置スペース及び消費電力等を削減できるメリットがあり、近年、注目を浴びている。ただし、物理計算機で運用していたシステムと同様な信頼性の確保のため、仮想化環境を提供する物理計算機の多重化とともに、各物理計算機で制御・運用される仮想マシンの多重系構成も要求されている。

30

【0003】

特許文献 1 には、物理計算機上の各仮想マシン単位で、障害を検知し系を切替える方法が開示されている。特許文献 1 では、図 20 に示すように、物理計算機 2001 と 2031 が、LAN を介して多重化されており、物理計算機 A 2001 はホストマシン 2002 および仮想マシン 2005 と 2015 を備えたサーバ仮想化環境を備えており、物理計算機 B 2031 も同様な構成を備えている。また、障害検知方法および系切替方法として、物理計算機 A 2001 上の仮想マシン A 2005 上にゲストクラスタプログラム 2007、仮想マシン B 2015 上にゲストクラスタプログラム 2017、および、ホストマシン 2002 上にホストクラスタプログラム 2003 を設けている。物理計算機 B 2031 も同様な構成である。

40

【0004】

ゲストクラスタプログラム 2007 と 2017 または 2037 と 2047 は、アプリケーションプログラム 2006、2016、2036、2046 の稼働状況を監視しつつ、ホストクラスタプログラム 2003 または 2033 に監視結果を通知する。また、ゲストクラスタプログラム 2007、2017 は、ホストクラスタプログラム 2003 から系切

50

替え命令を受けた場合には、各々対応する仮想マシン2005、2015に対する主系/従系モードの系切替えを実施する。仮想マシンが主系モードの時には、アプリケーションが入力されたデータに対して処理を行い処理結果を出力する。一方、仮想マシンが従系モードの時には、アプリケーションが入力されたデータに対して処理を行うが処理結果の出力は行わない。また、ゲストクラスタプログラム2037、2047も同様に、ホストクラスタプログラム2033から系切替え命令を受けた場合には、各々対応する仮想マシン2035、2045に対する主系/従系モードの系切替えを実施する。一方、ホストクラスタプログラム2003または2033は、同じ物理計算機上で稼働している各仮想マシン(仮想マシン2005と2015または仮想マシン2035と2045)の稼働状況を監視し、例えば、主系モードの仮想マシンA2005が停止していることを検知した場合には、仮想マシンA2035を従系モードから主系モードへ切り替える。

10

【先行技術文献】

【特許文献】

【0005】

【特許文献1】特開2008-269332号公報

【発明の概要】

【発明が解決しようとする課題】

【0006】

特許文献1では、物理計算機上の仮想マシン単位での障害を検知し障害発生時には主系と従系との間の系切り替えを実施する。例えば、図20において、物理計算機A2001上の各仮想マシン2005と2015が全て主系モードで、物理計算機B2031上の各仮想マシン2035と2045が全て従系モードの場合に、物理計算機A2001上の仮想マシンA2005が障害とする。この場合には、物理計算機B上の仮想マシンA2035が従系モードから主系モードに切り替わって稼働する。この場合物理計算機Aには主系モードの仮想マシンと従系モードの仮想マシンが混在した状態となる。物理計算機Bも同様に主系モードおよび従系モードの仮想マシンが混在した状態となる。

20

【0007】

しかしながら、物理計算機に対してハードウェア保守を行いたい場合には、物理計算機上の全ての仮想マシンが従系モード(または停止モード)でなければ、物理計算機を一旦停止させることができない。なぜなら、主系モードの仮想マシンは実際の処理を行った結果を出力しているから、処理中の仮想マシンを停止することができない。つまり、物理計算機上に主系モードの仮想マシンおよび従系モードの仮想マシンが混在している状態では、ハードウェア保守を行うことができない問題がある。

30

【0008】

そこで、仮想化計算機の多重系構成制御において、物理計算機上に搭載されている複数の仮想マシンがなるべく同じの動作モードとなるように、各仮想マシンの系切替えを実施する技術を提供する。

【課題を解決するための手段】

【0009】

計算機システムは複数の計算機ノードを有し、各計算機ノードは複数の仮想計算機と仮想計算機を制御する制御基盤部とを有する。各仮想計算機は、自計算機ノードとは別の他の計算機ノード上で動作している他の仮想計算機と多重化グループを構成し、いずれか一方が主系仮想計算機として動作し他方が従系仮想計算機として動作している。制御基盤部は、各仮想計算機について、主系/従系仮想計算機のいずれとして動作しているのかを管理し、各仮想計算機各々の状態を監視している。制御基盤部は、自計算機ノード上で主系仮想計算機として動作する仮想計算機の障害を検知した場合に、予め定められた規則に従って、障害が生じた仮想計算機とともに自計算機ノード上で動作する他の仮想計算機も主系仮想計算機から従系仮想計算機へ切り替えるかどうか判定する。

40

【発明の効果】

【0010】

50

仮想化計算機の多重系構成制御において、物理計算機上に搭載されている複数の仮想マシンがなるべく同じの動作モードとなるように、各仮想マシンの系切替えを実施することができる。

【図面の簡単な説明】

【0011】

【図1】仮想化多重システムの構成例を示す図である。

【図2】計算機ノードの構成例を示す図である。

【図3】計算機ノード構成管理テーブルの構成例を示す図である。

【図4】VM状態管理テーブルの構成例を示す図である。

【図5】リソース管理テーブルの構成例を示す図である。

【図6】系切替制御テーブルの構成例を示す図である。

【図7】VM1を対象とした系切替制御テーブルの例を示す図である。

【図8】計算機ノード間通信フォーマットの一例を示す図である。

【図9】多重系制御エージェントプログラムと多重系制御基盤プログラム間の通信フォーマットの一例を示す図である。

【図10】仮想化多重系システムで実行される処理の一例を示すフローチャートである。

【図11】リソース監視処理の一例を示す図である。

【図12】リソース監視処理の一例を示すフローチャートである。

【図13】系切替通知処理の一例を示すフローチャートである。

【図14】メッセージ送信処理の一例を示すフローチャートである。

【図15】ノード監視処理の一例を示すフローチャートである。

【図16】生死監視処理の一例を示すフローチャートである。

【図17】全VM切替処理の一例を示すフローチャートである。

【図18】主系処理の一例を示すフローチャートである。

【図19】従系処理の一例を示すフローチャートである。

【図20】計算機システムの一例を示す図である。

【発明を実施するための形態】

【0012】

以下、実施の形態を図面を用いて説明する。なお、本発明が実施の形態に制限されることは無く、本発明の思想に合致するあらゆる応用例が本発明に該当する。

【実施例1】

【0013】

<図1：全体構成例>

図1は、仮想化多重システムの全体構成例である。

【0014】

仮想化多重系システム1は一つまたは複数の端末2と外部ネットワーク3を介して接続される。仮想化多重系システム1は二台の計算機ノード100（図1では計算機ノードA 100-Aと計算機ノードB 100-B）と内部ネットワーク4より構成されている。

【0015】

計算機ノード100はメモリ101、プロセッサ102、記憶装置103、外部通信インタフェース104（図中、インタフェースをI/Fと略記）、ノード間通信I/F105を備える。計算機ノードA 100-Aと計算機ノードB 100-Bは共に同様の構成を有する。

【0016】

メモリ101内にはホストオペレーティングシステム400（図中、ホストOSと略記）、多重系制御基盤プログラム300と一つまたは複数の仮想マシン環境200（図中、VMと略記）が保持される。ホストOS400内には仮想化プログラム401とデバイス制御プログラム402が保持される。多重系制御基盤プログラム300は計算機ノード構成管理テーブル301、VM状態管理テーブル302、リソース管理テーブル303、系

10

20

30

40

50

切替制御テーブル304と計算機ノード監視部305から構成される。VM200はアプリケーション211(図中、APと略記)、多重系制御エージェントプログラム212、ゲストオペレーティングシステム213(図中、ゲストOSと略記)から構成される。

【0017】

尚、図1では端末2として三台の端末2-A、2-B、2-Cが仮想化多重系システム1に接続されており、また計算機ノードA 100-A内にはVM200としてVM1200-A、VM2 200-B、VM3 200-Cが備えられている例を示したが、特に台数は限定されるものではない。

【0018】

仮想化多重系システム1内の計算機ノードA 100-A、計算機ノードB 100-Bは内部ネットワーク4を介して接続される。計算機ノード100では、外部ネットワーク3は外部通信I/F104に接続し、内部ネットワーク4はノード間通信I/F105に接続する。

【0019】

端末2は外部ネットワーク3を介して仮想化多重系システム1に要求を送信し、仮想化多重系システム1は要求を処理した結果を端末2に返信する。仮想化多重系システム1内の計算機ノードA 100-A、計算機ノードB 100-Bは、一方が「主系ノード」、他方が「従系ノード」として動作する。端末2からの要求は計算機ノードA 100-A、計算機ノードB 100-B両方で受取られ、処理されるが、正常な状態で動作する場合には、処理結果は「主系ノード」である計算機ノード100だけが端末2に返信する。外部通信I/F104などの故障で主系ノードの処理継続に支障が生じた時に、計算機ノード100間の主系ノード/従系ノードの関係は切替る。

【0020】

計算機ノード100の記憶装置103はホストOS400、多重系制御基盤プログラム300(プログラムと各種テーブル情報)、VM200を構成するAP211、多重系制御エージェントプログラム212、ゲストOS213を格納する。プロセッサ102は記憶装置103からホストOS400、多重系制御基盤プログラム300、VM200構成物をメモリ101に展開、実行し、また外部通信I/F104やノード間通信I/F105からの割込みを処理する。外部通信I/F104は外部ネットワーク3を通じて端末2とのデータの送受信を行う。ノード間通信I/F105は内部ネットワーク4を通じて他の計算機ノード100との間でデータの送受信を行う。

【0021】

仮想化プログラム401はVM200を作成、管理するためのプログラムであり、デバイス制御プログラム402は外部通信I/F104、ノード間通信I/F105や記憶装置103にアクセスするためのプログラムである。ホストOS400は仮想化プログラム401やデバイス制御プログラム402を用いて、多重系制御基盤プログラム300やVM200の実行を制御し、外部通信I/F104、ノード間通信I/F105や記憶装置103へのアクセス制御を行う。

【0022】

VM200は固有のプログラムやオペレーティングシステムが動作することが可能な、仮想化プログラム401により作成された仮想的な実行環境である。複数のVM200が存在する場合に、異なるVM200内で動作するプログラムやオペレーティングシステムは同じでも良いし、異なっても良い。また、異なるVM200内で動作するプログラムやオペレーティングシステムが互いに直接影響を与え合うことはない。VM200は仮想化多重系システム1内で一意の識別子を持ち、例えば、図1中のVM200は「VM1」、「VM2」、「VM3」の識別子を保有する。また異なる計算機ノード100内で動作する任意のVM200の対(ペア)は、「主系VM」/「従系VM」の関係を持つ。通常、主系ノードの計算機ノード100内のVM200は「主系VM」となり、従系ノードの計算機ノード100内のVM200は「従系VM」となる。「主系VM」/「従系VM」の関係を持つVM200の対は、二つの単体VM200間で構成される場合もあるし、

10

20

30

40

50

一つの計算機ノード100内の複数のVMグループが協同してアプリケーションを実行してサービスを提供する場合には複数のVM200から構成されるVMグループ間で構成される場合もある。主系VMの動作に異常が生じた場合には、「主系VM」/「従系VM」の関係を入れ替える、系切替処理が行われる。

【0023】

AP211は端末2からの要求を処理する。AP211は自身が動作するVM200が主系VM、従系VMのいずれであっても端末2からの要求を読み取り、演算処理を行うが、端末2への応答動作は異なる。主系VMで動作するAP211は演算結果を端末2へ返答するが、従系VMで動作するAP211は演算結果を返答しない。AP211は「系切替」を行うことで端末2への返答動作を切替える。

10

【0024】

多重系制御エージェントプログラム212は多重系制御基盤プログラム300との間で情報通信を行うプログラムであり、AP211の動作状態を監視して多重系制御基盤プログラム300に通知したり、多重系制御基盤プログラム300からの指示に従ってAP211の系切替を行ったりする。AP211の動作状態を監視する方法の詳細は図11、図12と共に後述する。

【0025】

ゲストOS213はVM200の環境内でAP211や多重系制御エージェントプログラム212の動作を制御する。

【0026】

多重系制御基盤プログラム300は、他の計算機ノード100内で動作する多重系制御基盤プログラム300との間で情報を交換することで仮想化多重系システム1内のVM200の状態を判断し、VM200の「主系VM」/「従系VM」の系切替を制御する。主系ノードである計算機ノード100内で動作する多重系制御基盤プログラム300はVM200の状態判断と系切替の実行を指示し、従系ノードの計算機ノード100内で動作する多重系制御基盤プログラム300は主系ノードの生死状態を監視する。計算機ノード100の監視は、主系ノード、従系ノードの双方から行う方式も有り得るが、本実施形態では従系ノードからのみ実行する。しかし、本発明の思想がこれに限定されるものではない。多重系制御基盤プログラム300のVM200の状態判断の詳細は図13と共に後述する。系切替や生死状態監視処理についての詳細は、計算機ノード監視部305の処理説明として、図14、図15、図16、図17、図18、図19と共に後述する。

20

30

【0027】

計算機ノード構成管理テーブル301は多重系制御基盤プログラム300が計算機ノード100の主系ノード/従系ノードなどの状態を管理するためのテーブルである。詳細は図3と共に後述する。

【0028】

VM状態管理テーブル302は多重系制御基盤プログラム300が計算機ノード100内のVM200の主系VM/従系VMなどの状態を管理するためのテーブルである。詳細は図4と共に後述する。

【0029】

リソース管理テーブル303は多重系制御基盤プログラム300が、VM200が使用しているメモリ101や実行しているAP211、他のVM200との関係などの「リソース」を管理するためのテーブルである。詳細は図5と共に後述する。

40

【0030】

系切替制御テーブル304は多重系制御基盤プログラム300が、VM200が使用するリソースに異常が発生した時などにおける、障害内容とこれに対する対応動作を記載したテーブルである。詳細は図6、図7と共に後述する。

【0031】

計算機ノード監視部305は多重系制御基盤プログラム300が他の計算機ノード100内で動作する多重系制御基盤プログラム300との間で通信を行い、生死状態監視処理

50

を行う処理である。計算機ノード監視部305は多重系制御基盤プログラム300を構成するプログラムではあるが、多重系制御基盤プログラム300の処理とは独立した契機で活動する。詳細は図14、図15、図16、図17、図18、図19と共に後述する。

<図2：仮想化環境構成例>

図2は計算機ノードの構成の例を示す図である。アプリケーションやオペレーティングシステムの動作環境であるVM200は仮想的な計算機の実行環境であり、実際には、計算機ノード100内のメモリ101やプロセッサ102などの現存するリソースを割当てることで構築される。

【0032】

図2は計算機ノード100のメモリ101、プロセッサ102、外部通信I/F104等のリソースを複数のVM200やホストOS400への割当てた例を示した図である。

【0033】

図2中において、計算機ノードA 100 - A内には複数のVM200 (VM1 200 - A、VM2 200 - B、VM3 200 - C)が作成されている。メモリ101は複数のメモリ領域に分割することが可能で、それぞれのメモリ領域が個別に各VM200やホストOS400に割り当てられたり、共有されたりする。プロセッサ102は「マルチコア」のプロセッサで、コア単位でVM200やホストOS400に割り当てられたり、共有されたりする。外部通信I/F104は計算機ノード100内には複数存在し、それぞれの外部通信I/F104がVM200やホストOS400に割り当てられたり、共有されたりする。

【0034】

仮想環境VM1 200 - Aは、プロセッサ102のVM1占有コア#1 102 - A、VM1占有メモリ101 - AとVM1とVM2間共有メモリ101 - Bのメモリ領域、VM1とVM2共有外部通信I/F104 - Aのリソースを使用して構築される。仮想環境VM1 200 - Aは、プロセッサ102のVM2占有コア#2 102 - B、VM2占有メモリ101 - CとVM1とVM2共有メモリ101 - Bのメモリ領域、VM1とVM2共有外部通信I/F104 - Aのリソースを使用して構築される。

VM1 200 - A、VM2 200 - Bに割当てられていないリソースであるプロセッサ102のホストOS占有コア#0 102 - Cは、ホストOS400のみが使用するプロセッサ・コアで、VM200から使用されることは無い。また外部通信I/F104 - BはホストOS400と、VM1 200 - AとVM2 200 - B以外のVM200が使用する。

【0035】

計算機ノードB 100 - B内にも複数のVM201 (VM4 201 - A、VM5 201 - B、VM6 201 - C)が作成されており、各VM201やホストOS400には計算機ノードB 100 - B内のリソースが割り当てられる。

<図3：計算機ノード構成管理テーブル>

図3は計算機ノード構成管理テーブル301の構成例を示す図である。計算機ノード構成管理テーブル301は多重系制御基盤プログラム300内に保持され、仮想化多重系システム1内の計算機ノード100の状態情報を格納する。仮想化多重系システム1内の各計算機ノードA 100 - A、計算機ノードB 100 - Bは、同じ内容の計算機ノード構成管理テーブル301を保持する。

【0036】

計算機ノード識別子311には仮想化多重系システム1内の計算機ノード100を識別する一意の情報を格納する。

【0037】

主/従系状態フラグ312には各計算機ノード100が「主系ノード」、「従系ノード」のいずれの状態にあるかを識別する情報を格納する。対象の計算機ノード100が主系ノードの場合は「M」を格納し、従系ノードの場合は「S」を格納する。

【0038】

10

20

30

40

50

マルチキャストアドレス 313 には計算機ノード 100 にアクセスするための、内部ネットワーク 4 のマルチキャストアドレスを格納する。

【0039】

生死状態フラグ 314 には計算機ノード 100 の正常/異常状態を識別する情報を格納する。計算機ノード 100 が「正常状態」の場合には「0」値を格納し、「異常状態」の場合には「1」を格納する。本発明の実施の形態では正常/異常状態を「0/1」値で表すが、正常/異常状態の識別が可能な情報であれば格納情報は「0/1」値に限らない。

【0040】

監視周期 315 は計算機ノード 100 の状態を監視する周期時間を格納する。従系ノードの多重系制御基盤プログラム 300 の計算機ノード監視部 305 は、監視周期 315 の時間周期で、主系ノードの計算機ノード 100 の状態を取得する。主系ノードは従系ノードの計算機ノード 100 の状態取得要求に応じて、状態情報を返信する。

【0041】

生死確定カウンタ 316 は計算機ノード 100 の状態取得要求に対して応答が無かった場合に、相手計算機ノード 100 が異常状態にあることを判断するためのしきい値を格納する。生死確定カウンタ 316 の回数分、状態取得要求に対する相手計算機ノード 100 からの返信が連続して無かった場合は、相手計算機ノード 100 が異常状態にあると判断する。

<図 4：VM 状態管理テーブル>

図 4 は本発明の実施の形態の VM 状態管理テーブル 302 の構成を示す図である。VM 状態管理テーブル 302 は多重系制御基盤プログラム 300 内に保持され、計算機ノード 100 内に作成されている VM 200 の状態情報を保持する。

【0042】

VM 状態管理テーブル 302 に格納される状態情報は計算機ノード 100 毎に異なる。計算機ノード A 100 - A の VM 状態管理テーブル 302 は VM 1 200 - A、VM 2 200 - B、VM 3 200 - C の状態情報を保持する。計算機ノード B 100 - B の VM 状態管理テーブル 302 は VM 4 201 - A、VM 5 201 - B、VM 6 201 - C の状態情報を保持する。

管理 VM 総数 321 には一台の計算機ノード 100 内に作成されている VM 200 の総数を格納する。

【0043】

障害 VM 総数 322 には一台の計算機ノード 100 内に作成されている VM 200 内、異常状態にある VM 200 の個数を格納する。障害 VM 総数 322 は「0」以上、管理 VM 総数 321 の値以下の値となる。

【0044】

VM 識別子 323 には仮想化多重系システム 1 内で VM 200 を一意に識別する情報を格納する。行 328 は VM 1 200 - A の状態情報を格納した行であることを示す。

【0045】

ホスト計算機ノード識別子 324 は VM 200 が作成されている計算機ノード 100 の識別情報を格納する。

【0046】

主/従系状態フラグ 325 には VM 200 の「主系 VM」、「従系 VM」を識別する情報を格納する。対象の VM 200 が主系 VM の場合は「M」を格納し、従系 VM の場合は「S」を格納する。

【0047】

生死状態フラグ 326 には VM 200 の正常/異常状態を識別する情報を格納する。VM 200 が「正常状態」の場合には「0」値を格納し、「異常状態」の場合には「1」を格納する。本発明の実施の形態では正常/異常状態を「0/1」値で表すが、正常/異常状態の識別が可能な情報であれば格納情報は「0/1」値に限らない。

【0048】

10

20

30

40

50

主/従系構成VM情報327には、VM識別子323のVM200と主系VM/従系VMの関係に対応するVM200の情報を格納する。ホスト計算機ノード識別子327-Aには対応するVM200が作成されている計算機ノード100の識別情報を格納し、VM識別子327-Bには対応するVM200を一意に特定する識別情報を格納する。

<図5：リソース管理テーブル>

図5はリソース管理テーブル303の構成例を示す図である。リソース管理テーブル303は多重系制御基盤プログラム300内に保持され、VM200が使用する個別リソースの状態情報を格納する。リソース管理テーブル303に格納される状態情報は計算機ノード100毎に異なる。計算機ノードA 100-Aのリソース管理テーブル303はVM1 200-A、VM2 200-B、VM3 200-Cが使用するリソースの状態情報を保持する。計算機ノードB 100-Bのリソース管理テーブル303はVM4 201-A、VM5 201-B、VM6 201-Cが使用するリソースの状態情報を保持する。

10

【0049】

VM識別子331にはVM200を一意に識別する情報を格納し、リソース識別子332にはVM200が使用するリソースを一意に特定する情報を格納する。行336はVM1 200-AがAP1 211-Aをリソースとして使用していることを意味する。

【0050】

監視周期333には使用リソースの正常/異常状態を監視するための周期時間を格納する。リソース状態の取得をイベントなどの契機で行い、周期的に監視しない場合は無効値である「0」を格納する。

20

【0051】

リソース生死状態フラグ334には使用リソースの正常/異常状態を識別する情報を格納する。リソースの状態が「正常状態」の場合には「0」値を格納し、「異常状態」の場合には「1」を格納する。本発明の実施の形態では正常/異常状態を「0/1」値で表すが、正常/異常状態の識別が可能な情報であれば格納情報は「0/1」値に限らない。

【0052】

リソース詳細情報335には所有状態335-Aと個別情報335-Bの項目が含まれる。所有状態335-AにはVM200のリソースを使用する形態を格納し、個別情報335-Bにはリソース状態の取得手段を示す情報を格納する。

30

【0053】

例えば図5における所有状態335-Aの値、「占有」は使用リソースをVM1 200-Aのみで使用することを意味し(行336)、「共有」は使用リソースをVM1 200-Aと他VM200とで使用することを意味し(行338)、「論理/SERVER」はVM1 200-AがVM2 200-Bと論理的な繋がりのあるグループを形成して「サーバ」の関係になっていることを意味し(行339)、「論理/CLIENT」はVM2 200-BがVM1 200-Aと論理的な繋がりのあるグループを形成して「クライアント」の関係になっていることを意味する(行340)。

【0054】

例えば図5における個別情報335-Bの値、「212」は使用リソースAP1 211-Aの状態を多重系制御エージェントプログラム212が取得することを意味し(行336)、「EVT」は使用リソースVM1占有メモリ101-A状態の取得はハードウェア状態を検知する計算機ノード100のハードウェア機能などが送出するシグナル信号を検出することで行う事を意味する(行337)。図5の例では、多重系制御エージェントプログラム212は監視周期333に格納された値の周期で使用リソースAP1 211-Aの状態を取得するが(行336)、「EVT」の場合はハードウェア機能は使用リソースVM1占有メモリ101-Aの状態に変化が生じた時にだけシグナル信号を送出するので監視周期333の格納値は使用しない(行337)。

40

<図6：系切替制御テーブル>

図6は系切替制御テーブル304の構成例を示す図である。系切替制御テーブル304

50

は多重系制御基盤プログラム300内に保持され、障害発生時に多重系制御基盤プログラム300がVM200の主系VM/従系VMの系切替を行うための規則を格納する。仮想化多重系システム1内の各計算機ノード100は同じ内容の系切替制御テーブル304を保持している。

【0055】

障害内容341には計算機ノード100やVM200の障害発生部位を指定する情報を格納する。図6における「計算機ノード障害」は計算機ノード100自体の動作継続に支障が生じる部位の指定情報であることを示す。VM200の障害発生部位は、全てのVM200を対象に、使用リソースの所有状態種別毎にリソース識別子を格納する。

【0056】

系切替対象VM342には、障害内容341に障害発生時に系切替対象となるVM200のVM識別子を格納する。図6における「全VM」は障害が生じた計算機ノード100内に存在する全てのVM200を系切替対象とすることを意味する。「該当VMのみ」は障害が発生したリソースを使用するVM200だけが系切替対象となることを意味し、実際には当該VM200の識別子が格納される。「関連全VM」は障害が発生したリソースを共用する複数のVM200、または障害が発生したリソースを使用しているVM200及び当該VM200と論理的なグループ関係にある複数のVM200と一緒に系切替対象となることを意味し、実際には系切替対象となる複数のVM200の識別子が格納される。

【0057】

過半数判断ロジック適用可否343にはVM200の系切替実行時に「過半数判断ロジック」の適用の有無を指定する情報を格納する。「要」は「過半数判断ロジック」を適用することを意味し、「否」は「過半数判断ロジック」を適用しないことを意味する。「過半数判断ロジック」は主系VM/従系VMの系切替対象となるVM200を再評価する処理であり、VM状態管理テーブル302の障害VM総数322と系切替対象VM342のVM200の合計数が、VM状態管理テーブル302の管理VM総数321の半数以上であるかを判断し、系切替対象のVM200を決定する。過半数であれば「過半数判断ロジック」は全VMを系切替対象と決定し、半数未満であれば系切替対象VM342に格納されたVM200のみを系切替対象とする。

【0058】

Failback処理適用可否344には系切替を行ったVM200に「フェイルバック」処理の適用の可否を指定する情報を格納する。「可」は系切替を行ったVM200をフェイルバックしても良いことを意味し、「否」は系切替を行ったVM200をフェイルバックしてはいけないことを意味する。「フェイルバック」とはVM200の主系VM/従系VMの状態を系切替実行前に戻す処理のことである。主系VMのVM200に障害が発生すると多重系制御基盤プログラム300により系切替が実行され、VM200は従系VMに切替る。「フェイルバック」処理は、系切替後の従系VMのVM200を、後に障害回復した際に再度系切替を実行して、主系VMに戻す処理を行う。

<図7：VM1障害内容と系切替制御内容例>

図7は計算機ノードA 100-Aに作成されたVM1 200-Aを対象とした系切替制御テーブル304の例を示した図である。VM1 200-Aは、占有リソースとしてAP1 211-AとVM1占有メモリ101-Aを、共有リソースとしてVM1とVM2共有メモリ101-BとVM1とVM2共有外部通信I/F104-Aを使用し、論理リソースとしてVM2 200-Bとグループ関係にある。障害内容341には「VM1占有リソース障害」としてAP1 211-AとVM1占有メモリ101-Aを格納し(行352)、「VM1共有リソース障害」としてVM1とVM2共有メモリ101-BとVM1とVM2共有外部通信I/F104-Aを格納し(行353)、「VM1論理リソース障害」としてVM2 200-Bを格納する。

【0059】

行351はVM1 200-Aが存在する計算機ノードA 100-Aが障害により動

10

20

30

40

50

作継続不可となった場合の系切替規則で、計算機ノードA 100 - A内の全VM200が系切替対象となり、「過半数判断ロジック」と「フェイルバック」処理は適用しない。

【0060】

行352は占有リソースのAP1 211 - AとVM1占有メモリ101 - Aが障害で使用不能となった場合の系切替規則で、VM1 200 - Aのみが系切替対象となり、系切替実行時に「過半数判断ロジック」を適用し、占有リソースの障害回復時には「フェイルバック」処理を行う。

【0061】

行353は共有リソースのVM1とVM2共有メモリ101 - BとVM1とVM2共有外部通信I/F104 - Aが障害で使用不能となった場合の系切替規則で、共に同じリソースを使用するVM1 200 - AとVM2 200 - Bが系切替対象となり、系切替実行時に「過半数判断ロジック」を適用し、共有リソースの障害回復時には「フェイルバック」処理を行う。

10

【0062】

行354はグループ関係にあるVM2 200 - Bが障害で動作不能となった場合の系切替規則で、同一グループ内のVM1 200 - AとVM2 200 - Bが系切替対象となり、系切替実行時に「過半数判断ロジック」を適用し、VM2 200 - Bの障害回復時には「フェイルバック」処理を行う。

<図8：計算機ノード間通信フォーマット>

図8は計算機ノード間通信フォーマット500を示す図である。計算機ノード間通信フォーマット500は異なる計算機ノード100の多重系制御基盤プログラム300の間で送受信される通信データのフォーマットで、計算機ノード100の動作状況を取得する「生死監視」、及び主系VM/従系VMの系切替要求である「フェイルバック」、「異常VM系切替」、「全VM系切替」の用途で用いられる。

20

【0063】

計算機ノード識別子501には通信データを生成した計算機ノード100を特定する識別子を格納する。通信データ生成時には、送受信される当該通信データを識別する情報をシーケンス番号502に格納し、また当該通信データ生成時の時刻情報を送信タイムスタンプ503に格納する。

【0064】

通信種類504は通信データの用途を識別する情報を格納する。通信用途を識別する情報として、生死504 - A、Fail Back 504 - B、異常VM 504 - C、全VM 504 - Dのうちの一つを使用する。従系ノードから主系ノードへ「生死監視」を行う場合には生死504 - Aを使用し、主系ノードから従系ノードへ異常状態になった特定の主系VMの系切替を指示する「異常VM系切替」の場合には異常VM 504 - Cを使用し、全ての主系VMの系切替を指示する「全VM系切替」の場合には全VM 504 - Dを使用し、VM200の状態が回復した際に切替っていた主系VM/従系VMの状態を元に戻す「フェイルバック」を指示する場合にはFail Back 504 - Bを使用する。

30

【0065】

計算機ノード100の間でデータ通信を行う際には、相手計算機ノード100に自計算機ノード100内のVM200の状態を通知するために、VM状態管理テーブル内容505にはVM状態管理テーブル302の内容を格納し、リソース管理テーブル内容506にはリソース管理テーブル303の内容を格納する。

40

「生死監視」などの用途での計算機ノード100間でデータ通信では、相手計算機ノード100状態の異常発生などにより、発信したデータ通信に対する応答が返ってこないなどの応答異常が発生する場合がある。応答異常カウント507には応答異常の発生回数を格納する。

<図9：多重系制御エージェントプログラムと多重系制御基盤プログラム間の通信フォーマット>

図9は多重系制御エージェントプログラムと多重系制御基盤プログラム間の通信フォー

50

マット 600 を示す図である（以降、プログラム間通信フォーマット 600 と称する）。プログラム間通信フォーマット 600 は計算機ノード 100 内の多重系制御基盤プログラム 300 と多重系制御エージェントプログラム 212 の間で送受信されるデータ通信のフォーマットである。

【0066】

VM 識別子 601 には通信元 / 先の VM 200 を特定する情報を格納し、シーケンス番号 602 には送受信される通信データを識別する情報を格納し、送信タイムスタンプ 603 には通信データ生成時の時刻情報を格納する。

【0067】

多重系制御基盤プログラム 300 と多重系制御エージェントプログラム 212 の間で VM 200 が使用するリソース情報の通知を行う場合には、リソース管理テーブル内容 604 にリソース管理テーブル 303 の内容を格納する。

10

【0068】

多重系制御基盤プログラム 300 と多重系制御エージェントプログラム 212 の間で主系 VM / 従系 VM の系切替要求を通知したい場合には、系切替要求 605 に値を設定する。本発明の形態では、系切替要求がある場合には「1」を設定し、系切替要求が無い場合には「0」を設定するが、系切替要求の有無が識別可能であれば「0 / 1」値に限られるものではない。

< 図 10 : 仮想化多重系システムの全体フローチャート >

図 10 は仮想化多重系システム 1 で実行される処理の全体フローチャートである。仮想化多重系システム 1 の全体処理は、多重系制御基盤プログラム 300 の起動処理及び系切替通知処理 S116、多重系制御エージェントプログラム 212 の起動処理及びリソース監視処理 S106 と系切替処理、計算機ノード監視部 305 のメッセージ送信処理 S112 及びノード監視処理 S123 より成り立つ。

20

【0069】

ホスト OS 400 及びプロセッサ 102 は多重系制御基盤プログラム 300 及び VM 200 を計算機ノード 100 のメモリ 101 内に展開すると、多重系制御基盤プログラム 300 の処理を開始する。計算機ノード監視部 305 及び VM 200 は多重系制御基盤プログラム 300 の処理の中から処理を開始される。

【0070】

多重系制御基盤プログラム 300 は処理が開始されると、初めに計算機ノード監視部 305 の起動を行う (S111)。

30

【0071】

次に多重系制御基盤プログラム 300 は記憶装置 103 から VM 状態管理テーブル 302 とリソース管理テーブル 303 をメモリ 101 内に読み込み (S112)、系切替制御テーブル 304 もメモリ 101 内に読み込む (S113)。

【0072】

多重系制御基盤プログラム 300 は制御下にある各 VM 200 のメモリ領域にリソース管理テーブル 303 をコピーし (S114)、VM 200 の処理を起動する (S115)。

40

【0073】

多重系制御基盤プログラム 300 は S115 の後、系切替通知処理 S116 を実行する。

【0074】

計算機ノード監視部 305 は多重系制御基盤プログラム 300 によって立ち上げられ処理を開始すると、多重系制御基盤プログラム 300 からの「VM 切替」または「フェイルバック」実行の送信要求の有無を判定する (S121)。

【0075】

多重系制御基盤プログラム 300 からの送信要求が存在する場合 (S121 の結果が「Y」) はメッセージ送信処理 S112 を実行し、その後ノード監視処理 S123 を実行す

50

る。多重系制御基盤プログラム300からの送信要求が存在しない場合(S121の結果が「N」)はノード監視処理S123を実行する。

【0076】

計算機ノード監視部305はノード監視処理S123の実行終了後は、再び多重系制御基盤プログラム300からの送信要求の有無判定S121から処理を実行する。

【0077】

多重系制御エージェントプログラム212は処理を開始すると、自VM200のメモリ領域内のリソース管理テーブル303を読み込む(S101)。

【0078】

多重系制御エージェントプログラム212は読み込んだリソース管理テーブル303から、VM識別子331の格納値が自VM200の識別子と一致し、かつ個別情報335-Bの格納値が「多重系制御エージェントプログラム212」である個別リソース管理情報を選別する。そして選別した個別リソース管理情報のリソース識別子332と監視周期333から状態監視対象のリソースと監視周期を取得する。(S102)

次に多重系制御エージェントプログラム212は自VM200の状態を多重系制御基盤プログラム300に通知する。多重系制御エージェントプログラム212は状態監視対象リソースの状態を取得、自VM200のメモリ領域内のリソース管理テーブル303を更新し、プログラム間通信フォーマット600の通信データを作成して多重系制御基盤プログラム300に送信する。(S103)

次に多重系制御エージェントプログラム212は多重系制御基盤プログラム300からの主系VM/従系VMの系切替要求の有無を確認する(S104)。

【0079】

系切替要求がある場合(S104の結果が「Y」)、多重系制御エージェントプログラム212は系切替処理S105を実行の後、再びS104を実行する。系切替処理S105は自VM200内で動作するAP211に対して系切替を要求する処理である。

【0080】

系切替要求が無い場合(S104の結果が「N」)、多重系制御エージェントプログラム212はリソース監視処理S106を実行の後、再びS104を実行する。

<図11：リソース監視処理概要>

図11はリソース監視処理S106の概要を示す図である。本実施形態では、多重系制御エージェントプログラム212のリソース状態は、AP211の実行状態を取得することで判断する。

【0081】

AP211が動作する時には、VM200内の任意の場所(例えば/var/run)にAP211の実行状態やエラー情報を出力するApdog700ファイルが確保される。Apdog700ファイルはAP211以外のプログラムからも参照が可能なファイルである。AP211は自身の実行状態を定期的にApdog700ファイルに出力し、多重系制御エージェントプログラム212はApdog700ファイルを参照することでAP211の実行状態の監視を行う。

【0082】

多重系制御エージェントプログラム212のリソース状態の監視は、例えばAP211リソースの稼動状態はApdog700が定期的に更新されていることを確認することで認識し、外部通信I/F104リソースの異常はAP211がApdog700に出力した端末2との外部通信異常情報を取得することで行い、論理グループ関係にある他のVM200リソースの異常はAP211がApdog700に出力した他のAP211との内部通信異常情報を取得することで行う。

リソース状態の変化を検知した多重系制御エージェントプログラム212は、リソース状態を多重系制御基盤プログラム300に通知する。

<図12：リソース監視処理のフローチャート>

図12はリソース監視処理S106の一例を示すフローチャートである。多重系制御エ

10

20

30

40

50

エージェントプログラム 212 は監視対象リソースに異常を検出すると、異常状態を多重系制御基盤プログラム 300 に通知する。

【0083】

多重系制御エージェントプログラム 212 は起動すると、初めに A P d o g 7 0 0 を参照してリソース状態の監視を開始し (S 2 0 1)、監視対象リソースの異常の有無を判断する (S 2 0 2)。

【0084】

監視対象リソースの内のいずれかに異常が認められた場合 (S 2 0 2 の結果が「 Y 」)、 V M 2 0 0 内に保持されているリソース管理テーブル 3 0 3 の該当リソースのリソース生死状態フラグ 3 3 4 に死状態である「 1 」を設定し (S 2 0 3)、リソース管理テーブル 3 0 3 をプログラム間通信フォーマット 6 0 0 のリソース管理テーブル内容 6 0 4 に設定して通信データを作成して多重系制御基盤プログラム 3 0 0 に送信して (S 2 0 4)、リソース監視処理 S 1 0 6 を終了する (S 2 0 5)。

10

【0085】

監視対象リソースの内のいずれにも異常が認められない場合 (S 2 0 2 の結果が「 N 」) は、リソース監視処理 S 1 0 6 を終了する (S 2 0 5)。

【0086】

リソース監視処理 S 1 0 6 を終了した後、多重系制御エージェントプログラム 212 は、多重系制御基盤プログラム 300 からの主系 V M / 従系 V M の系切替要求の有無を確認するために図 10 に示した S 1 0 4 を起動する。

20

< 図 13 : 系切替通知処理のフローチャート >

図 13 は本発明の実施の形態の系切替通知処理 S 1 1 6 のフローチャートである。多重系制御基盤プログラム 300 は系切替通知処理 S 1 1 6 で多重系制御エージェントプログラム 212 からの V M 2 0 0 状態の通知を取得し、計算機ノード監視部 305 に主系 V M / 従系 V M の系切替要求を行う。

【0087】

多重系制御基盤プログラム 300 は系切替通知処理 S 1 1 6 を開始すると、初めに多重系制御エージェントプログラム 212 からの V M 2 0 0 状態の通知、及び計算機ノード 100 やホスト O S 4 0 0 からのハードウェア状態検出の通知を待つ (S 3 0 1)。

【0088】

V M 2 0 0 状態やハードウェア状態の通知内容を解釈し、計算機ノード 100 内のある V M 2 0 0 に新たに異常が認められた場合 (S 3 0 2 の結果が「 Y 」)、 V M 状態管理テーブル 3 0 2 の該当 V M 2 0 0 の主 / 従系状態フラグ 3 2 5 には従系 V M 状態を意味する「 S 」を設定して、生死状態フラグ 3 2 6 には死状態を意味する「 1 」を設定する (S 3 1 1)。

30

【0089】

そして V M 状態管理テーブル 3 0 2 の障害 V M 総数 3 2 2 の値を「 1 」増加させる (S 3 1 2)。

【0090】

次に多重系制御基盤プログラム 300 は計算機ノード構成管理テーブル 3 0 1 の自計算機ノード 100 に該当する主 / 従系状態フラグ 3 1 2 を参照し、自計算機ノード 100 の主系ノード / 従系ノード状態を判別する (S 3 1 3)。

40

【0091】

主 / 従系状態フラグ 3 1 2 の値が「 S 」で自計算機ノード 100 が従系ノード状態である場合 (S 3 1 3 の結果が「 N 」)、多重系制御基盤プログラム 300 は何も処理することなく、再び V M 2 0 0 状態やハードウェア状態の通知を待つために S 3 0 1 を起動する。

【0092】

主 / 従系状態フラグ 3 1 2 の値が「 M 」で自計算機ノード 100 が主系ノード状態である場合 (S 3 1 3 の結果が「 Y 」)、多重系制御基盤プログラム 300 は異常 V M 2 0 0

50

の主系VM / 従系VMの系切替要求を行うために計算機ノード間通信フォーマット500の通信パケットを作成する(S331)。

【0093】

多重系制御基盤プログラム300は、異常VM200のリソース障害内容が「過半数判断ロジック」の適用を要するものである場合には、VM状態管理テーブル302の障害VM総数322と異常VM200の数量の和が管理VM総数321の半数以上であるかの判断を行う(S332)。

【0094】

障害VM200の総数が過半数となる場合には(S332の結果が「Y」)、正常なVM200も含めて全てのVM200を系切替対象とするために、計算機ノード間通信フォーマット500の通信パケットの通信種類504には全VM504-Dの識別値を設定する(S341)。

10

【0095】

障害VM200の総数が過半数とならない場合には(S332の結果が「N」)、異常VM200のみを系切替対象とするために、計算機ノード間通信フォーマット500の通信パケットの通信種類504には異常VM504-Cの識別値を設定する(S351)。

【0096】

計算機ノード間通信フォーマット500の通信パケットの作成が完了すると、多重系制御基盤プログラム300は計算機ノード監視部305に送信要求を行う(S342)。

【0097】

20

次に異常VM200を停止させるためにリセットを行い(S343)、リソース管理テーブル303を参照して異常VM200が使用しているリソースに対してもリセットを行い(S344)、停止させた異常VM200を再起動させるためにS115を起動する。

【0098】

VM200状態やハードウェア状態の通知内容から計算機ノード100内のあるVM200に異常が認められない場合(S302の結果が「N」)、当該VM200は「死状態」から回復したものであるかを確認するために、VM状態管理テーブル302に記録された当該VM200の生死状態フラグ326の値を参照する(S321)。

【0099】

生死状態フラグ326の値が「0」で当該VM200が「正状態」であると判断されていた場合には(S321の結果が「N」)、当該VM200の状態に変化は起きていないため、多重系制御基盤プログラム300は再び多重系制御エージェントプログラム212からの通知や計算機ノード100 / ホストOS400からのハードウェア状態検出の通知を待つために、S301を起動する。

30

【0100】

生死状態フラグ326の値が「1」で当該VM200が「死状態」であると判断されていた場合には(S321の結果が「Y」)、当該VM200は状態が正常に回復したものと判断されるため、多重系制御エージェントプログラム212から受信したプログラム間通信フォーマット600の通信データのリソース管理テーブル内容604の該当VM200の状態を、多重系制御基盤プログラム300内のVM状態管理テーブル302に反映する(S322)。

40

【0101】

次に当該VM200の主系VM / 従系VMの状態のフェイルバック要求を行うために、計算機ノード構成管理テーブル301の主 / 従系状態フラグ312を参照して、自計算機ノード100の主系ノード / 従系ノード状態を判別する(S323)。

【0102】

主 / 従系状態フラグ312の値が「S」で自計算機ノード100が従系ノードであった場合には(S323の結果が「N」)、従系ノードが自らフェイルバック要求は行わないため、再び多重系制御エージェントプログラム212からの通知や計算機ノード100 / ホストOS400からのハードウェア状態検出の通知を待つためにS301を起動する。

50

【 0 1 0 3 】

主/従系状態フラグ 3 1 2 の値が「M」で自計算機ノード 1 0 0 が主系ノードであった場合には (S 3 2 3 の結果が「Y」)、計算機ノード間通信フォーマット 5 0 0 の通信パケットを作成し (S 3 2 4)、フェイルバック要求であるため通信種類 5 0 4 には F a i l B a c k 5 0 4 - B の識別値を設定し (S 3 2 5)、計算機ノード監視部 3 0 5 に作成した通信パケットの送信要求を行い (S 3 2 6)、再び多重系制御エージェントプログラム 2 1 2 からの通知や計算機ノード 1 0 0 / ホスト OS 4 0 0 からのハードウェア状態検出の通知を待つために S 3 0 1 を起動する。

< 図 1 4 : メッセージ送信処理のフローチャート >

図 1 4 はメッセージ送信処理 S 1 1 2 のフローチャートである。計算機ノード監視部 3 0 5 は多重系制御基盤プログラム 3 0 0 から受けた他計算機ノード 1 0 0 への「フェイルバック」要求、「異常 VM 切替」要求、「全 VM 切替」要求を、メッセージ送信処理 S 1 1 2 で発信する。

10

【 0 1 0 4 】

主系 VM / 従系 VM の「全 VM 切替」が実行される場合、切替対象となる VM 2 0 0 の中には本来切替を必要としない正常状態の VM 2 0 0 が含まれることもある。「全 VM 切替」を実行するために、正常 VM 2 0 0 の系切替を行うための事前処理はメッセージ送信処理 S 1 1 2 内で実行する。

【 0 1 0 5 】

計算機ノード監視部 3 0 5 はメッセージ送信処理 S 1 1 2 を開始すると、初めに通信要求の種類が「全 VM 切替」とその他を特定するために、計算機ノード間通信フォーマット 5 0 0 の通信要求パケットの通信種類 5 0 4 を確認する (S 4 0 1)。

20

【 0 1 0 6 】

通信種類 5 0 4 の値が全 VM 5 0 4 - D の識別値ではなく通信要求が「全 VM 切替」以外の場合 (S 4 0 1 の結果が「N」)、系切替対象は状態が異常に転じた、または状態が回復した VM 2 0 0 に限られるため、S 4 1 1 を起動して通信要求パケットを仮想化多重系システム 1 内の他の計算機ノード 1 0 0 にマルチキャスト送信する。

【 0 1 0 7 】

通信種類 5 0 4 の値が全 VM 5 0 4 - D の識別値であり通信要求が「全 VM 切替」の場合は (S 4 0 1 の結果が「Y」)、正常状態の VM 2 0 0 を系切替するための処理を起動する。

30

【 0 1 0 8 】

計算機ノード監視部 3 0 5 は VM 状態管理テーブル 3 0 2 の主/従系状態フラグ 3 2 5 の値が「M」の主系 VM の VM 2 0 0 を検索する (S 4 0 2)。

【 0 1 0 9 】

検索された該当 VM 2 0 0 に系切替を要求するために、プログラム間通信フォーマット 6 0 0 の系切替要求 6 0 5 に「1」を設定した通信データを作成し (S 4 0 3)、該当 VM 2 0 0 の多重系制御エージェントプログラム 2 1 2 に対して送信する (S 4 0 4)。

【 0 1 1 0 】

次に VM 状態管理テーブル 3 0 2 の全ての VM 2 0 0 の主/従系状態フラグ 3 2 5 に従系 VM を意味する「S」を設定し、また生死状態フラグ 3 2 6 を死状態である「1」に変更する (S 4 0 5)。

40

【 0 1 1 1 】

そして VM 状態管理テーブル 3 0 2 の障害 VM 総数 3 2 2 の値を「0」にリセットして (S 4 0 6)、「全 VM 切替」の通信要求パケットを仮想化多重系システム 1 内の他の計算機ノード 1 0 0 にマルチキャスト送信する (S 4 1 1)。

【 0 1 1 2 】

S 4 1 1 で通信要求パケットを送信し終えた後は、メッセージ送信処理 S 1 1 2 を終了し (S 4 2 1)、次のノード監視処理 S 1 2 3 を起動する。

< 図 1 5 : ノード監視処理のフローチャート >

50

図15はノード監視処理S123のフローチャートである。主系ノード及び従系ノードの計算機ノード100内で動作する多重系制御基盤プログラム300の計算機ノード監視部305の間では、「生死監視」及び「フェイルバック」、「異常VM切替」、「全VM切替」要求の種別の計算機ノード間通信が行われる。計算機ノード監視部305は受信した計算機ノード間通信データの種別を判別し、種別に応じた処理の起動を行う。

【0113】

計算機ノード監視部305はノード監視処理S123の処理を開始すると、初めに他の計算機ノード100からの計算機ノード間通信パケットを受信しているかを判別する。従系ノードの計算機ノード監視部305は、「生死監視」に対する応答、および「異常VM切替」、「全VM切替」、「フェイルバック」要求の計算機ノード間通信データを受信する。主系ノードの計算機ノード監視部305は、「生死監視」の計算機ノード100状態取得要求、および「異常VM切替」、「全VM切替」、「フェイルバック」に対する応答の計算機ノード間通信データを受信する。(S501)。

10

【0114】

計算機ノード間通信パケットを受信している場合(S501の結果が「Y」)、計算機ノード監視部305は計算機ノード構成管理テーブル301の主/従系状態フラグ312を参照し、自計算機ノード100の主系ノード/従系ノード状態を判別する。主/従系状態フラグ312が「M」で自計算機ノード100が主系ノードの場合(S502の結果が「Y」)は主系処理S503を起動し、主/従系状態フラグ312が「S」で自計算機ノード100が従系ノードの場合(S502の結果が「N」)は従系処理S504を起動する。(S502)

20

主系処理S503または従系処理S504が終了すると、計算機ノード監視部305はノード監視処理S123を終了する(S531)。

【0115】

計算機ノード間通信データを受信していない場合(S501の結果が「N」)、計算機ノード監視部305は計算機ノード構成管理テーブル301の主/従系状態フラグ312を参照し、自計算機ノード100の主系ノード/従系ノード状態を判別する。

【0116】

主/従系状態フラグ312が「M」で自計算機ノード100が主系ノードの場合(S511の結果が「N」)はノード監視処理S123を終了し、主/従系状態フラグ312が「S」で自計算機ノード100が従系ノードの場合(S511の結果が「Y」)は「生死監視」要求に対する応答が返ってこないことであるため次のS512を起動する。(S511)

30

S512では、「生死監視」要求に対する応答の無い計算機ノード100の計算機ノード構成管理テーブル301の生死確定カウンタ316の値と、計算機ノード間通信フォーマット500の応答異常カウンタ507の値を比較し、当該計算機ノード100の生死状態の判別を行う。

【0117】

応答異常カウンタ507の値が生死確定カウンタ316の値未満の場合(S512の結果が「N」)は当該計算機ノード100がまだ正常状態であると判断して生死監視処理S513を起動し、応答異常カウンタ507の値が生死確定カウンタ316の値以上の場合(S512の結果が「Y」)は当該計算機ノード100が異常状態にあると判断して全VM切替処理S521を起動する。

40

【0118】

生死監視処理S513または全VM切替処理S521が終了すると、計算機ノード監視部305はノード監視処理S123を終了する(S531)。

【0119】

ノード監視処理S123の終了後、計算機ノード監視部305は多重系制御基盤プログラム300からの送信要求の有無判定S121を起動する。

<図16：生死監視処理のフローチャート>

50

図16は本発明の実施の形態の生死監視処理S513のフローチャートである。従系ノードである計算機ノード100の多重系制御基盤プログラム300の計算機ノード監視部305は生死監視処理S513で「生死監視」要求を実行する。

【0120】

従系ノードの計算機ノード監視部305は生死監視処理S513を起動すると、計算機ノード構成管理テーブル301の主系ノードである計算機ノード100の監視周期315を参照する(S601)。

【0121】

前回の「生死監視」要求からの経過時間が監視周期315を超えていない場合(S601の結果が「N」)は生死監視処理S513を終了する。

10

【0122】

経過時間が監視周期315を超えている場合(S601の結果が「Y」)は「生死監視」要求を行うために次のS602を起動する。

【0123】

主系ノードである計算機ノード100に送る計算機ノード間通信フォーマット500の packets を作成するために、自計算機ノード100のVM状態管理テーブル302をVM状態管理テーブル内容505に、リソース管理テーブル303の内容をリソース管理テーブル内容506に設定する(S602)。そして通信種類504には生死504-Aの識別値を設定する(S603)。

【0124】

20

次に作成した packets を、仮想化多重系システム1内の主系ノードである計算機ノード100に対してマルチキャストで送信する(S604)。

【0125】

「生死監視」要求の packets 送信が完了すると生死監視処理S513を終了する。

<図17：全VM切替処理のフローチャート>

図17は全VM切替処理S521のフローチャートである。従系ノードである計算機ノード100で動作する多重系制御基盤プログラム300は、全VM切替処理S521を実行して、当該計算機ノード100内の全てのVM200を主系VM状態に、当該計算機ノード100を主系ノード状態に変更する。

【0126】

30

多重系制御基盤プログラム300は全VM切替処理S521を起動すると、初めに計算機ノード構成管理テーブル301の自計算機ノード100に該当する主/従系状態フラグ312に「M」を設定し、主系ノード状態に変更する(S701)。

【0127】

次に自計算機ノード100内の全てのVM200の系切替を行うために、プログラム間通信フォーマット600の系切替要求605に系切替の実行を指定する「1」を設定した、通信データを作成する(S702)。

【0128】

次にS703で自計算機ノード100内の全てのVM200の多重系制御エージェントプログラム212に対して作成したプログラム間通信フォーマット600の通信データを送信した後、S704で全VM切替処理S521を終了する。

40

<図18：主系処理のフローチャート>

図18は主系処理S503のフローチャートである。ノード監視処理S123のS502から起動された主系処理S503は、従系ノードである計算機ノード100からの「生死監視」要求、および「フェイルバック」、「異常VM切替」、「全VM切替」の応答に対する処理を行う。

【0129】

処理を開始した主系処理S503は、初めに計算機ノード間通信フォーマット500の受信 packets の通信種類504を参照し(S801)、「生死監視」要求の受信 packets であるかを判別する。

50

【 0 1 3 0 】

受信パケットの通信種類 5 0 4 が生死 5 0 4 - A の識別値である場合 (S 8 0 2 の結果が「 Y 」)、 S 8 1 1 を起動して自計算機ノード 1 0 0 の状態の返信処理を開始する。受信パケットの通信種類 5 0 4 が生死 5 0 4 - A の識別値でなかった場合 (S 8 0 2 の結果が「 N 」)、受信パケットの種別を更に特定するために S 8 2 1 を起動する。(S 8 0 2)

S 8 1 1 では計算機ノード間通信フォーマット 5 0 0 の返信パケットに自計算機ノード 1 0 0 の VM 状態管理テーブル 3 0 2 の内容を VM 状態管理テーブル内容 5 0 5 に設定し、リソース管理テーブル 3 0 3 の内容をリソース管理テーブル内容 5 0 6 に設定する。

【 0 1 3 1 】

次に作成した「生死監視」要求に対する返信パケットを、 S 8 1 2 で従系ノードである計算機ノード 1 0 0 にマルチキャストで送信した後、主系処理 S 5 0 3 を終了する。

【 0 1 3 2 】

S 8 2 1 では受信パケットが「フェイルバック」の応答であるかを判別する。受信パケットの通信種類 5 0 4 が F a i l B a c k 5 0 4 - B の識別値である場合 (S 8 2 1 の結果が「 Y 」)、主系ノードである計算機ノード 1 0 0 での「フェイルバック」処理を行うために S 8 3 1 を起動する。受信パケットの通信種類 5 0 4 が F a i l B a c k 5 0 4 - B の識別値でない場合 (S 8 2 1 の結果が「 N 」)、受信パケットの種別を更に特定するために S 8 2 2 を起動する。

【 0 1 3 3 】

S 8 3 1 では VM 状態管理テーブル 3 0 2 の主 / 従系状態フラグ 3 2 5 を参照して、値が「 S 」である従系 VM の VM 2 0 0 を探索し、これを主系 VM への系切替対象とする。

【 0 1 3 4 】

次にプログラム間通信フォーマット 6 0 0 の系切替要求 6 0 5 に系切替要求である「 1 」を設定した通信データを作成する (S 8 3 2)。

【 0 1 3 5 】

次の S 8 3 3 において、 S 8 3 1 で探索した系切替対象の VM 2 0 0 の多重系制御エージェントプログラム 2 1 2 に S 8 3 2 で作成した通信データを送信した後、主系処理 S 5 0 3 を終了する。

【 0 1 3 6 】

S 8 2 2 では受信パケットが「全 VM 切替」の応答であるかを判別する。受信パケットの通信種類 5 0 4 が全 VM 5 0 4 - D の識別値である場合 (S 8 2 2 の結果が「 Y 」)、 S 8 2 3 を起動して計算機ノード構成管理テーブル 3 0 1 の自計算機ノード 1 0 0 に該当する主 / 従系状態フラグ 3 1 2 に「 S 」を設定することで自計算機ノード 1 0 0 を従系ノード状態に変更した後、主系処理 S 5 0 3 を終了する。

【 0 1 3 7 】

受信パケットの通信種類 5 0 4 が全 VM 5 0 4 - D の識別値でない場合 (S 8 2 2 の結果が「 N 」) は主系処理 S 5 0 3 を終了する。

< 図 1 9 : 従系処理のフローチャート >

図 1 9 は従系処理 S 5 0 4 のフローチャートである。ノード監視処理 S 1 2 3 の S 5 0 2 から起動された従系処理 S 5 0 4 は、主系ノードである計算機ノード 1 0 0 からの「生死監視」への応答、及び「フェイルバック」、「異常 VM 切替」、「全 VM 切替」要求に対する処理を行う。

【 0 1 3 8 】

処理を開始した従系処理 S 5 0 4 は、初めに計算機ノード間通信フォーマット 5 0 0 の受信パケットの通信種類 5 0 4 を参照し (S 9 0 1)、「生死監視」への応答の受信パケットであるかを判別する。受信パケットの通信種類 5 0 4 が生死 5 0 4 - A の識別値である場合 (S 9 0 2 の結果が「 Y 」) は S 9 1 1 を起動して「生死監視」の処理を行い、通信種類 5 0 4 が生死 5 0 4 - A の識別値で無い場合 (S 9 0 2 の結果が「 N 」) は受信パケットの種別を更に特定するために S 9 2 1 を起動する。(S 9 0 2)

10

20

30

40

50

S 9 1 1において、「生死監視」要求に対する返答が合ったことから主系ノードの計算機ノード100は正常状態にあると判断できるため、計算機ノード間通信フォーマット500の通信パケットの応答異常カウンタ507の値を「0」にリセットした後、従系処理S504を終了する。

【0139】

S 9 2 1では受信パケットが「フェイルバック」要求であるかを判別する。受信パケットの通信種類504がFailBack504-Bの識別値である場合(S 9 2 1の結果が「Y」)、従系ノードである計算機ノード100での「フェイルバック」処理を行うためにS 9 4 1を起動する。受信パケットの通信種類504がFailBack504-Bの識別値で無い場合(S 9 2 1の結果が「N」)、受信パケットの種別を更に特定するためにS 9 2 2を起動する。

10

【0140】

S 9 4 1では計算機ノード間通信フォーマット500の受信パケットのVM状態管理テーブル内容505に格納されているVM状態管理テーブル302情報を参照して、主/従系状態フラグ325の値が「S」である従系VM状態にある主系ノードの計算機ノード100内のVM200を検索する。

【0141】

次に、自計算機ノード100の識別子及び自計算機ノード100内のVM200の識別子を、S 9 4 1で検索したVM200の主/従系構成VM情報327のホスト計算機ノード識別子327-A及びVM識別子327-Bと比較し、系切替に該当する自計算機ノード100内のVM200を検索する(S 9 4 2)。

20

【0142】

次に該当VM200に系切替要求を行うために、S 9 4 3でプログラム間通信フォーマット600の系切替要求605に系切替要求である「1」を設定した通信データを作成し、S 9 4 4で該当VM200の多重系制御エージェントプログラム212に作成した通信データを送信する。

【0143】

次に主系ノードの計算機ノード100に「フェイルバック」の応答を行うために、S 9 4 5で自計算機ノード100のVM状態管理テーブル302とリソース管理テーブル303の内容を計算機ノード間通信フォーマット500のVM状態管理テーブル内容505とリソース管理テーブル内容506に格納した通信パケットを作成し、次のS 9 4 6では通信パケットの通信種類504にFailBack504-Bの識別値を設定する。

30

【0144】

次のS 9 4 7で、作成した通信パケットを仮想化多重系システム1内の主系ノードである計算機ノード100にマルチキャストで送信した後、従系処理S504を終了する。

【0145】

S 9 2 2では受信パケットが「全VM切替」要求であるかを判別する。受信パケットの通信種類504が全VM504-Dの識別値である場合(S 9 2 2の結果が「Y」)、従系ノードである計算機ノード100での「全VM切替」処理を行うために全VM切替処理S521を起動する。受信パケットの通信種類504が全VM504-Dの識別値で無い場合(S 9 2 2の結果が「N」)、従系ノードである計算機ノード100での「異常VM切替」処理を行うためにS 9 2 3を起動する。

40

【0146】

全VM切替処理S521の終了後、主系ノードの計算機ノード100に「全VM切替」の応答を行うために、S 9 3 2で自計算機ノード100のVM状態管理テーブル302とリソース管理テーブル303の内容を計算機ノード間通信フォーマット500のVM状態管理テーブル内容505とリソース管理テーブル内容506に格納した通信パケットを作成し、次のS 9 3 3では通信パケットの通信種類504に全VM504-Dの識別値を設定する。

【0147】

50

次のS 9 3 4で、作成した通信パケットを仮想化多重系システム1内の主系ノードである計算機ノード1 0 0にマルチキャストで送信した後、従系処理S 5 0 4を終了する。

【0 1 4 8】

S 9 2 3では、「異常VM切替」処理を行うために、計算機ノード間通信フォーマット5 0 0の受信パケットのVM状態管理テーブル内容5 0 5に格納されているVM状態管理テーブル3 0 2情報を参照して、主ノード状態フラグ3 2 5の値が「S」である従系VM状態にある主系ノードの計算機ノード1 0 0内のVM 2 0 0を検索する。

【0 1 4 9】

次に、自計算機ノード1 0 0の識別子及び自計算機ノード1 0 0内のVM 2 0 0の識別子を、S 9 2 3で検索したVM 2 0 0の主ノード構成VM情報3 2 7のホスト計算機ノード識別子3 2 7 - A及びVM識別子3 2 7 - Bと比較し、系切替に該当する自計算機ノード1 0 0内のVM 2 0 0を検索する(S 9 2 4)。

【0 1 5 0】

次に該当VM 2 0 0に系切替要求を行うために、S 9 2 5でプログラム間通信フォーマット6 0 0の系切替要求6 0 5に系切替要求である「1」を設定した通信データを作成し、次のS 9 2 6で該当VM 2 0 0の多重系制御エージェントプログラム2 1 2に作成した通信データを送信し、従系処理S 5 0 4を終了する。

【0 1 5 1】

以上の実施形態によれば、物理計算機上で障害検知した主系モードの仮想マシンが搭載済みの仮想マシンよりも半数未満の場合には、仮想マシン毎に対して系切替え(フェイルオーバー)を行った後にフェイルバックする。一方、物理計算機上で障害検知した主系モードの仮想マシンが搭載済みの仮想マシンよりも半数以上の場合には、全ての仮想マシンに対して系切替えを行う。以上によって、単一の主系モードの仮想マシンが障害を起こしたとしても、複数の主系モードの仮想マシンが障害を起こしたとしても、つまり、いずれの場合にも、最終的には物理計算機の片方に全ての仮想マシンが主系モードで稼働するようになり、利用者にとって物理計算機の全体の動作モードが管理しやすくなる。その結果、物理計算機上の全ての仮想マシンを主系モードで稼働させながら、もう一方の物理計算機に対してハードウェアやソフトウェアの保守を行うことが可能となる。

【符号の説明】

【0 1 5 2】

- 1 ... 仮想化多重系システム
- 2 ... 端末
- 3 ... 外部ネットワーク
- 4 ... 内部ネットワーク
- 1 0 0 ... 計算機ノード
- 1 0 1 ... メモリ
- 1 0 2 ... プロセッサ
- 1 0 3 ... 記憶装置、
- 1 0 4 ... 外部通信 I / F
- 1 0 5 ... ノード間通信 I / F
- 2 0 0 ... VM
- 2 0 1 ... VM
- 2 1 1 ... AP
- 2 1 2 ... 多重系制御エージェントプログラム
- 2 1 3 ... ゲストOS
- 3 0 0 ... 多重系制御基盤プログラム
- 3 0 1 ... 計算機ノード構成管理テーブル
- 3 0 2 ... VM状態管理テーブル
- 3 0 3 ... リソース管理テーブル
- 3 0 4 ... 系切替制御テーブル

10

20

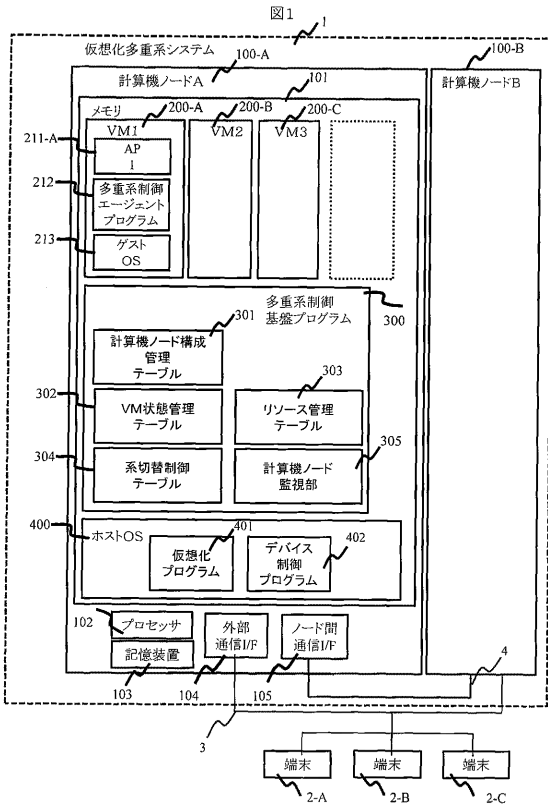
30

40

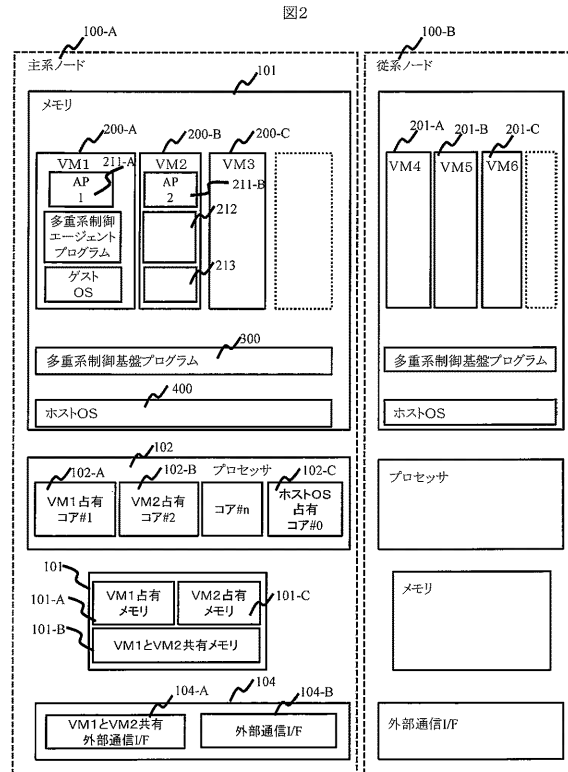
50

- 3 0 5 ... 計算機ノード監視部
- 4 0 0 ... ホストOS
- 4 0 1 ... 仮想化プログラム
- 4 0 2 ... デバイス制御プログラム
- 7 0 0 ... A P d o g

【 図 1 】



【 図 2 】



【図3】

図3

計算機ノード構成管理テーブル

311 計算機ノード識別子	312 主/従系状態フラグ	313 マルチキャストアドレス	314 生死状態フラグ	315 監視周期 [ms]	316 生死確定カウンタ
100-A	M	226.94.1.1	0	1000	3
100-B	S	226.94.1.1	0	1000	3

【図4】

図4

VM状態管理テーブル

321 管理VM総数	322 障害VM総数			327 主/従系構成VM情報	
323 VM識別子	324 ホスト計算機ノード識別子	325 主/従系状態フラグ	326 生死状態フラグ	327-A 327-B 主/従系構成VM情報	327 主/従系構成VM情報
3					0
VM1	100-A	M	0	100-B	VM4
VM2	100-A	M	0	100-B	VM5
VM3	100-A	M	0	100-B	VM6

【図7】

図7

VM1障害内容と系切替制御内容例

351 障害内容	352 系切替対象VM	353 過半数判断ロジック適用要否	354 Failback処理適用可否
計算機ノード障害	全VM切替	否	否
VM1占有リソース障害	AP1	VM1切替	要
	101-A		
VM1共有リソース障害	101-B	VM1、VM2切替	要
	104-A		
VM1論理リソース障害	VM2	VM1、VM2切替	要

【図8】

図8

計算機ノード間通信フォーマット

501 計算機ノード識別子	502 シーケンス番号	503 送信タイムスタンプ	504 通信種類	505 VM状態管理テーブル内容	506 リソース管理テーブル内容	507 応答異常カウンタ
200-A	001	HHMMSS	504-A 生死	302	303	0
			504-B 系切替			
			504-C Fail Back			
			504-D 異常VM			

【図5】

図5

リソース管理テーブル

331 VM識別子	332 リソース識別子	333 監視周期 [ms]	334 リソース生死状態フラグ	335-A 335-B リソース詳細情報
VM1	AP1	100	0	占有 212
VM1	101-A	0	0	占有 EVT
VM1	101-B	0	0	共有 EVT
VM1	104-A	50	0	共有 212
VM1	VM2	100	0	論理 /SERVER 212
VM2	AP2	100	0	占有 212
VM2	101-C	0	0	占有 EVT
VM2	101-B	0	0	共有 EVT
VM2	104-A	50	0	共有 212
VM2	VM1	100	0	論理/CLIENT 212

【図6】

図6

系切替制御テーブル

341 障害内容	342 系切替対象VM	343 過半数判断ロジック適用要否	344 Failback処理適用可否
計算機ノード障害	全VM	否	否
VM占有リソース障害	該当VMのみ	要	可
VM共有リソース障害	関連全VM	要	可
VM論理リソース障害	関連全VM	要	可

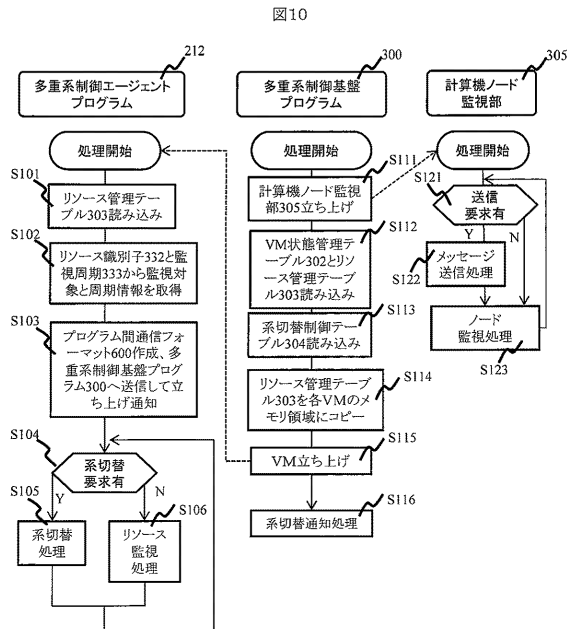
【図9】

図9

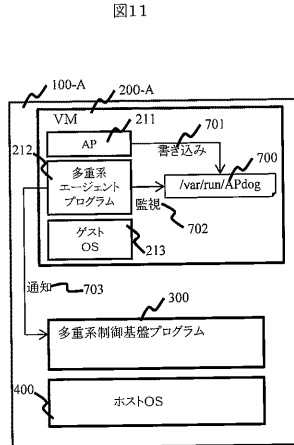
プログラム間通信フォーマット

601 VM識別子	602 シーケンス番号	603 送信タイムスタンプ	604 リソース管理テーブル内容	605 系切替要求
323	001	HHMMSS (例:081203)	303	0

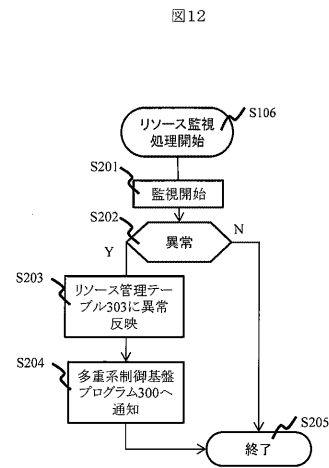
【図10】



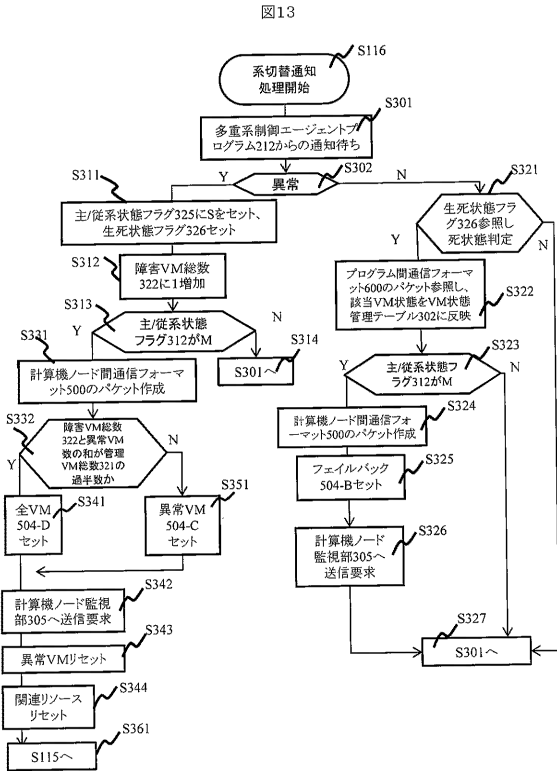
【図11】



【図12】

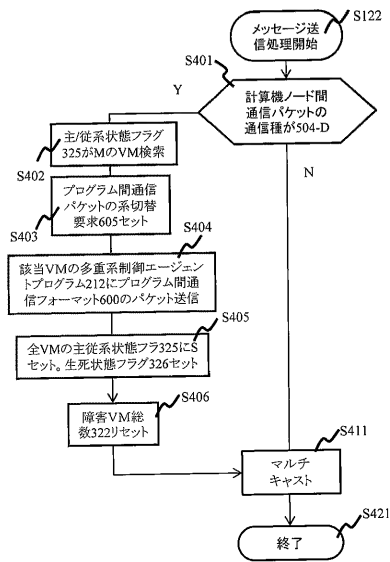


【図13】



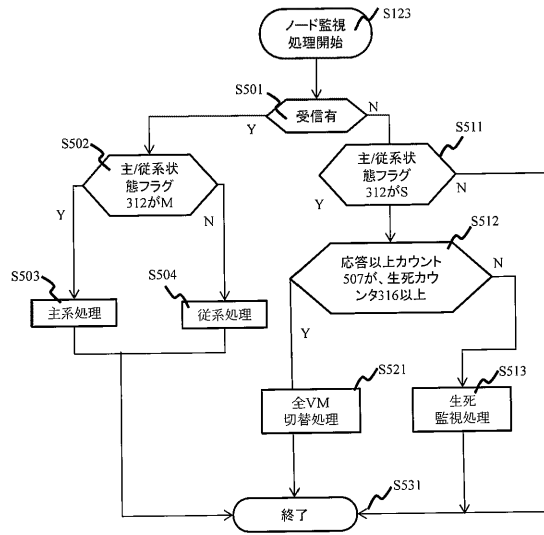
【図14】

図14



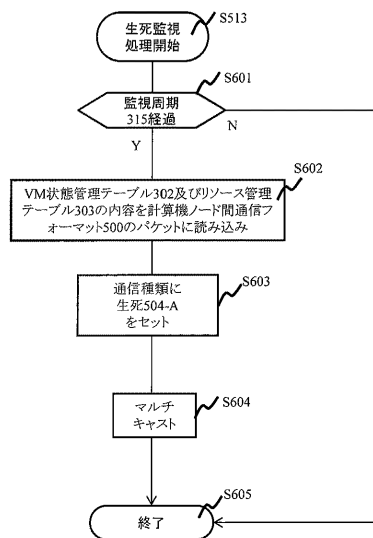
【図15】

図15



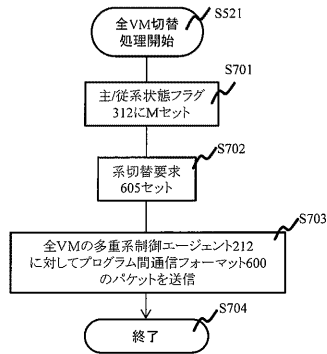
【図16】

図16

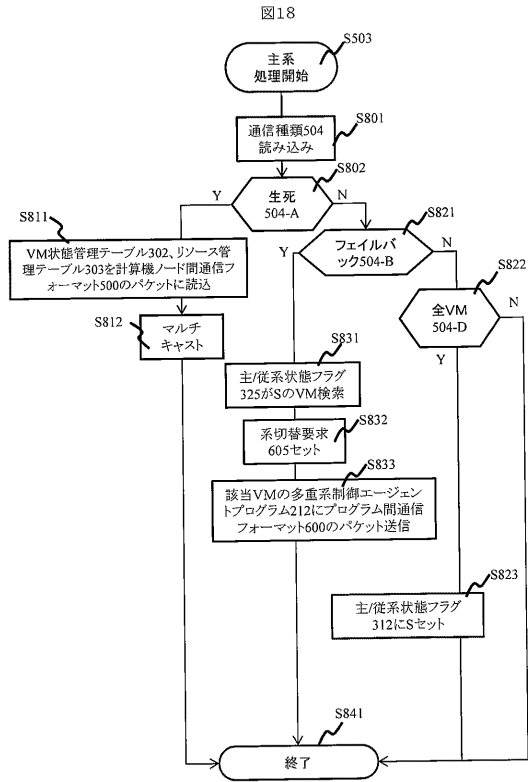


【図17】

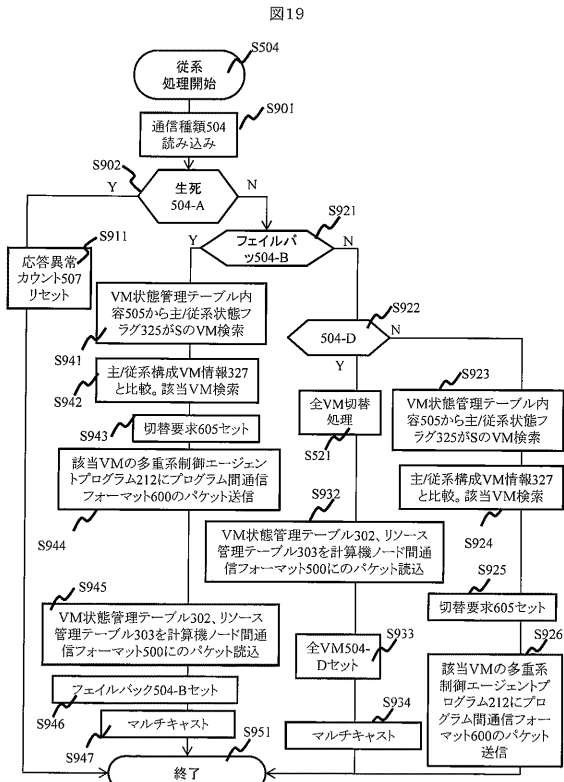
図17



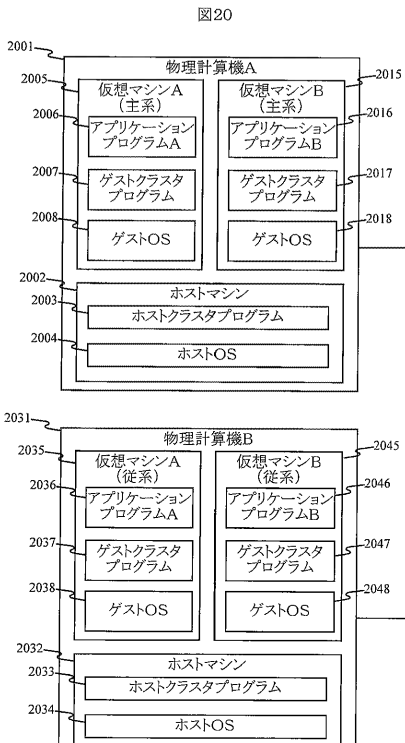
【図18】



【図19】



【図20】



フロントページの続き

- (56)参考文献 特開2008-269332(JP,A)
特開2009-211517(JP,A)
特開2005-327279(JP,A)
特開2009-187090(JP,A)
特開2007-148839(JP,A)
特開2009-140194(JP,A)
特開2009-129148(JP,A)

(58)調査した分野(Int.Cl., DB名)

G06F 11/16 - 11/20,
G06F 9/46 - 9/54