

(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2012-146152  
(P2012-146152A)

(43) 公開日 平成24年8月2日(2012.8.2)

(51) Int.Cl.

G06F 9/50 (2006.01)

F I

G06F 9/46 462A

テーマコード (参考)

審査請求 未請求 請求項の数 12 O L (全 37 頁)

(21) 出願番号 特願2011-4454 (P2011-4454)  
(22) 出願日 平成23年1月13日 (2011.1.13)

(71) 出願人 00004237  
日本電気株式会社  
東京都港区芝五丁目7番1号  
(74) 代理人 100079108  
弁理士 稲葉 良幸  
(74) 代理人 100109346  
弁理士 大貫 敏史  
(74) 代理人 100117189  
弁理士 江口 昭彦  
(74) 代理人 100134120  
弁理士 内藤 和彦  
(74) 代理人 100109586  
弁理士 土屋 徹雄  
(72) 発明者 大井 堂史昌  
東京都港区芝五丁目7番1号 日本電気株式会社内

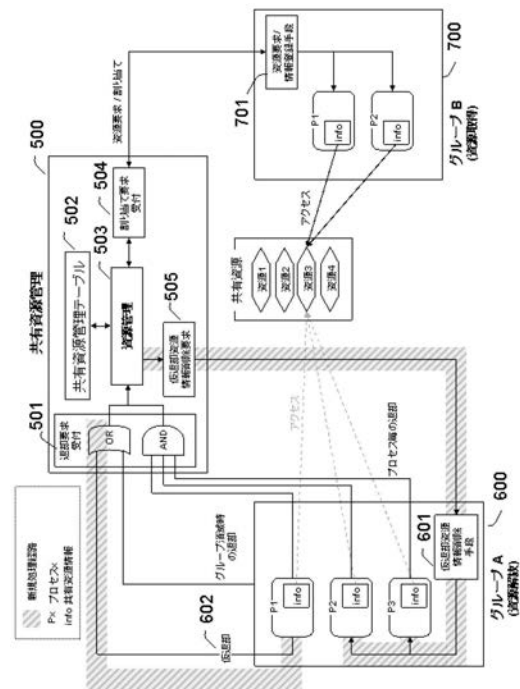
(54) 【発明の名称】 共有資源管理システム及び共有資源管理方法

(57) 【要約】

【課題】 資源運用の効率化と資源返却処理コスト削減を両立することができる共有資源システムを提供する。

【解決手段】 複数の資源からなる共有資源の割り当てと当該割り当てた共有資源の返却を管理する共有資源管理手段と、複数のプロセスを有するプロセスグループにおける前記共有資源の割り当てと返却を管理する管理マスタと、を備え、前記共有資源管理手段は、前記管理マスタから当該プロセスグループに割り当てられた資源の返却を受け付ける手段と、他のプロセスグループを管理する他の管理マスタから前記共有資源の取得要求を受け付ける手段と、前記共有資源中に未使用の資源があるかを判定し、未使用の資源がないと判定した場合は、前記返却された資源を前記他のプロセスグループに割り当てる割り当て手段と、を備える。

【選択図】 図 3 7



**【特許請求の範囲】****【請求項 1】**

複数の資源からなる共有資源の割り当てと当該割り当てた共有資源の返却を管理する共有資源管理手段と、

複数のプロセスを有するプロセスグループにおける前記共有資源の割り当てと返却を管理する管理マスタと、を備え、

前記共有資源管理手段は、

前記管理マスタから当該プロセスグループに割り当てられた資源の仮返却を受け付ける手段と、

他のプロセスグループを管理する他の管理マスタから前記共有資源の取得要求を受け付ける手段と、

前記共有資源中に未使用の資源があるか否かを判定し、未使用の資源がないと判定した場合は、前記仮返却された資源を前記他のプロセスグループに割り当てる割当手段と、を備えることを特徴とする共有資源管理システム。

10

**【請求項 2】**

前記管理マスタは、

当該管理マスタが対象とするプロセスグループ中の少なくとも 1 つのプロセスが当該プロセスグループに割り当てられた資源を返却可能な状態になった場合に、当該プロセスグループに割り当てられた資源を前記共有資源管理手段へ仮返却することを特徴とする請求項 1 に記載の共有資源管理システム。

20

**【請求項 3】**

前記共有資源管理手段は、

前記仮返却された資源を前記他のプロセスグループに割り当てる場合は、当該仮返却された資源の資源情報の削除を当該仮返却した管理マスタに要求することを特徴とする請求項 1 または 2 に記載の共有資源管理システム。

**【請求項 4】**

前記共有資源管理手段は、

前記未使用の資源がないと判定した場合において前記仮返却された資源が複数あるときは、当該複数の仮返却された資源について算定された各情報削除処理コストに基づいて、前記他のプロセスグループに割り当てる資源を選択することを特徴とする請求項 1 乃至 3 いずれか 1 項に記載の共有資源管理システム。

30

**【請求項 5】**

前記管理マスタは、

当該管理マスタが対象とするプロセスグループ中の全てのプロセスが当該プロセスグループに割り当てられた資源を返却可能な状態になった場合に、当該プロセスグループに割り当てられた資源を前記共有資源管理手段へ返却することを特徴とする請求項 1 乃至 4 いずれか 1 項に記載の共有資源管理システム。

**【請求項 6】**

前記管理マスタは、

当該管理マスタが対象とするプロセスグループが消滅した場合に、当該プロセスグループに割り当てられた資源を前記共有資源管理手段へ返却することを特徴とする請求項 1 乃至 4 いずれか 1 項に記載の共有資源管理システム。

40

**【請求項 7】**

複数の資源からなる共有資源の割り当てと当該割り当てた共有資源の返却を管理する共有資源管理手段と、複数のプロセスを有するプロセスグループにおける前記共有資源の割り当てと返却を管理する管理マスタと、を備えるシステムにおける共有資源管理方法であって、

前記共有資源管理手段が、

前記管理マスタから当該プロセスグループに割り当てられた資源の仮返却を受け付けるステップと、

50

他のプロセスグループを管理する他の管理マスタから前記共有資源の取得要求を受け付けるステップと、

前記共有資源中に未使用の資源があるか否かを判定し、未使用の資源がないと判定した場合は、前記仮返却された資源を前記他のプロセスグループに割り当てる割当ステップと、

を備えることを特徴とする共有資源管理方法。

【請求項 8】

前記管理マスタは、

当該管理マスタが対象とするプロセスグループ中の少なくとも 1 つのプロセスが当該プロセスグループに割り当てられた資源を返却可能な状態になった場合に、当該プロセスグループに割り当てられた資源を前記共有資源管理手段へ仮返却するステップを備えることを特徴とする請求項 7 に記載の共有資源管理方法。

10

【請求項 9】

前記共有資源管理手段は、

前記仮返却された資源を前記他のプロセスグループに割り当てる場合は、当該仮返却された資源の資源情報の削除を当該仮返却した管理マスタに要求するステップを備えることを特徴とする請求項 7 または 8 に記載の共有資源管理方法。

【請求項 10】

前記共有資源管理手段は、

前記未使用の資源がないと判定した場合において前記仮返却された資源が複数あるときは、当該複数の仮返却された資源について算定された各情報削除処理コストに基づいて、前記他のプロセスグループに割り当てる資源を選択するステップを備えることを特徴とする請求項 7 乃至 9 いずれか 1 項に記載の共有資源管理方法。

20

【請求項 11】

前記管理マスタは、

当該管理マスタが対象とするプロセスグループ中の全てのプロセスが当該プロセスグループに割り当てられた資源を返却可能な状態になった場合に、当該プロセスグループに割り当てられた資源を前記共有資源管理手段へ返却するステップを備えることを特徴とする請求項 7 乃至 10 いずれか 1 項に記載の共有資源管理方法。

【請求項 12】

前記管理マスタは、

当該管理マスタが対象とするプロセスグループが消滅した場合に、当該プロセスグループに割り当てられた資源を前記共有資源管理手段へ返却するステップを備えることを特徴とする請求項 7 乃至 10 いずれか 1 項に記載の共有資源管理方法。

30

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、共有資源管理システム及び共有資源管理方法に係り、特に、HPC (High Performance Computing) 分野に適用可能な共有資源管理システム及び共有資源管理方法に関する。

40

【背景技術】

【0002】

コンピュータの共有資源の割り当てや返却を管理する技術が知られている。下記特許文献 1 (特開 2001 - 195268 号公報) には、このような共有資源管理技術の一例として、資源割当手段 12 により各資源の資源量を実行要求対象の AP に割り当てることができない場合、AP 実行待ち合わせ手段 15 が、実行要求対象の AP より低い AP サービスレベルの AP の実行を中断して各資源を解放させ、その解放された各資源を実行要求対象の AP に優先的に割り当てて実行させることが記載されている。

【0003】

また、下記特許文献 2 (特開 2001 - 356921 号公報) には、資源空き状況通知

50

手段 1 c が、資源割り当て状況管理手段 1 b を参照して資源の空き状況が変化したことを検知した場合は、ジョブに対してその旨を通知し、資源再割り当て手段 1 e が、資源の再割り当ての要求がジョブからなされた場合は、それらのジョブに対して解放された資源を割り当てることが記載されている。

【 0 0 0 4 】

また、下記特許文献 3 (特開平 6 - 3 4 2 3 7 7 号公報) には、資源確保管理部 6 が、プログラム制御文 1 4 により起動されるプログラムの実行開始前に、必要な複数のプログラム使用資源 1 5 を、複数のプロセスにて分散して並列的に確保し、資源解放管理部 7 は、使用を終えた複数のプログラム使用資源 1 5 を、複数のプロセスにて分散して並列的に開放する処理を管理することが記載されている。

10

【先行技術文献】

【特許文献】

【 0 0 0 5 】

【特許文献 1】特開 2 0 0 1 - 1 9 5 2 6 8 号公報

【特許文献 2】特開 2 0 0 1 - 3 5 6 9 2 1 号公報

【特許文献 3】特開平 6 - 3 4 2 3 7 7 号公報

【発明の概要】

【発明が解決しようとする課題】

【 0 0 0 6 】

ところで、H P C (High Performance Computing) 分野のジョブはプロセス群 (グループ) によって並列処理で実行され、この並列内容はデータパラレルである場合が多いところ、これらグループの共有資源返却方式として、例えば以下の 2 つの方式が知られている。

20

【 0 0 0 7 】

第 1 の方式は、プロセスからの返却通知をとりまとめ、グループ内で資源の利用者がなくなった時点でシステムに資源を返却するものであり、例えば「プロセス毎の返却」方式と呼ばれる (図 3 3 を参照)。この方式は、実行中に資源返却が可能であるため、効率よく資源を運用することができる一方、全プロセスによる資源返却処理のとりまとめが必要であるためプロセス数の増大と共に処理コストも増加するという問題がある。

【 0 0 0 8 】

第 2 の方式は、グループ消滅時に資源を返却するものであり、例えば「グループ消滅時の返却」方式と呼ばれる (図 3 4 を参照)。この方式は、返却処理コストが一定である (プロセス数の影響をうけない) 一方、各プロセスが処理の途中で資源を必要としなくなっても処理が終了するまで資源を保持し続けるので効率のよい資源運用ができないという問題がある。

30

【 0 0 0 9 】

すなわち、第 1 の方式 (プロセス毎の返却方式) と第 2 の方式 (グループ消滅時の返却方式) は、それぞれ、資源運用の効率化と資源返却処理コストがトレードオフの関係にあり、どちらも一長一短である。よって、双方の欠点を解消するとともに双方の利点を両立することができる新たな仕組みが望まれるところ、上述した特許文献 1 ~ 3 には、かかる課題については何ら考慮されていない。

40

【 0 0 1 0 】

従って、本発明の目的は、資源運用の効率化と資源返却処理コスト削減を両立することができる共有資源システムを提供することにある。

【課題を解決するための手段】

【 0 0 1 1 】

本発明に係る共有資源管理システムは、複数の資源からなる共有資源の割り当てと当該割り当てた共有資源の返却を管理する共有資源管理手段と、複数のプロセスを有するプロセスグループにおける前記共有資源の割り当てと返却を管理する管理マスタと、を備え、前記共有資源管理手段は、前記管理マスタから当該プロセスグループに割り当てられた資

50

源の仮返却を受け付ける手段と、他のプロセスグループを管理する他の管理マスタから前記共有資源の取得要求を受け付ける手段と、前記共有資源中に未使用の資源があるか否かを判定し、未使用の資源がないと判定した場合は、前記仮返却された資源を前記他のプロセスグループに割り当てる割り当て手段と、を備える。

【0012】

また、本発明に係る共有資源管理方法は、複数の資源からなる共有資源の割り当てと当該割り当てた共有資源の返却を管理する共有資源管理手段と、複数のプロセスを有するプロセスグループにおける前記共有資源の割り当てと返却を管理する管理マスタと、を備えるシステムにおける共有資源管理方法であって、前記共有資源管理手段が、前記管理マスタから当該プロセスグループに割り当てられた資源の仮返却を受け付けるステップと、他のプロセスグループを管理する他の管理マスタから前記共有資源の取得要求を受け付けるステップと、前記共有資源中に未使用の資源があるか否かを判定し、未使用の資源がないと判定した場合は、前記仮返却された資源を前記他のプロセスグループに割り当てる割り当てステップと、を備える。

10

【発明の効果】

【0013】

本発明の共有資源システムによれば、資源運用の効率化と資源返却処理コスト削減を両立することができるようになる。

【図面の簡単な説明】

【0014】

20

【図1】 仮定システムの概略構成の一例を示す図である。

【図2】 複数ジョブが仮定システムを共有する場合の構成を示す図である。

【図3】 共有資源管理とグループ管理の一例を説明するための図である。

【図4】 共有資源管理とグループ管理の一例を説明するための図である。

【図5】 共有資源管理とグループ管理の一例を説明するための図である。

【図6】 グループ管理（マスタ）とグループ管理（スレーブ）の一例を説明するための図である。

【図7】 グループ管理（マスタ）とグループ管理（スレーブ）の一例を説明するための図である。

【図8】 共有資源管理とグループ管理の一例を説明するための図である。

30

【図9】 グループ管理（マスタ）とグループ管理（スレーブ）の一例を説明するための図である。

【図10】 仮定システムの概略構成の一例を示す図である。

【図11】 共有資源管理とグループ管理の一例を説明するための図である。

【図12】 共有資源管理とグループ管理の一例を説明するための図である。

【図13】 共有資源管理とグループ管理の一例を説明するための図である。

【図14】 共有資源管理とグループ管理の一例を説明するための図である。

【図15】 共有資源管理とグループ管理の一例を説明するための図である。

【図16】 共有資源管理とグループ管理の一例を説明するための図である。

【図17】 共有資源管理とグループ管理の一例を説明するための図である。

40

【図18】 グループ管理（マスタ）とグループ管理（スレーブ）の一例を説明するための図である。

【図19】 グループ管理（マスタ）とグループ管理（スレーブ）の一例を説明するための図である。

【図20】 共有資源管理とグループ管理の一例を説明するための図である。

【図21】 共有資源管理とグループ管理の一例を説明するための図である。

【図22】 共有資源管理とグループ管理の一例を説明するための図である。

【図23】 共有資源管理とグループ管理の一例を説明するための図である。

【図24】 共有資源管理とグループ管理の一例を説明するための図である。

【図25】 共有資源管理とグループ管理の一例を説明するための図である。

50

- 【図 2 6】共有資源管理とグループ管理の一例を説明するための図である。
- 【図 2 7】共有資源管理構成の一例を示す図である。
- 【図 2 8】資源取得の流れの一例を示すフローチャートである。
- 【図 2 9】資源共有の流れの一例を示すフローチャートである。
- 【図 3 0】資源取得の流れ（未使用資源なし）の一例を示すフローチャートである。
- 【図 3 1】資源返却の流れの一例を示すフローチャートである。
- 【図 3 2】共有資源の割り当てとアクセスを説明するための図である。
- 【図 3 3】従来一般的な資源返却方法の一例を説明するための図である。
- 【図 3 4】従来一般的な資源返却方法の一例を説明するための図である。
- 【図 3 5】代表返却方法を説明するための図である。
- 【図 3 6】本実施形態に係る共有資源管理方法を説明するための図である。
- 【図 3 7】本実施形態に係る共有資源管理システムの構成の一例を示す図である。
- 【図 3 8】従来共有資源管理システムの構成の一例を示す図である。
- 【図 3 9】本実施形態に係る共有資源管理方法を説明するための図である。
- 【発明を実施するための形態】

10

【0015】

以下、本発明の実施の形態について図面を参照しつつ詳細に説明する。なお、同一の要素には同一の符号を付し、重複する説明を省略する場合がある。

【0016】

本発明が想定する共有資源の使用方法は、共有資源をシステムが管理し、その割り当てはプロセスを束ねたグループ単位とし、1つの共有資源を複数のグループで共有することはないという構成を前提とする。図 3 2 は、このような共有資源の使用方法を示している。当該方法では、グループ A、B 内の各プロセス P 1、P 2、P 3 は、自身が属するグループが保持する共有資源にのみアクセスが可能である。グループ A、B 内の各プロセス P 1、P 2、P 3 は、グループが保持する共有資源にアクセスするために、その共有資源に関する情報を保持し、その保持情報をもとに割り当てられた共有資源にアクセスが可能となる。

20

【0017】

また、共有資源を使用するシステムは、複数のノード(計算資源)をネットワークスイッチによって接続したシステム(以下「仮定システム」という。)の構成を採用することができる。図 1 は、このような仮定システムの構成を示している。なお、ノードは物理的、論理的のどちらであってもよい。仮定システム 1 a は、各ノード 1 0 0 a ~ 1 1 1 a から共通にアクセスすることが可能な共有資源 2 0 0 a を装備する。なお、この共有資源 2 0 0 a はネットワークスイッチ 3 0 0 a に接続する形態、ネットワークスイッチ 3 0 0 a 内に装備する形態のどちらでもよい。共有資源 2 0 0 a とは、同じ機能を有する複数の資源のことであり、例えば、各ノードから共通にアクセス可能な多数のレジスタ、もしくは共有メモリ等が該当する。

30

【0018】

仮定システム 1 a の運用形態としては、ある 1 ジョブが仮定システム 1 a の全てを占有する場合と、複数のジョブで仮定システム 1 a を同時に共有する場合がある。図 2 は、5 つのジョブ A ~ E による仮定システム 1 a の共有例を示している。仮定システム 1 a を複数のジョブで共有する場合、あるジョブの計算結果が他のジョブの計算結果に影響を及ぼさないことを保証するために、仮定システム 1 a は、あるジョブが使用するノードを予めグループ化した上でジョブを実行する。よって仮定システム 1 a は、ノードをグループ化する手段と、それを管理する機能を有する。ここで仮定システム 1 a によってグループ化されるノードは、ジョブの実行前に用意されるが、ジョブの実行中に、グループに対するノードの追加、削除に関する機能の是非を問わない。

40

【0019】

仮定システム 1 a を複数ジョブで共有可能な場合、仮定システム 1 a に接続された共有資源 2 0 0 a は、グループ毎に割り当てられるものであり、グループ間で共有をしないも

50

のとする。このため仮定システム1は、共有資源200aをグループに割り当てる際、資源を割り当てた結果により、他ジョブの計算結果に影響を及ぼすことがないことを保証する必要がある。よって仮定システム1aは、共有資源に関して、グループからの「資源の割り当て要求」と「資源の返却要求」を受け付け、各グループに対して排他的な割り当てを保証するための「共有資源管理」を実行する。

#### 【0020】

次に、このようなグループが仮定システム1aに対して、共有資源200aの取得を要求し、グループ内の各ノードで共有資源を200a共有し、共有資源200aを返却するまでの流れについて説明する。

#### 【0021】

図3は、共有資源管理テーブル、グループ管理テーブルの構成の一例を示している。同図で示される共有資源は、同じ機能を有する資源を複数個有し、各資源は資源番号で区別される。ここでグループは「グループ管理(マスタ)」と「グループ管理(スレーブ)」によって構成される。グループ管理(マスタ)は、グループ内のグループ管理(スレーブ)を管理し、グループ管理(スレーブ)はノード内のプロセスを管理する。

#### 【0022】

共有資源管理(「共有資源管理手段」ともいう)における各資源の管理情報は、例えば、グループへの割り当て状況を示すフラグ「状態」、資源を割り当てたグループの管理情報をもつノード番号を保持する「グループ管理ノード」、グループを識別する「グループID」である。例えば図3では、資源番号1の共有資源は、「状態」が「使」であるため割り当てがされており、割り当てたグループは「グループID」に記載された「X」で、グループを管理するテーブル(グループ管理(マスタ))の所在ノードは「グループ管理ノード」に記される「#05」であることが示されている。また、ここで「状態」が「未」である共有資源は、割り当てされていない未使用資源を意味する。

#### 【0023】

グループ管理(マスタ)(「グループ管理手段(マスタ)」ともいう)は、当該ジョブが他ジョブの計算結果に影響を与えないことを保証するために組まれるグループを管理する。保持情報は、グループを識別するための「グループID」、当該グループに所属するノード情報を保持する「参加ノード情報」、グループがシステムより割り当てられた共有資源の番号と、その資源を使用するノードの数をカウントする「保有共有資源情報」である。例えば図3では「グループID」は「X」で、当該グループは「#04、#05、#06、#07」の4ノードから構成されており、共有資源1を保有し、その共有資源を使用しているノードは3ノードであることが示されている。ここで保有共有資源情報の共有数に関して、使用ノード数の管理は行なうが、使用ノード番号の情報は管理しない。これは保持可能な共有資源毎に、使用ノード情報をもつ場合、参加ノード数が多い大規模グループでは、その管理テーブルが巨大になるためである。グループ管理(マスタ)において、共有資源の「管理」として必要とされる情報は、共有資源を仮定システムに返却するタイミングを判断する情報であり、それはグループ内で共有資源の利用者がいなくなったときである。そのため、利用者数の管理のみでよい。

#### 【0024】

グループ管理(スレーブ)(「グループ管理手段(スレーブ)」ともいう)は、各ノードが、どのグループに属するかを管理する。保持情報は、所属グループを識別するための「グループID」、グループ管理の「マスタ」が存在するノード番号を保持する「マスタノード」、当該ノードが使用できる共有資源番号を保持する「共有資源番号」である。例えば図3では、「グループID」は「X」であり、グループ管理のマスタはノード「#05」に存在し、当該ノードが使用できる共有資源は1番の共有資源である。

#### 【0025】

以上の管理テーブルを用いて、共有資源の取得について説明する。図4は、初期状態を示している。以下、グループID「X」をもつグループを「グループX」と呼ぶ。今、グループXは、当該グループに属するノードが「#04、#05、#06」の3ノードで構成されており

10

20

30

40

50

、グループが保有する資源として共有資源1を保持している。共有資源1はグループ内の各ノードから使用可能な状態にあるため、その共有数は3である。また各ノードのグループ管理(スレーブ)の状態は、現在、みな同じであるため、代表として1つだけを示している。この状態でグループXが新たに共有資源を保有する場合、グループ管理(マスタ)は仮定システム1の「共有資源管理」に対して、共有資源の割り当てを要求する(図4-(1))。続いて「共有資源管理」は要求に応えるために、共有資源管理テーブルを探索し、未使用資源として共有資源2を見つける(図4-(2))。資源を割り当てるために、共有資源管理テーブル上の共有資源2に対応する情報を「状態：未使」、「グループ管理ノード：#05」、「グループID：X」に変更する(図5-(3))。「共有資源管理」は割り当て共有資源の番号2を、割り当て要求したグループ管理(マスタ)に通知する(図5-(4))。グループ管理(マスタ)は、通知された共有資源番号2を自身の管理テーブルに登録する(図5-(5))。このとき共有資源2はグループとして所有されただけであり、各ノードから、その資源の使用の申し出を受けていないため、共有数は0である。続いてグループ管理(スレーブ #04)が、共有資源2を使用するために、グループ管理(マスタ)に対して、共有資源2の共有を要求する(図6-(6))。グループ管理(マスタ)は、グループ管理(スレーブ #04)からの要求に応えるために、管理テーブルの共有資源2に対応する共有数を1加算し(図6-(7))、共有番号2をグループ管理(スレーブ #04)に通知する(図6-(8))。要求の応答を受けたグループ管理(スレーブ #04)は、返却値である共有資源番号2を自身の管理テーブルに登録する(図6-(9))。

10

#### 【 0 0 2 6 】

以上の手順を経て、グループ管理(スレーブ #04)のノードから共有資源2へアクセスすることが可能となる(図6-(10))。この時点でグループ管理(スレーブ #04)のノードは、自身の管理テーブルに登録された共有資源1と共有資源2にアクセス可能であるが、グループ管理(スレーブ #05)とグループ管理(スレーブ #06)のノードは、自身の管理テーブルに共有資源2の情報が無いため、同共有資源にアクセスすることはできない。使用を希望する場合は、グループ管理(スレーブ #04)と同様の手順を踏んで、自身の管理テーブルに情報を登録することが必要である。

20

#### 【 0 0 2 7 】

続いて、共有資源の返却について説明する。図7はグループXが保持する共有資源2に関して、資源の返却における初期状態を示したものである。グループXが所有する共有資源2の共有数は2であり、グループ管理(スレーブ #04)とグループ管理(スレーブ #05)によって共有されている。この状態でグループ管理(スレーブ #04)のノードは共有資源2の使用を終えたため、その資源の返却要求処理を開始する(図7-(1))。同処理において、まず始めに、自身の管理テーブルに登録された共有資源2に関する情報を削除する(図7-(2))。続いて、グループ管理(スレーブ #04)は、グループ管理(マスタ)に対して共有資源2の使用終了を通知する(図7-(3))。通知を受けたグループ管理(マスタ)は、管理テーブルの共有資源2に対応する共有数を1つ下げる(図7-(4))。共有数変更した後の値は1であり、同資源の利用者がまだ存在するため、共有資源管理に共有資源2を返却せずに保持し続ける。これにより共有資源2に対して、グループ管理(スレーブ #04)のノードからアクセスすることはできないが、グループ管理(スレーブ #05)のノードからアクセスすることは依然として可能である。続いてグループ管理(スレーブ #05)のノードも共有資源2の使用を終え、資源の返却要求処理を開始する(図8-(5))。これまでの手順同様に、自身の管理テーブルから共有資源2に対応する情報を削除し(図8-(6))、グループ管理(マスタ)に対して、共有資源2の返却を通知する(図8-(7))。通知を受けたグループ管理(マスタ)は、管理テーブルから共有資源2に対応する共有数を1つ下げる(図8-(4))。共有数を変更した結果、その値は0であるため、グループ内に共有資源2の利用者がいないことが分かる。よってグループ管理(マスタ)は、共有資源2を、仮定システム1の共有資源管理手段に返却する処理を開始する。まず始めにグループ管理(マスタ)は、自身の管理テーブルから共有資源2に対応する情報を削除する(図8-(5))。続いて、グループ管理(マスタ)は、共有資源管理手段に対して、共有資源2の返却を通知する(図8-(6))。通知を受けた共有資源管理手段は、自身の管理テーブルから共有資源2に対応する情報に対して、状態を「使 未」へ変更し、そ

30

40

50

の他の情報を削除する(図8-(7))。この時点で共有資源2に関する情報は、仮定システム1内のどのノードも保持しておらず、割り当てが解消された状態となる。よって今後の要求に応じて、同資源は再度割り当て可能となる。

【0028】

ここで、以上のような共有資源管理方法においては、次に述べるような問題がある。上述したように、HPC分野では、実行時間短縮のために並列処理が一般的であり、この並列処理は、各プロセスが同じ動作を行なうデータパラレル方式である場合が多いところ、グループが所有する共有資源の使用期間も各プロセスで同等であるため、各プロセスによる共有資源の使用終了もほぼ同じタイミングで発生する。ここで、プロセスによる資源使用終了は、グループ管理(スレーブ)によってグループ管理(マスタ)に通知されるが、データ  
10  
パラレル方式では、時間的空間的にもグループ管理(マスタ)にアクセスが集中してしまう。そこで、全グループ管理(スレーブ)の使用終了報告を待たずに、特定のグループ管理(スレーブ)の使用終了報告をもって、システムに資源返却するという方法も考えられるが、HPC分野では、従来の資源管理方法による返却方式も必要であるため、新たな方式は、従来方式と同時に使用が可能であることが望ましい。換言すれば、1つのプログラム内で従来方式と新方式の混在使用ができれば望ましい。

【0029】

また、多数のノードから構成されるグループ(システム全体を占有するような大規模ジョブ)において、ジョブ実行途中に共有資源を解放する場合、返却要求はネットワークを通じて行なわれるため、システム内の全体的なネットワーク負荷を高めてしまうという問題がある。また、グループ管理(マスタ)は、1箇所にあるため、ネットワークに局所的な高負荷をもたらす(図9参照)。また、これらの要求はグループ管理(マスタ)の管理テーブルを更新することを伴う処理であるところ、当該処理は、管理テーブルの一貫性を保つため並列に処理することはできず、順次処理となるため、グループ規模が大きく、共有数の多い資源の解放は、時間的に大きな処理コストを伴うなどの問題もある。  
20

【0030】

これらの問題を回避する簡易な方法としては、一度保有した共有資源を、ジョブが終了するまで保持し続け、グループの終了処理の際に一緒に資源返却を行なうことが考えられる。一般に仮定システムがもつ共有資源の数は限りがあり、その数も少ないが、仮定システムをジョブが占有している状態では、資源を返却しても、その資源を要求する者がいな  
30  
ため、ジョブの終了まで保持し続けても問題はないからである。よってシステム占有状態の大規模ジョブにおいて、ジョブ終了まで共有資源を保持し続けることは、共有資源の返却コスト削減となる。

【0031】

一方、少数のノードから構成されるグループが、仮定システム内に多数存在する場合には(図2参照)、共有資源の不足が考えられるため、このような小規模ジョブにおいては、ジョブ実行中においても、小まめな共有資源返却が望ましい。共有資源返却コストから見た場合においても、グループを構成するノード数が少ないため、返却に伴うネットワーク使用箇所も局所的であり、グループ管理(マスタ)のノードに対する一極集中も、大規模ジョブに比べて、その度合いはかなり小さくなる。  
40

【0032】

すなわち、仮定システムは、稼働率の面から見て、同じシステムが大規模ジョブと小規模ジョブの2つの種類のジョブに対応する必要があるものの、共有資源管理に関して、その要求は相反する。また、資源返却の効率は、ユーザプログラムにゆだねられているともいえる。例えば自身のプログラムが大規模ジョブであるにも関わらず、小まめな共有資源返却を行なうならば、それはジョブの実行時間は不必要に長くなり、システムを浪費していることになる。逆に自身のジョブが小規模ジョブであるにも関わらず、共有資源を不必要に保持し続けるならば、同時に実行可能なジョブ数が減り、計算資源の有効活用が困難になるかもしれない。このようにユーザが自身のジョブの大小を判断してプログラムを作成することは、望ましい状況ではなく、システムが現状に応じて判断することが好ましい  
50

。

よって、データパラレル方式のプログラム実行時における共有資源の返却処理コストを削減する(すなわち、マスター極集中を緩和する)ことができる共有資源システムを提供することが望まれる。

## 【0033】

また、共有資源返却に関して従来方式と新方式を相互運用することができる共有資源システムを提供することが望まれる。

## 【0034】

また、システムの状態に合わせて共有資源を効果的に運用することができる共有資源システムを提供することが望まれる。

## 【0035】

なお、本発明は、上述のHPC分野で使用されることを想定しており、より具体的には、複数のプロセスからなるプロセス群(グループ)で構成されること、グループ内での共有資源はメモリ・レジスタ等であること、及び、各プロセスの処理がほぼ同じである(資源の返却タイミングが同じ)データパラレル方式であることを前提としている。

## 【0036】

本発明は、上記の特徴を活かして、グループ内の代表プロセスの返却処理のみで資源を返却できる「代表返却方式」を採用する。ただし、この方式を単純に適用すると、図35で説明するような管理不正を発生させる場合がある。そこで、本発明は、この問題を解決する手段として、従来2方式のどちらにも遷移可能な中間状態である「仮返却」(図39参照)という状態を新たに設ける。図39で示すように、「仮返却」状態の資源割り当ては、従来方式の状態に遷移させたのちに行なうので、管理不正の問題は発生しない。また「仮返却」状態は従来方式へ遷移する前の中間状態なので、同じ管理テーブルで従来方式の運用も可能である。この理由によって従来プログラムの実行も可能となっている。すなわち、後述するように、本発明によれば、「仮返却」を行なう新設の資源返却インターフェースを用意することで、1つのグループ管理(スレーブ)の返却処理で資源返却が可能になる。また、1つの共通資源管理テーブルで「従来インターフェース」と「新設インターフェース」の処理の対応が可能であるため、従来方式と新方式で同時使用が可能になる。また、「仮返却」により資源要求が行なわれるまで、実質的な情報削除処理が遅延されるため、システムの稼働状況に合わせた共有資源の割り当て処理が可能になる。

## 【0037】

図37は、本実施形態に係るシステムの概略構成を示す図である。なお、従来技術は、図38に示されている。本発明は、従来方式の返却方法に、「仮返却」状態を生成する処理経路(「処理手段」ともいう)である「仮返却」(「仮返却手段」ともいう)を新設したものであり、新設経路による管理の不整合を解決するために「仮返却資源情報削除要求」505(「仮返却資源情報削除要求手段」ともいう)と「仮返却資源情報削除手段」601の処理経路を追加した構成をとる(図37の斜線経路を参照)。ここで「仮返却」状態とは、図39で示すように、従来方式の2つの返却方法に遷移することができる中間状態を意味する。

## 【0038】

図37で示すように、グループA 600の「仮返却」602の斜線経路によって共有資源管理500(「共有資源管理手段」ともいう)に返された共有資源は、返却要求受付501により受け付けられ、従来方式の状態と異なる「仮返却」という状態で共有資源管理500によって管理される。またグループB 700の「資源要求/情報登録手段」701からの資源要求を割り当て要求受付504が受け付けて、共有資源管理500が、仮返却状態の資源をグループに割り当てる場合、「仮返却資源情報削除要求」505と「仮返却資源情報削除手段」601を経由する斜線経路により、プロセスがもつ「仮返却」状態資源の情報を削除したのちに、割り当てを行なう。これは図39で「プロセス毎の返却」に遷移したことと同等である。また仮返却状態の資源が割り当てられる前にグループが消滅した場合、その資源の状態は未使用資源に移行する。これは図39で「グループ消滅時返却」に遷移したことと同等である。

10

20

30

40

50

## 【 0 0 3 9 】

以下、共有資源を管理テーブルにより管理する場合の詳細な実施例の構成について説明する。本実施例では、共有資源管理の資源の状態(図3の共有資源管理の「状態」として、次の3つを定義する。第1の状態は「使」であり、対象共有資源がグループに割り当てられている状態(使用中)を意味する。第2の状態は「未」であり、対象共有資源がどのグループにも割り当てられていない状態(未使用)を意味する。第3の状態は「仮」であり、対象共有資源はグループからシステムに返却されているが、グループ内のノードが対象共有資源の情報を保持していることを意味する。

## 【 0 0 4 0 】

図10は、仮定システム1、共有資源200、共有資源管理テーブル400、グループ、グループX内における各管理テーブル(100m、100s、101s、102s)の配置について示したものである。図10において、共有資源200はネットワーク300に接続された物理的に孤立した存在として示されているが、これは便宜上の表記である。従って各ノード(100~111)で共有可能なものであれば、その存在はノードの内外を問わない。また共有資源管理テーブル400も同様に示されているが、これも「共有資源管理」がアクセス可能な位置にあれば、ノードの内外を問わない。

10

## 【 0 0 4 1 】

「共有資源」とは、ある機能を有する資源が複数集まったものである。例えばネットワーク上に存在し、各ノードよりアクセス可能なレジスタが複数存在する等である。

## 【 0 0 4 2 】

「共有資源管理」(「共有資源管理手段」ともいう)とは、上記共有資源の個々を管理する機能を有する。各グループ管理(マスタ)より「共有資源の割り当て要求」、「共有資源の返却」の通知を受け、実際の処理を行なう。「共有資源情報削除要求」をグループ管理(マスタ)に要求する役割も担う。

20

## 【 0 0 4 3 】

「グループ管理(マスタ)」(「グループ管理手段(マスタ)」ともいう)とは、グループ内の「グループ管理(スレーブ)」を管理する機能を有する。グループに参加しているノードの番号、グループが保有している共有資源の番号等を管理する。グループ管理(マスタ)のみが、仮定システムに対して「共有資源の割り当て要求」と「共有資源の返却」を行なうことができる。また仮定システム1より「共有資源情報削除要求」を受け、処理の完了を仮定システムに通知する役割もグループ管理(マスタ)が行なう。

30

## 【 0 0 4 4 】

「グループ管理(スレーブ)」(「グループ管理手段(スレーブ)」ともいう)とは、ノード内において、グループに属するプロセスを管理する機能を有する。またそのプロセスがアクセス可能な共有資源情報も管理する。グループ管理(スレーブ)がグループ管理(マスタ)に「資源共有」を要求することによって、グループ管理(スレーブ)に共有資源の情報が登録され、同資源にアクセスが可能となる。よって、プロセスはグループ管理(スレーブ)に登録された共有資源にのみアクセスが可能であり、グループ管理(スレーブ)はグループ管理(マスタ)が保有する共有資源の情報のみ登録することが可能であるため、プロセスは、グループが保有する共有資源以外の共有資源にアクセスすることはできない。またグループ管理(スレーブ)は、直接、「共有資源管理」に問合せすることはできない。グループ管理(スレーブ)がグループ管理(マスタ)に要求し、グループ管理(マスタ)が共有資源管理に問合せする形態になる。ここでグループ管理(スレーブ)はプロセスを管理するものであるが、本実施例では、説明の便宜上、グループ管理(スレーブ)を末端として説明し、その管理を管理テーブルにおいて省略している。

40

## 【 0 0 4 5 】

図11は、各管理テーブルの内容の一例を示したものである。各列の意味は、以下の通りである。

## 【 0 0 4 6 】

まず、共有資源管理の管理テーブルについて説明する。「資源番号」とは、管理対象の

50

共有資源に振られた一意な番号である。「状態」とは、共有資源の割り当て状況である。「使」はあるグループに資源を割り当てている状態である。「未」はどのグループにも資源を割り当てておらず、すぐに割り当て可能な状態である。「仮」は資源割り当て可能であるが、割り当て前に「共有資源情報削除処理」を必要とする状態である。「情報削除処理コスト」は、共有資源情報削除処理を行なう場合の処理コスト値である。「グループ管理ノード」は、そのグループを管理する「グループ管理(マスタ)」が存在するノード番号である。「グループID: 仮定システム内でグループを一意に識別するためのIDである。

#### 【0047】

ここで「情報削除処理コスト」はとは、「仮」状態の資源が複数あった場合、どれか1つを選択するための指標であり、値が小さいほど割り当てるときの処理コストが小さくなるような評価指標とする。以下の指標の例を示す。

情報削除処理コスト = a + b

a = \*グループに参加しているノード数

b = \*資源の共有数

aはグループ管理(マスタ)がグループ内の各グループ管理(スレーブ)に削除要求を行なうときの通信コストであり、bは各グループ管理(スレーブ)から削除完了を行なうときの通信コストである。またa、bにおける と は重み係数であり、実行システムの状態に合わせて適切に設定される値である。実行システム依存であるため、本実施例はこの値を特定しない。この重みの決め方の一例として、パケット数などがある。グループ管理(マスタ)からグループ管理(スレーブ)に送信される情報削除要求にかかるパケット数を3、グループ管理(スレーブ)からグループ管理(マスタ)に送信される完了通知にかかるパケット数を1とするならば、 =3、 =1とすることが考えられる。以下、本実施例では、説明の便宜上、 =1、 =1として動作説明を行なうが、値はこれらに特定されない。

#### 【0048】

次に、グループ管理(マスタ)の管理テーブルについて説明する。「グループID」は、仮定システム内でグループを一意に識別するためのIDである。「参加ノード情報」は、そのグループに参加しているノードの番号である。各ノードは仮定システム内で一意な値が割り振られている。「保有共有資源情報」は、その分子はそのグループが保有する(仮定システムによって割り当てられた)共有資源番号であり、その分母はグループ内でその資源情報をもつグループ管理(スレーブ)の数である。

#### 【0049】

次に、グループ管理(スレーブ)の管理テーブルについて説明する。「グループID」は、仮定システム内でグループを一意に識別するためのIDである。「マスタノード」は、グループ管理(スレーブ)が属するグループにおいて、グループ管理(マスタ)がいるノード番号である。「共有資源番号」は、グループ管理(スレーブ)のノードからアクセス可能な共有資源の番号である。グループ管理(マスタ)に情報が登録されていても、グループ管理(スレーブ)に情報がない場合、プロセスは共有資源にアクセスできない。

#### 【0050】

図27に本実施例の管理構成を示す。まず、グループ管理(スレーブ)について説明する。「共有要求」(「共有要求手段」ともいう)は、グループが保持する共有資源の使用をマスタに要求する手段である。「資源登録削除」(「資源登録削除手段」ともいう)は、共有資源の情報を自身の管理テーブルに登録・削除を行なう手段である。「使用終了通知」(「使用終了通知手段」ともいう)は、自身が指定共有資源の終了をマスタに通知する手段(従来方式と発明方式の2手段あり)である。「情報削除通知」(「情報削除通知手段」ともいう)は、自身の管理テーブルから指定された共有資源の情報を削除したことをマスタに通知する手段である。

#### 【0051】

次に、グループ管理(マスタ)について説明する。「資源要求」(「資源要求手段」ともいう)は、グループの代表として共有資源管理に共有資源の割り当てを要求する手段である。「資源登録削除」(「資源登録削除手段」ともいう)は、共有資源の情報を自身の管

10

20

30

40

50

理テーブルに登録・削除を行なう手段である。「テーブル更新」(「テーブル更新手段」ともいう)は、自身の管理テーブルの情報を更新する手段である。「資源返却」(「資源返却手段」ともいう)は、指定した共有資源を共有資源管理テーブルに返却する手段(従来方式と発明方式の2手段あり)である。「情報削除要求」(「情報削除要求手段」ともいう)は、指定した共有資源の情報削除をスレーブに要求する手段である。「情報削除通知」(「情報削除通知手段」ともいう)は、グループ内で指定された共有資源の情報削除が完了したことを共有資源管理に通知する手段である。

**【0052】**

次に、共有資源管理について説明する。「資源探索」(「資源探索手段」ともいう)は、自身の管理テーブルから未使用資源、仮解放資源を探索する手段である。「テーブル更新」(「テーブル更新手段」ともいう)は、指定共有資源の管理情報を更新する手段である。「情報削除要求」(「情報削除要求手段」ともいう)は、指定した共有資源の情報削除をグループ管理(マスタ)に要求する手段である。

10

**【0053】**

次に、以上のように構成される仮想システム1の動作について説明する。

**【0054】**

まず、「1.共有資源の要求(未使用資源がある場合)」の動作について説明する。図12はグループRが共有資源の割り当てを要求した場合の処理手順を示したものである。対応フローは図28である。この条件では、まだ割り当てられていない未使用資源がある場合を仮定している。これは共有資源管理テーブルにおいて、状態が「未」である資源が存在することを意味し、該当する共有資源は資源番号3である。このときノード番号55であるグループRのグループ管理(マスタ)が共有資源の割り当てを要求する場合の手順を以下に示す。

20

**【0055】**

グループ管理(マスタ)は共有資源を得るために、共有資源管理に「共有資源割り当て要求」を送信する(図12-(1))(図28のS1)。要求を受け取った共有資源管理は自身が管理する「共有資源管理テーブル」から、状態が「未」である資源をさがす(図12-(2))(図28のS2)。探索の結果、資源番号3が該当することを見つけ(図28のS2の「有」)、割り当てを行なうために管理情報である状態を「使」、グループ管理ノードを「#55」、グループIDを「R」に更新する(図12-(3))(図28のS4)。その後、情報を更新した資源番号3を、共有使資源の割り当て要求を行なったグループ管理(マスタ)に通知する(図12-(4))(図28のS5)。共有資源管理より資源番号を通知されたグループ管理(マスタ)は、自身が管理する「グループ管理(マスタ)テーブル」に、通知された資源番号3とその共有数0を登録する(図12-(5))(図28のS6)。以上の手順を行なった後の各テーブルの状態を図13に示す。

30

**【0056】**

次に、「2.グループ内における共有資源の共有」に関する動作について説明する。図14はグループが保有する共有資源を、グループ内のグループ管理(スレーブ)により共有する場合の処理手順を示したものである。対応フローは図29である。この条件において、ノード番号55であるグループ管理(マスタ #55)は保有共有資源として資源番号3の共有資源をもつ。このときノード番号57であるグループ管理(スレーブ #57)が、この共有資源の共有を要求する場合の処理手順を以下に示す。

40

**【0057】**

グループ管理(スレーブ)はグループが保有する資源の共有をグループ管理(マスタ)に要求する(図14-(1))(図29のS11)。要求を受けたグループ管理(マスタ)は資源番号3に対応する保有共有資源情報の「共有数」を1つ増やす(図14-(2))(図29のS12)。更新後、グループ管理(マスタ)は、共有を要求したグループ管理(スレーブ)に共有資源番号3を通知する(図14-(3))(図29のS13)。通知を受けたグループ管理(スレーブ)は、自身が管理する「グループ管理(スレーブ)テーブル」に共有資源番号3を登録する(図14-(4))(図29のS14)。以上の手順を行なった後の各テーブルの状態を図15に示す。

50

## 【 0 0 5 8 】

次に、「3. 共有資源の要求(未使用資源がない場合)」に関する動作について説明する。図16はグループXが共有資源の割り当てを要求した場合の処理手順を示したものである。対応フローは図30である。この条件では、共有資源管理テーブル上に未使用資源はなく、割り当て可能な資源として「仮返却」された資源のみがある場合を仮定している。これに該当する資源は共有資源管理テーブルにおいて、状態が「仮」である資源番号2と4である。このときノード番号00であるグループ管理(マスタ)が共有資源の割り当てを要求する場合の手順を以下に示す。

## 【 0 0 5 9 】

グループ管理(マスタ)は共有資源を得るために、共有資源管理に「共有資源割り当て要求」を送信する(図16-(1))(図30のS 2 1)。要求を受け取った共有資源管理は自身が管理する「共有資源テーブル」から、状態が「未」である資源をさがす(図16-(2))(図30のS 2 2)。探索の結果、「未」の資源が見つからなかった(図30のS 2 3の「無」)、続いて「仮」の資源をさがす(図16-(3))(図30のS 2 4)。探索の結果、資源番号2と4が見つかる(図16-(4))(図30のS 2 5の「有」)。資源番号2と4の「情報削除処理コスト」値を比較して、最も小さな値をもつ資源番号4を割り当て候補として選択する(図16-(5))。共有資源管理は資源番号4を仮返却したグループZのグループ管理(マスタ)に対して、資源番号4の「共有資源情報削除」を要求する(図17-(6))(図30のS 2 6)。要求を受け取ったグループZのグループ管理(マスタ)は、自身が管理する管理テーブルの「参加ノード情報」をもとに、グループの全グループ管理(スレーブ)に対して「共有資源番号4の情報削除」を要求する(図18-(7))(図30のS 2 7)。グループ管理(マスタ)は情報の削除要求を送信した後、各グループ管理(スレーブ)からの削除完了通知を待つ。要求を受け取ったグループZの各グループ管理(スレーブ)は、自身が管理する管理テーブルから共有資源番号4の情報を削除する(図18-(8))(図30のS 2 8)。情報を削除したグループZの各グループ管理(スレーブ)は処理の完了通知をグループZのグループ管理(マスタ)に通知する(図30のS 2 9)。ここで削除要求対象である資源番号4の情報をもたないグループ管理(スレーブ)は何もしない。グループZのグループ管理(マスタ)が待つ各グループ管理(スレーブ)からの完了通知数は、情報削除対象である資源の「共有数」である。この数の完了通知を受け取ることで、グループ内において、対象資源の情報をもつグループ管理(スレーブ)の存在しないことが保証される(図19-(10))。よってこの時点で情報削除対象の資源に関する情報は、グループZのグループ管理(マスタ)のみが保持しており、自身が管理するテーブルから情報を削除することで、仮定システム内において、対象資源の情報をもつノードは存在しないことが保証される(図19-(11))(図30のS 3 0)。処理が完了したグループZのグループ管理(マスタ)は「情報削除完了」を共有資源管理に通知する(図20-(12))(図30のS 3 1)。通知を受けた共有資源管理は、資源番号4の情報を未使用状態に変更する(図20-(13))。この時点で資源番号4は未使用となったため、割り当てが可能となる。よって資源番号4をグループXに割り当てるため、管理情報を使用状態に更新する(図21-(14))(図30のS 3 2)。更新後、グループXのグループ管理(マスタ)に、割り当て共有資源番号4を通知する(図21-(15))(図30のS 3 3)。通知を受けたグループXのグループ管理(マスタ)は、自身が管理する管理テーブルの保有共有資源情報に「4/0」を登録する(図21-(16))(図30のS 3 4)。以上の手順を行なった後の各テーブルの状態を図22に示す。

## 【 0 0 6 0 】

次に、「4. 共有資源の仮返却(スレーブが仮返却する場合)」に関する動作について説明する。図23はグループ管理(スレーブ)が共有している共有資源をグループ管理(マスタ)に仮返却する場合の処理手順を示したものである。対応フローは図31である。この条件では、返却予定の資源に、返却者以外の共有者がいる。よってここでは、グループ管理(マスタ)が共有資源管理に対して行なう返却が「仮返却」となる場合を仮定している。このときグループXのノード番号#02のグループ管理(スレーブ)が、自身が共有する共有資源番号1の資源返却を要求する場合の手順を以下に示す。

## 【 0 0 6 1 】

10

20

30

40

50

グループ管理(スレーブ)は自身が共有する共有資源番号1の資源仮返却を希望する。まず始めに自身の管理テーブルから共有資源番号1の情報を削除する(図23-(1))(図31のS61)。返却を希望するグループ管理(スレーブ)は、グループ管理(マスタ)へ、共有資源番号1の仮返却を要求する(図23-(2))(図31のS62)。要求を受けたグループ管理(マスタ)は、共有資源番号1に対応する「保有共有資源情報」の「共有数」を1つ下げ、「仮」の印をつける(図23-(3))(図31のS63)。その後、グループ管理(マスタ)は共有資源管理へ共有資源1の仮返却を通知する(図31のS64、S65)。このとき「情報削除処理コスト」値も通知するが、説明の便宜上、 $\text{cost} = 1$ 、 $\text{cost} = 1$ としているので、値は5となる(図23-(4))。通知を受けた共有資源管理は資源番号1の状態を「仮」、情報削除処理コストを「5」に更新する(図31のS66)。以上の手順を行なった後の各テーブルの状態を図24に示す。

10

#### 【0062】

次に、「5. 共有資源の返却(マスタが返却する場合)」に関する動作について説明する。図25はグループ管理(マスタ)が保有する共有資源を共有資源管理に通常返却する場合の処理手順を示したものである。対応フローは図31である。グループ管理(マスタ)が共有資源を返却する場合は共有数が1から0になった場合とグループが消滅する場合である。どちらもグループ管理(スレーブ)が発端となるが、グループ管理(マスタ)から見ると、どちらも共有数が0になったときに返却することには変わりはないので、説明手順は共有数が0になったところから始め、発端のグループ管理(スレーブ)の動作は省略する。同図ではグループXが消滅する場合の資源返却を仮定したもので、グループ管理(マスタ)が保有する資源番号1と4の返却が必要となる。しかし各資源の返却手順は同じであるため、資源番号1の返却手順のみを以下に示す。

20

#### 【0063】

グループ管理(マスタ)はグループ消滅に伴い、保有共有資源の返却処理を開始する。保有共有資源は2つあるが、1つずつ返却されるため、共有資源番号1に着目する。グループが消滅する場合、参加ノードはもはやいない。よって資源の共有数は0である(図31のS48の「0」)。グループ管理(マスタ)は自身の管理テーブルから共有資源1に関する情報を削除する(図25-(1))。これからグループが消滅し、テーブルが放棄されるので情報削除処理は必要ないが、グループ消滅以外でも資源返却は可能であり、その手順との共通性を意識して、この処理を行なう。削除処理を行なった後、グループ管理(マスタ)は、共有資源管理へ共有資源番号1の返却を通知する(図25-(2))(図31のS49)。通知を受けた共有資源管理は、資源番号1の管理情報を未使用として更新する(図25-(3))(図31のS50)。以上の手順を共有資源番号4に対しても行なう。以上の手順を行なった後の各テーブルの状態を図26に示す。

30

#### 【0064】

次に、従来管理法との共存について説明する。本発明は、従来返却法(従来法ともいう)としても動作可能である。例えば、共有資源返却用の手続き手段(インターフェース)を以下のように2つ用意する。

1. freeOld()
2. freeNew()

40

freeOld()は従来法のための手続き手段であり、freeNew()は発明法のためのインターフェースとする。freeNew()を使用する場合は、これまで説明した手順で管理を実行する。従来法のfreeOld()を使用する場合は、単に従来通りに共有数が0になった段階で共有資源管理に資源を返却すればよい。単に本発明において「仮返却」という中間状態を除いた管理を行なえばよいだけであり、特別な追加処理は必要ない。

#### 【0065】

よって本発明を適用したシステムにおいても、プログラムの変更なしに、従来法のインターフェースによって書かれたプログラムの実行が可能である。また従来プログラムに対して、従来方式のインターフェースで新方式の処理を適用することも可能である。例えば従来方式のライブラリと新方式のライブラリを用意し、これら

50

をダイナミックリンクライブラリとする。実行時にユーザが環境変数等で指定することでリンクするライブラリを変更する方法で、従来方式のインターフェースにおいても新方式の処理が可能となる。

【0066】

HPCのプログラムはデータパラレルによりプロセスごとに同じ処理を行なうと仮定しているが、ネットワーク遅延やCPU数以上のプロセス実行によるプロセス切り替え、割り込み等が原因で処理タイミングがずれる場合が考えられる。よって従来方式で書かれたプログラムに対してそのまま新方式を適用すると、実行タイミングのズレにより資源解放されているにも関わらず資源を使用しているプロセスが発生する場合が考えられる。この場合の対策として、以下の4つが考えられる。

10

【0067】

第1の対策として、情報削除要求が来た場合、使用状況に関わらず強制的に情報を削除するようにしてもよい(資源を使用していたジョブはアポートになる)。

【0068】

第2の対策として、各プロセスが共有資源の使用状態をノード内で管理し、情報削除要求が来た場合、まだ使用していることを通知することで要求を拒否するようにしてもよい。拒否された共有資源管理は本「仮返却」資源の割り当てをあきらめ、次ぎの候補の「仮返却」資源に対して「情報削除要求」を行なう。

【0069】

第3の対策として、各プロセスが共有資源の使用状態をノード内で管理し、情報削除要求が来た場合、使用が終了するまで返答を保留するようにしてもよい。もしくは要求を拒否し、拒否された共有資源管理は、一定時間を置いてリトライするようにしてもよい。これはどちらの場合においても、共有資源管理が「仮返却」資源の情報が削除されるまで待つことを意味する。

20

【0070】

第4の対策として、仮資源返却の場合、管理テーブルの「状態」をすぐに「仮」にせず、想定される実行タイミングのズレより大きい時間、状態の変更を保留した後に「仮」に変更するようにしてもよい。

【0071】

上記本実施形態によれば、資源運用の効率化と資源返却処理コスト削減を両立することができるようになる。

30

【0072】

また、上記本実施形態によれば、「プロセス毎の返却」方法で全ての資源共有者が同時に資源返却処理を行なう場合、グループ管理(マスタ)にアクセスが集中することで、ネットワークに局所的な高負荷が発生したが、これを緩和することができる事が挙げられる。すなわち、グループの代表として1プロセスによる返却処理だけでシステムに資源を返却できるようにしたためである。

【0073】

また、上記本実施形態によれば、「プロセス毎の返却」方法において管理テーブルの整合性を保証するための排他アクセスが原因で、ノード数の多い大規模ジョブによる資源返却では非常に大きな時間的返却コストを必要としたが、これを緩和できる事が挙げられる。すなわち、ジョブの大小にかかわらずグループの代表として1プロセスによる返却処理だけでシステムに資源を返却できるようにしたためである。

40

【0074】

また、上記本実施形態によれば、未使用資源枯渇時に資源要求が発生した場合は資源返却の処理コストを支払うことで未使用資源を確保してジョブの並行実行を優先するが、未使用資源がある場合は資源の返却コスト削減を優先する動作を自動で行なうことができる事が挙げられる。すなわち、「仮返却」状態の資源を、未使用資源が無くなった時にはじめて割り当て対象とする管理手法のためである。

【0075】

50

また、上記本実施形態によれば、管理テーブルを分けることなく「仮返却」が従来との返却法と共存できるため、従来法の動作を期待するプログラムの実行も可能なことが挙げられる。すなわち、「仮返却」が従来法の状態に遷移可能な中間状態という位置づけのためである。

【0076】

なお、本明細書等において、手段とは、単に物理的手段を意味するものではなく、その手段が有する機能をソフトウェアによって実現する場合も含む。また、1つの手段が有する機能が2つ以上の物理的手段により実現されても、2つ以上の手段の機能が1つの物理的手段により実現されてもよい。

【0077】

また、本発明は、上記した実施の形態に限定されるものではなく、本発明の要旨を逸脱しない範囲内において、他の様々な形で実施することができる。このため、上記実施形態はあらゆる点で単なる例示にすぎず、限定的に解釈されるものではない。例えば、上述の各処理ステップは処理内容に矛盾を生じない範囲で任意に順番を変更して又は並列に実行することができる。

【0078】

なお、上記実施形態の一部又は全部は、以下の付記のようにも記載されうるが、以下には限らない。

【0079】

(付記1) 複数の資源からなる共有資源の割り当てと当該割り当てた共有資源の返却を管理する共有資源管理手段と、  
複数のプロセスを有するプロセスグループにおける前記共有資源の割り当てと返却を管理する管理マスタと、を備え、  
前記共有資源管理手段は、  
前記管理マスタから当該プロセスグループに割り当てられた資源の仮返却を受け付ける手段と、  
他のプロセスグループを管理する他の管理マスタから前記共有資源の取得要求を受け付ける手段と、  
前記共有資源中に未使用の資源があるか否かを判定し、未使用の資源がないと判定した場合は、前記仮返却された資源を前記他のプロセスグループに割り当てる割り当て手段と、  
を備えることを特徴とする共有資源管理システム。

【0080】

(付記2) 前記管理マスタは、  
当該管理マスタが対象とするプロセスグループ中の少なくとも1つのプロセスが当該プロセスグループに割り当てられた資源を返却可能な状態になった場合に、当該プロセスグループに割り当てられた資源を前記共有資源管理手段へ仮返却することを特徴とする付記1に記載の共有資源管理システム。

【0081】

(付記3) 前記共有資源管理手段は、  
前記仮返却された資源を前記他のプロセスグループに割り当てる場合は、当該仮返却された資源の資源情報の削除を当該仮返却した管理マスタに要求することを特徴とする付記1または2に記載の共有資源管理システム。

【0082】

(付記4) 前記共有資源管理手段は、  
前記未使用の資源がないと判定した場合において前記仮返却された資源が複数あるときは、当該複数の仮返却された資源について算定された各情報削除処理コストに基づいて、前記他のプロセスグループに割り当てる資源を選択することを特徴とする付記1乃至3いずれか1項に記載の共有資源管理システム。

【0083】

(付記5) 前記管理マスタは、

10

20

30

40

50

当該管理マスタが対象とするプロセスグループ中の全てのプロセスが当該プロセスグループに割り当てられた資源を返却可能な状態になった場合に、当該プロセスグループに割り当てられた資源を前記共有資源管理手段へ返却することを特徴とする付記 1 乃至 4 いずれか 1 項に記載の共有資源管理システム。

【 0 0 8 4 】

(付記 6) 前記管理マスタは、

当該管理マスタが対象とするプロセスグループが消滅した場合に、当該プロセスグループに割り当てられた資源を前記共有資源管理手段へ返却することを特徴とする付記 1 乃至 4 いずれか 1 項に記載の共有資源管理システム。

【 0 0 8 5 】

(付記 7) 複数の資源からなる共有資源の割り当てと当該割り当てた共有資源の返却を管理する共有資源管理手段と、複数のプロセスを有するプロセスグループにおける前記共有資源の割り当てと返却を管理する管理マスタと、を備えるシステムにおける共有資源管理方法であって、

前記共有資源管理手段が、

前記管理マスタから当該プロセスグループに割り当てられた資源の仮返却を受け付けるステップと、

他のプロセスグループを管理する他の管理マスタから前記共有資源の取得要求を受け付けるステップと、

前記共有資源中に未使用の資源があるか否かを判定し、未使用の資源がないと判定した場合は、前記仮返却された資源を前記他のプロセスグループに割り当てる割当ステップと、を備えることを特徴とする共有資源管理方法。

【 0 0 8 6 】

(付記 8) 前記管理マスタは、

当該管理マスタが対象とするプロセスグループ中の少なくとも 1 つのプロセスが当該プロセスグループに割り当てられた資源を返却可能な状態になった場合に、当該プロセスグループに割り当てられた資源を前記共有資源管理手段へ仮返却するステップを備えることを特徴とする付記 7 に記載の共有資源管理方法。

【 0 0 8 7 】

(付記 9) 前記共有資源管理手段は、

前記仮返却された資源を前記他のプロセスグループに割り当てる場合は、当該仮返却された資源の資源情報の削除を当該仮返却した管理マスタに要求するステップを備えることを特徴とする付記 7 または 8 に記載の共有資源管理方法。

【 0 0 8 8 】

(付記 10) 前記共有資源管理手段は、

前記未使用の資源がないと判定した場合において前記仮返却された資源が複数あるときは、当該複数の仮返却された資源について算定された各情報削除処理コストに基づいて、前記他のプロセスグループに割り当てる資源を選択するステップを備えることを特徴とする付記 7 乃至 9 いずれか 1 項に記載の共有資源管理方法。

【 0 0 8 9 】

(付記 11) 前記管理マスタは、

当該管理マスタが対象とするプロセスグループ中の全てのプロセスが当該プロセスグループに割り当てられた資源を返却可能な状態になった場合に、当該プロセスグループに割り当てられた資源を前記共有資源管理手段へ返却するステップを備えることを特徴とする付記 7 乃至 10 いずれか 1 項に記載の共有資源管理方法。

【 0 0 9 0 】

(付記 12) 前記管理マスタは、

当該管理マスタが対象とするプロセスグループが消滅した場合に、当該プロセスグループに割り当てられた資源を前記共有資源管理手段へ返却するステップを備えることを特徴とする付記 7 乃至 10 いずれか 1 項に記載の共有資源管理方法。

10

20

30

40

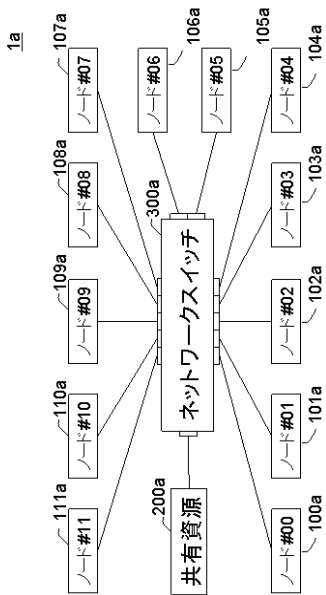
50

【符号の説明】

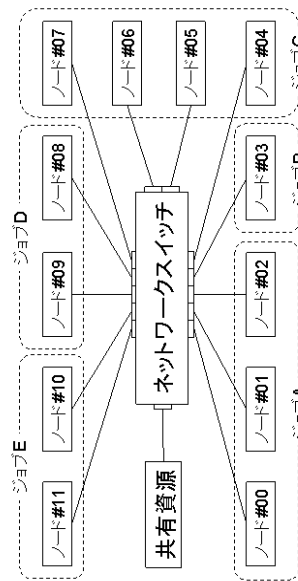
【0091】

1 ... 仮定システム、100 ... ノード#00、101 ... ノード#01、102 ... ノード#02、103 ... ノード#03、104 ... ノード#04、105 ... ノード#05、106 ... ノード#06、107 ... ノード#07、108 ... ノード#08、109 ... ノード#09、110 ... ノード#10、111 ... ノード#11、200 ... 共有資源、300 ... ネットワークスイッチ、400 ... 共有資源管理テーブル

【図1】



【図2】







【 図 1 1 】

共有資源管理

共有資源管理

資源番号	状態	情報削除処理コスト	グループ管理ノード	グループID
1	使	----	#00	X
2	仮	12	#24	S
3	未	----	----	----
4	仮	7	#05	Z

グループ管理 (スレーブ)

グループID	状態
X	X
#00	#00
共有資源番号	1

グループ管理 (マスター)

グループID	状態
X	X
#00, #01, #02	#00, #01, #02
共有資源番号 (番号 / 共有数)	1 / 3

保持共有資源の類に応じて併合する

保持共有資源の類に応じて併合する

【 図 1 3 】

共有資源管理

共有資源管理

資源番号	状態	情報削除処理コスト	グループ管理ノード	グループID
1	使	----	#00	X
2	仮	12	#24	S
3	使	----	#55	R
4	仮	7	#05	Z

グループ管理 (マスター #55)

グループID	状態
R	R
#55, #56, #57	#55, #56, #57
共有資源番号 (番号 / 共有数)	3 / 0

【 図 1 2 】

共有資源管理

共有資源管理

資源番号	状態	情報削除処理コスト	グループ管理ノード	グループID
1	使	----	#00	X
2	仮	12	#24	S
3	未	----	----	----
4	仮	7	#05	Z

未使用資源の廃棄 (状態未の資源を廃棄) ①追加された共有資源番号 (未, ..., #55, #56) に更新

グループ管理 (マスター #55)

グループID	状態
R	R
#55, #56, #57	#55, #56, #57
共有資源番号 (番号 / 共有数)	----

①共有資源番号に応じて要求

②併合して資源番号として「3」を返却する。

③併合して資源番号として「3」を返却する。

【 図 1 4 】

共有資源管理

共有資源管理

資源番号	状態	情報削除処理コスト	グループ管理ノード	グループID
1	使	----	#00	X
2	仮	12	#24	S
3	使	----	#55	R
4	仮	7	#05	Z

グループ管理 (マスター #55)

グループID	状態
R	R
#55, #56, #57	#55, #56, #57
共有資源番号 (番号 / 共有数)	3 / 0

グループ管理 (スレーブ #57)

グループID	状態
R	R
#55	#55
共有資源番号	----

①資源の共有を要求

②通知された共有資源番号を登録 (0→1)

③通知された共有資源番号を登録 (0→3)

【 図 1 5 】

「使」グループに割り当てられている共有資源割り当て(以下、「使」割り当て)で可能な共有資源 (使)割り当て可能な共有資源 (使)割り当て可能な共有資源

共有資源管理

資源番号	状態	情報削除処理コスト	グループ管理ノード	グループID
1	使	---	#00	X
2	仮	12	#24	S
3	使	---	#55	R
4	仮	7	#05	Z

グループ管理 (スレーブ #57)

グループID	R
マスターノード	#55
共有資源番号	3

グループ管理 (マスター #56)

グループID	R
参加ノード情報	#55, #56, #57
保有共有資源情報 (番号 / 共有数)	3 / 1

【 図 1 7 】

「使」グループに割り当てられている共有資源割り当て(以下、「使」割り当て)で可能な共有資源 (使)割り当て可能な共有資源 (使)割り当て可能な共有資源

共有資源管理

資源番号	状態	情報削除処理コスト	グループ管理ノード	グループID
1	使	---	#00	X
2	仮	12	#24	S
3	使	---	#55	R
4	仮	7	#05	Z

◎各資源番号の情報は保持グループのマスターノード#05に共有資源情報検索要求する。

グループ管理 (マスター #05)

グループID	Z
参加ノード情報	#04, #05, #06, #07
保有共有資源情報 (番号 / 共有数)	4 / 3 (仮)

【 図 1 6 】

「使」グループに割り当てられている共有資源割り当て(以下、「使」割り当て)で可能な共有資源 (使)割り当て可能な共有資源 (使)割り当て可能な共有資源

共有資源管理

資源番号	状態	情報削除処理コスト	グループ管理ノード	グループID
1	使	---	#00	X
2	仮	12	#24	S
3	使	---	#55	R
4	仮	7	#05	Z

- ◎未使用資源の探索 (状態「未」の資源を探索)
- ◎仮状態の資源を探索 (状態「仮」の資源を探索)
- ◎資源番号と4で、2つ候補が見つかる。
- ◎資源番号と4の「情報削除処理コスト」値を比較。コストの小さい資源番号が割り当て候補となる。

◎共有資源割り当て要求

グループ管理 (マスター #00)

グループID	X
参加ノード情報	#00, #01, #02
保有共有資源情報 (番号 / 共有数)	1 / 3

【 図 1 9 】

◎削除が完了したことをグループ管理(マスター)に通知する(通知しない)は通知の通知しない)

グループ管理 (スレーブ #04)

グループID	Z
参加ノード情報	#04, #05, #06, #07
保有共有資源情報 (番号 / 共有数)	4 / 3 (仮)

- ◎共有数の分だけ削除完了の通知を送る
- ◎共有数の削減完了通知を受けたら、保有共有資源情報を削除する。(4/3 → 1/3)

グループ管理 (スレーブ #04)

グループID	Z
マスターノード	#05
共有資源番号	---

グループ管理 (スレーブ #06)

グループID	Z
マスターノード	#05
共有資源番号	---

◎削除が完了したことをグループ管理(マスター)に通知する(通知しない)は通知の通知しない)

グループ管理 (スレーブ #05)

グループID	Z
マスターノード	#05
共有資源番号	---

グループ管理 (スレーブ #07)

グループID	Z
マスターノード	#05
共有資源番号	---

もどから情報交換していないので、グループ管理(マスター)に削除完了の通知は行わない。

【 図 2 0 】

共有資源管理

【注】グループに割り当てられている共有資源の割り当て状況  
【注】割り当て可能な共有資源  
【注】割り当て可能な共有資源

資源番号	状態	情報削除処理コスト	グループ管理ノード	グループID
1	使	----	#00	X
2	仮	12	#24	S
3	使	----	#55	R
4	仮	7	#05	Z

①削除完了通知受信。  
資源番号4の権限表を使用し  
(※、.....、.....)

②共有資源4の権限表を削除する

グループ管理 (マスタ #05)

グループID	参加ノード情報 (番号 / 共有数)
Z	#04, #05, #06, #07
----	----

【 図 2 3 】

共有資源管理

【注】グループに割り当てられている共有資源の割り当て状況  
【注】割り当て可能な共有資源  
【注】割り当て可能な共有資源

資源番号	状態	情報削除処理コスト	グループ管理ノード	グループID
1	使	----	#00	X
2	仮	12	#24	S
3	使	----	#65	R
4	使	----	#00	X

⑤資源番号1の情報を更新する。  
(仮、5, #00, X)

③共有資源管理に共有資源1の  
仮返却を通知する。  
また情報削除処理コスト5も通知する。

④共有資源1の仮返却を通知

グループ管理 (マスタ #00)

グループID	参加ノード情報 (番号 / 共有数)
X	#00, #01, #02
1 / 3	1 / 3
4 / 0	4 / 0

⑥共有資源10引き、  
仮の印を付ける  
(13 → 12(仮))

⑦共有資源1の仮返却を要する  
ので、同資源の権限表を削除する。  
(1 → .....)

【 図 2 4 】

共有資源管理

【注】グループに割り当てられている共有資源の割り当て状況  
【注】割り当て可能な共有資源  
【注】割り当て可能な共有資源

資源番号	状態	情報削除処理コスト	グループ管理ノード	グループID
1	仮	5	#00	X
2	仮	12	#24	S
3	使	----	#55	R
4	使	----	#00	X

グループ管理 (マスタ #00)

グループID	参加ノード情報 (番号 / 共有数)
X	#00, #01, #02
1 / 2 (仮)	1 / 2 (仮)
4 / 0	4 / 0

グループ管理 (スレーブ #02)

グループID	マスタノード	共有資源番号
R	#00	----

【 図 2 5 】

共有資源管理

【注】グループに割り当てられている共有資源の割り当て状況  
【注】割り当て可能な共有資源  
【注】割り当て可能な共有資源

資源番号	状態	情報削除処理コスト	グループ管理ノード	グループID
1	仮	5	#00	X
2	仮	12	#24	S
3	使	----	#55	R
4	使	----	#00	X

⑧追加通知を受けた共有資源10の  
管理情報を更新する。  
(※、.....、.....)

③共有資源10の返却を通知する。

グループ管理 (マスタ #00)

グループID	参加ノード情報 (番号 / 共有数)
X	----
1 / 0 (仮)	1 / 0 (仮)
4 / 0	4 / 0

⑨グループの消滅に伴って共有資源の返却の  
ための権限表を削除する。この時点で、  
グループ管理(マスタ)に追加している  
グループが保有する資源の情報はこの  
グループが保有する資源の情報とは異なる  
のめがつつ、よって共有資源も0である

④共有資源番号10に関しても、  
同様の処理を行う。

【 図 2 6 】

**共有資源管理**

(ただしグループIDが重複している共有資源は除外して可)  
(未: 未利用、仮: 利用中、使: 使用済み、未: 未利用)

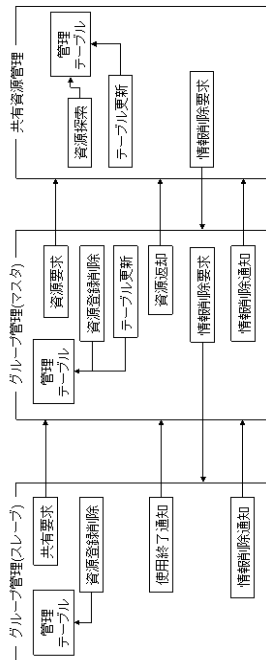
資源番号	状態	情報削除処理コスト	グループ管理ノード	グループID
1	未	.....	.....	.....
2	仮	12	#24	S
3	使	.....	#55	R
4	未	.....	.....	.....

グループ管理 (マスタ #00)

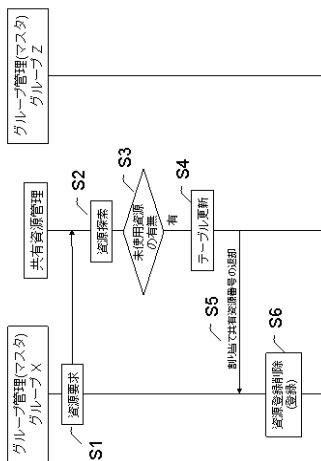
グループID	X
参加ノード情報	.....
保有共有資源情報 (番号 / 共有数)	.....
保有共有資源情報 (番号 / 共有数)	.....

全ての資源も返却した後、グループは消滅する。

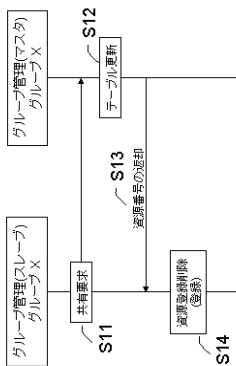
【 図 2 7 】



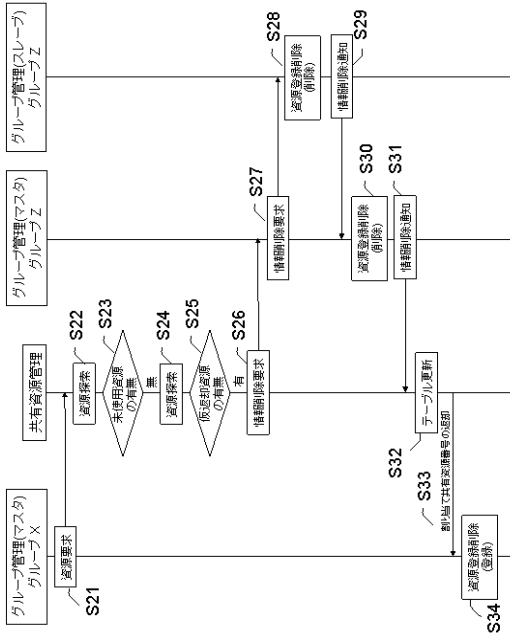
【 図 2 8 】



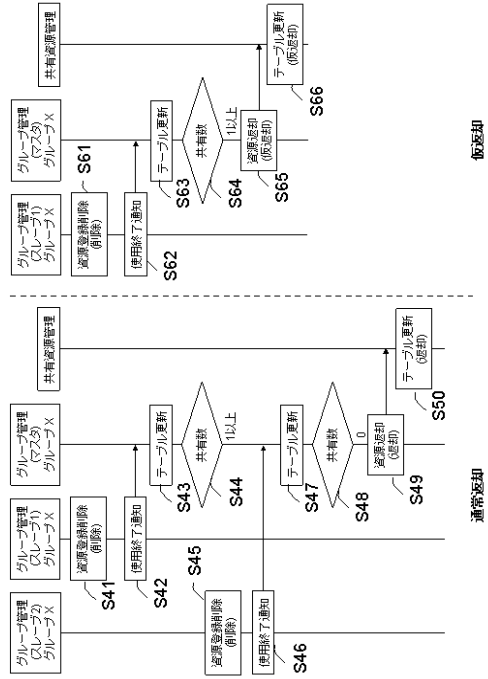
【 図 2 9 】

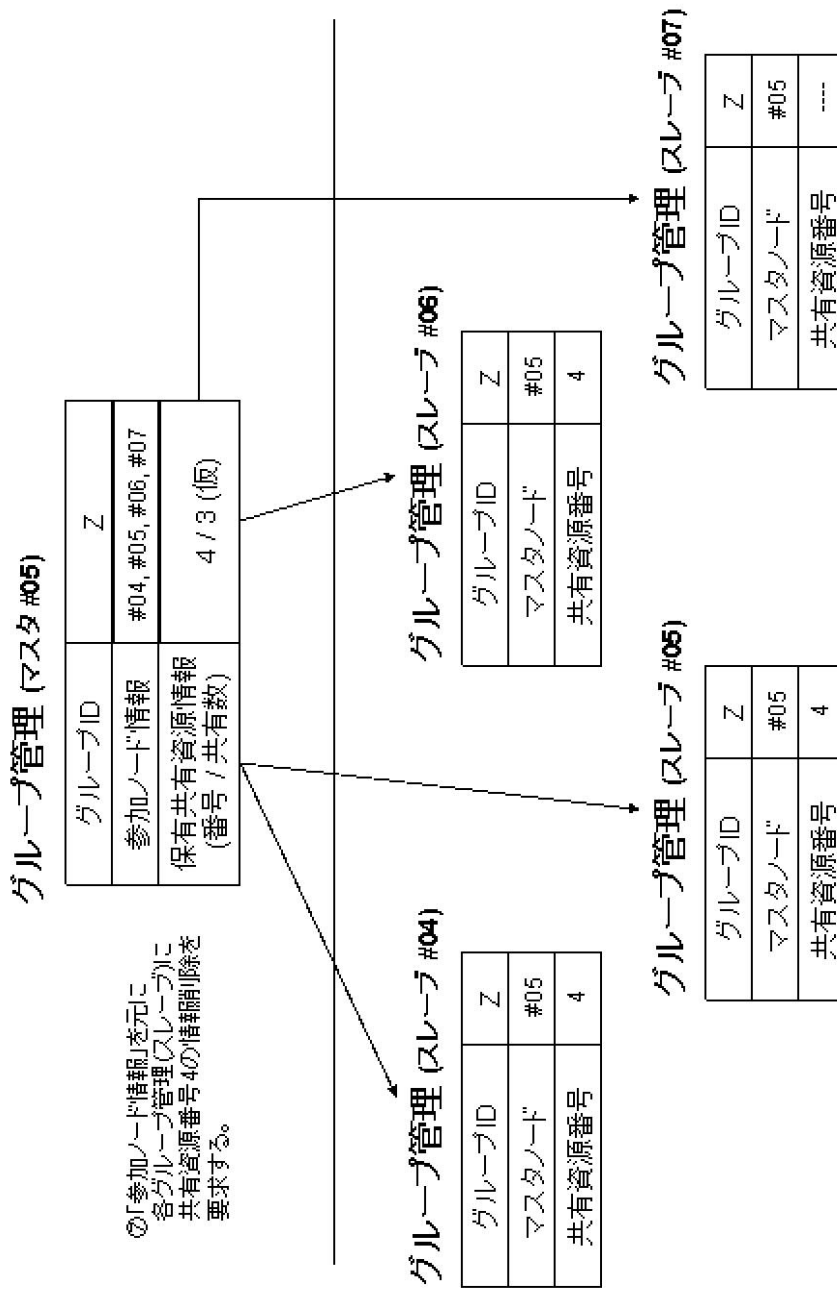


【 図 3 0 】



【 図 3 1 】





③要求を受信した各グループ管理 (スレーブ) は  
共有資源番号4の情報を削除する。  
(情報が無い場合は何もしない)

【 図 2 1 】

### 共有資源管理

「使」:グループに割り当てられている共有資源(割り当て不可)  
「未」:割り当て可能な共有資源  
「仮」:割り当て可能だが、情報削除処理を必要とする共有資源

資源番号	状態	情報削除処理コスト	グループ管理ノード	グループID
1	使	---	#00	X
2	仮	12	#24	S
3	使	---	#55	R
4	未	---	---	---

⑩資源番号4を割り当てるため、  
情報を更新する。  
(使, ---, #00, X)

⑨割り当て共有資源番号4を通知。

### グループ管理 (マスタ #00)

グループID	X
参加ノード情報	#00, #01, #02
保有共有資源情報 (番号 / 共有数)	1 / 3
保有共有資源情報 (番号 / 共有数)	

⑨通知された共有資源番号4を登録する。

【 図 2 2 】

### 共有資源管理

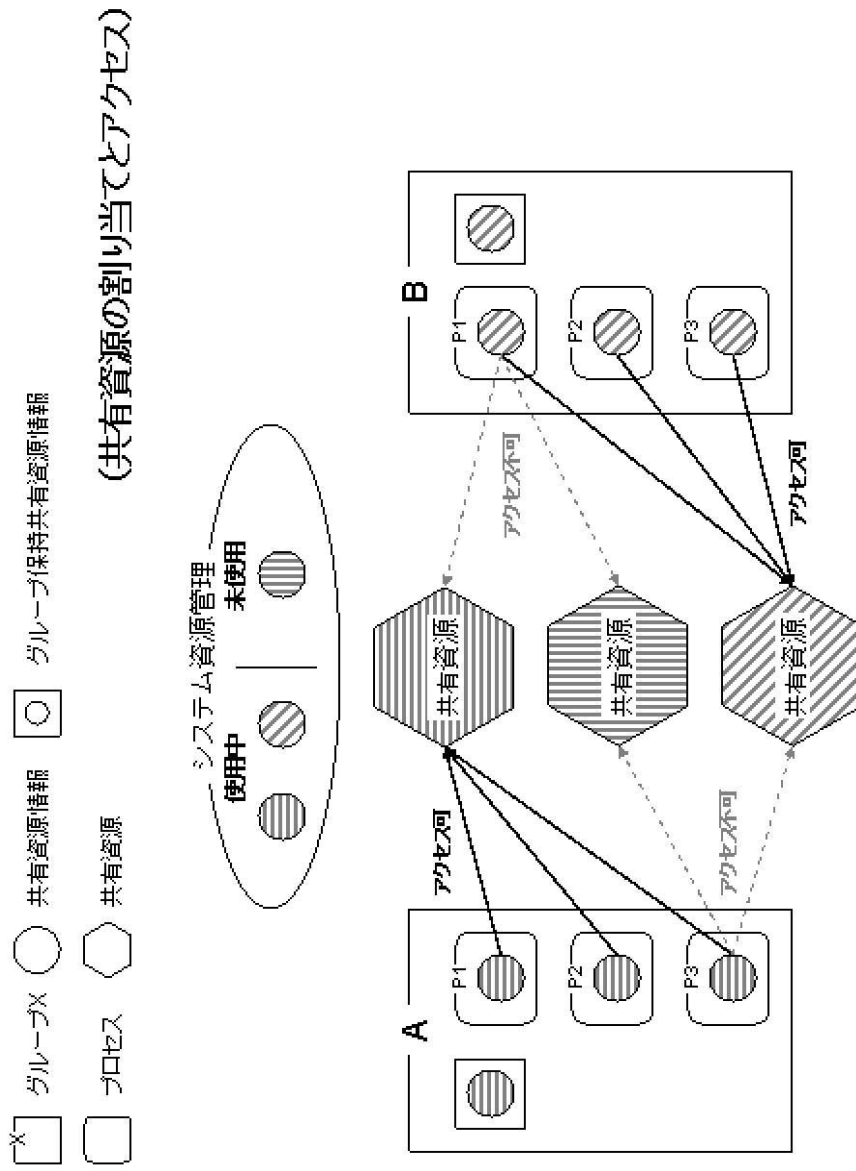
「使」:グループに割り当てられている共有資源割り当て不可)  
「未」:割り当て可能な共有資源  
「仮」:割り当て可能だが、情報削除処理を必要とする共有資源

資源番号	状態	情報削除処理コスト	グループ管理ノード	グループID
1	使	---	#00	X
2	仮	12	#24	S
3	使	---	#55	R
4	使	---	#00	X

### グループ管理 (マスタ #00)







グループID	X
参加ノード情報	#00, #01, #02
保有共有資源情報 (番号 / 共有数)	1 / 3
保有共有資源情報 (番号 / 共有数)	4 / 0

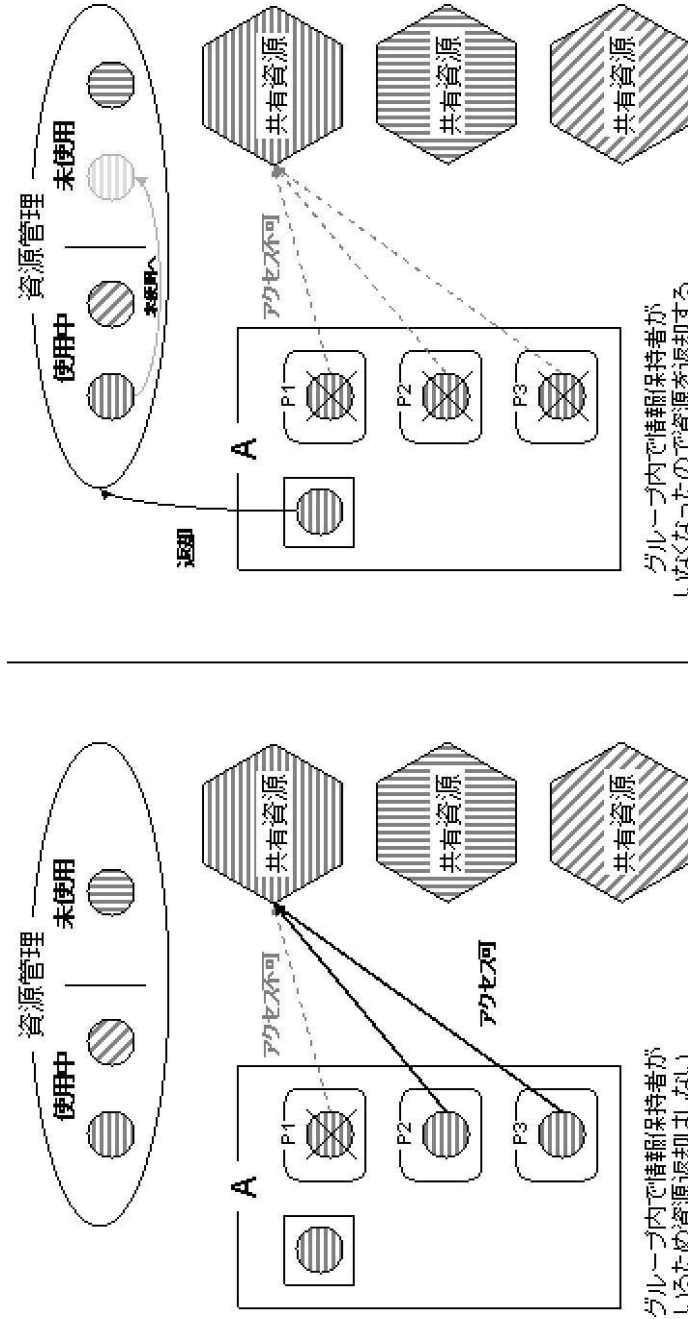
【図 3 2】



【図 33】

(従来の一般的な資源返却方法)  
(プロセス毎の返却)

-  グループX
-  共有資源情報
-  グループ保持共有資源情報
-  プロセス
-  共有資源
-  共有資源

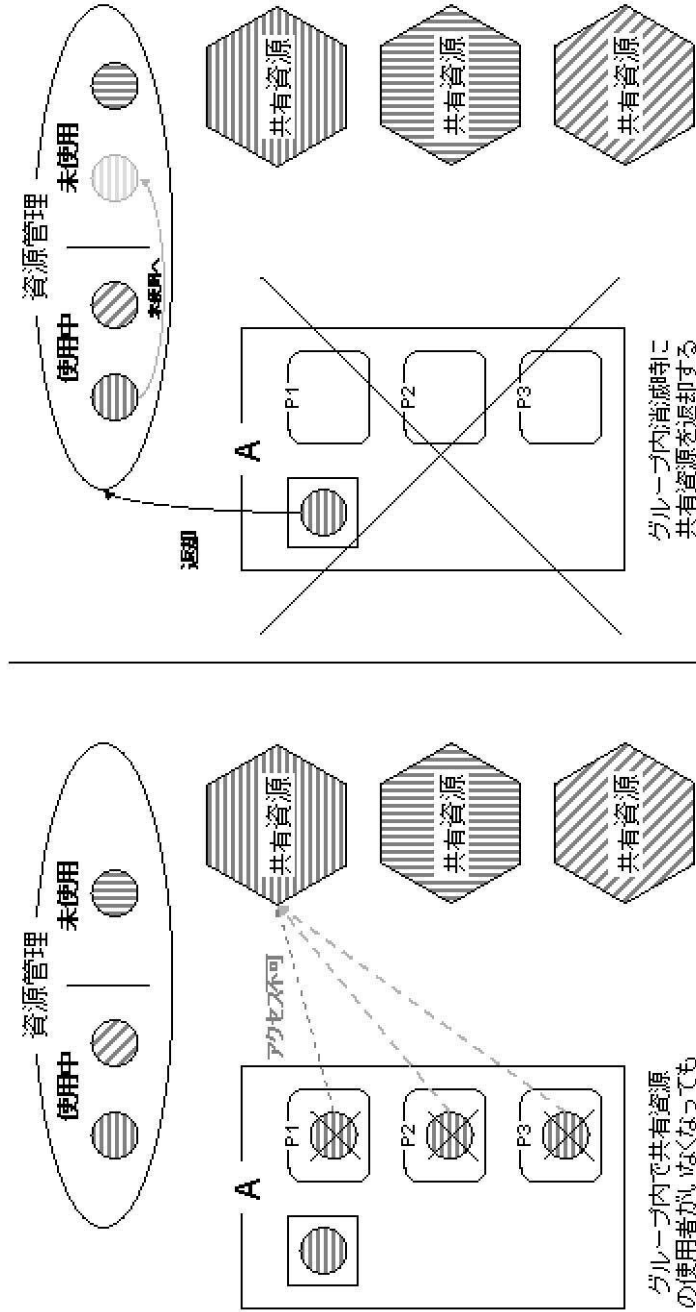


グループ内で情報保持者が  
いなくなったので資源を返却する

【 図 3 4 】







(従来の一般的な資源返却方法)  
(グループが消費時の返却)

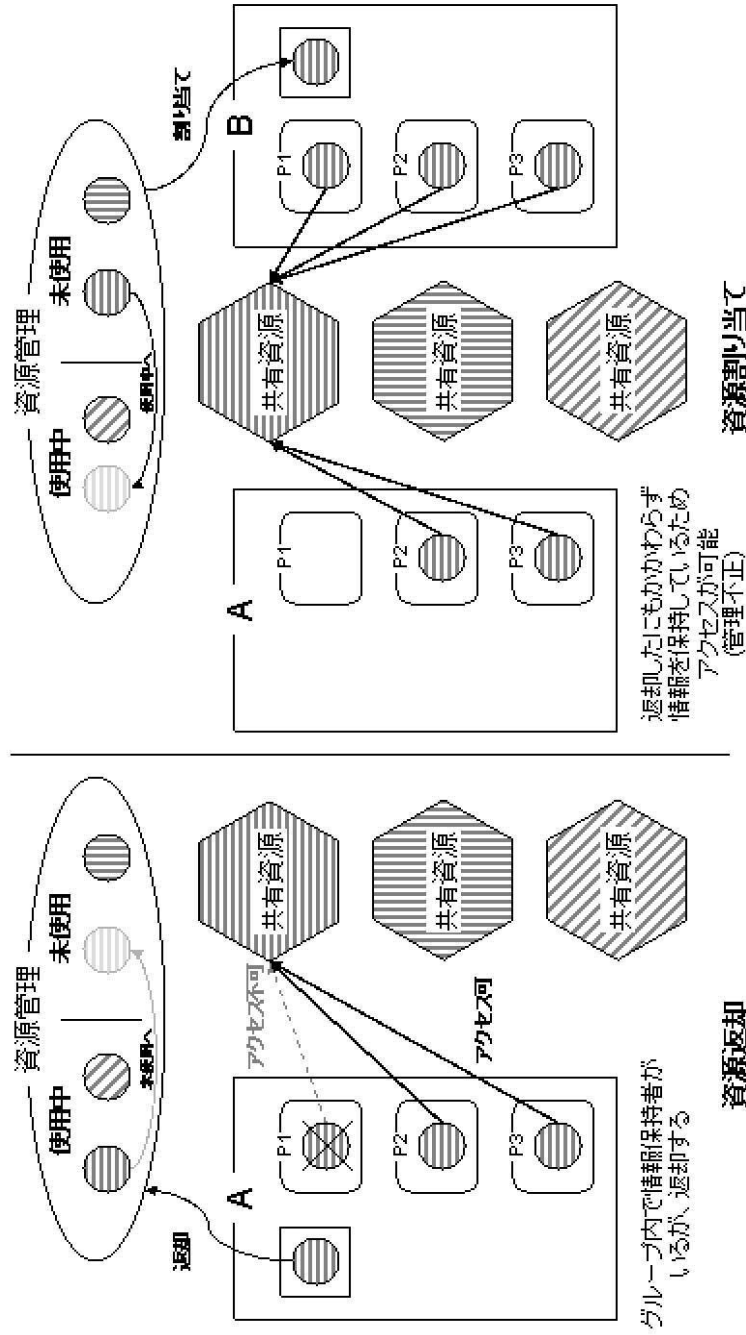
- グループX
- 共有資源情報
- グループ保持共有資源情報
- プロセス
- 共有資源



【 図 3 5 】

(代表返却方法)

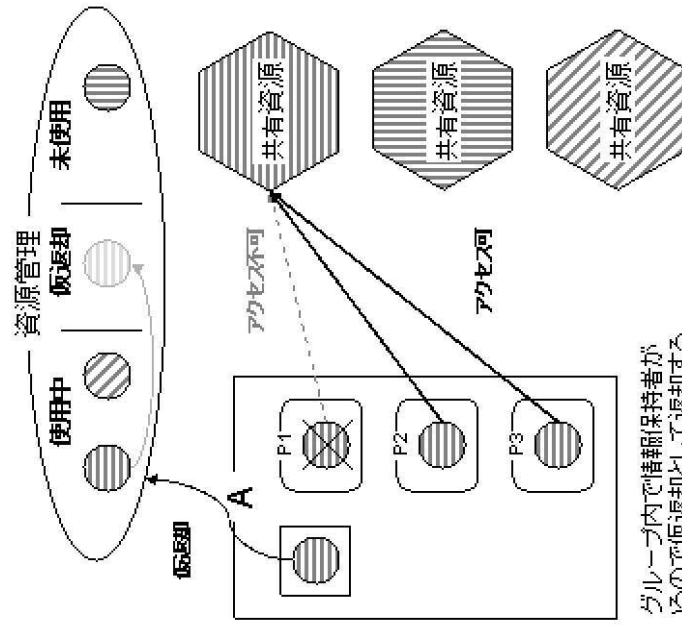
-  グループX
-  グループ
-  プロセス
-  共有資源情報
-  共有資源
-  グループ保持共有資源情報



資源割り当て

資源返却

グループX     共有資源情報     グループ保持共有資源情報  
 プロセス     共有資源



資源返却

○仮返却資源

- ・システム内に同資源の情報保持者がいる。
- ・仮返却資源を保持していたグループの消滅によって未使用資源になる。

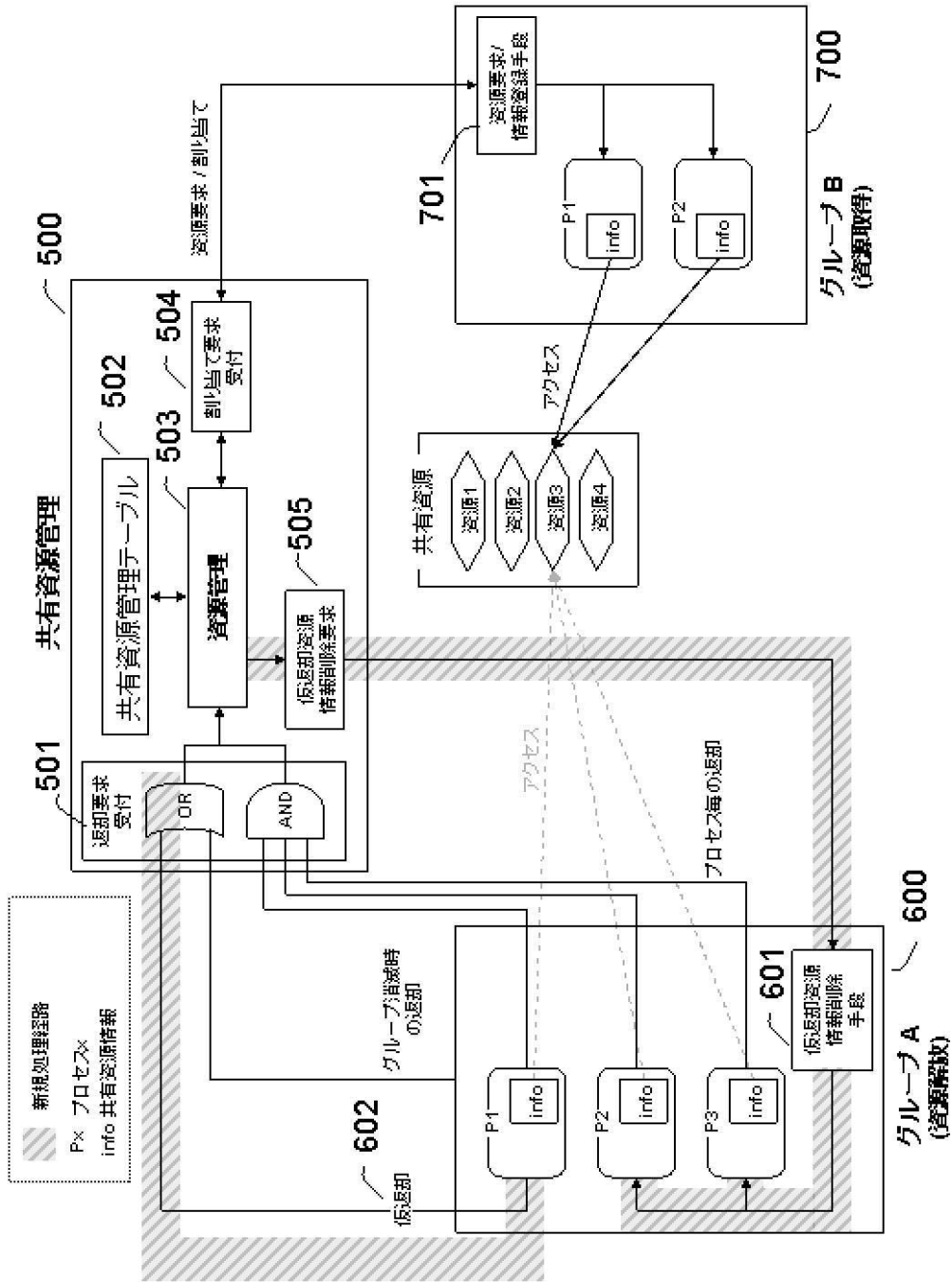
○資源割り当て

- ・未使用資源：従来同様に割り当てる。
  - ・仮返却資源：システム内に同資源の情報保持者に情報削除を要求し、同資源の情報保持者がいないことを確認した後に割り当てる。
- 未使用資源がある場合は、仮返却資源の割り当てを行わない。

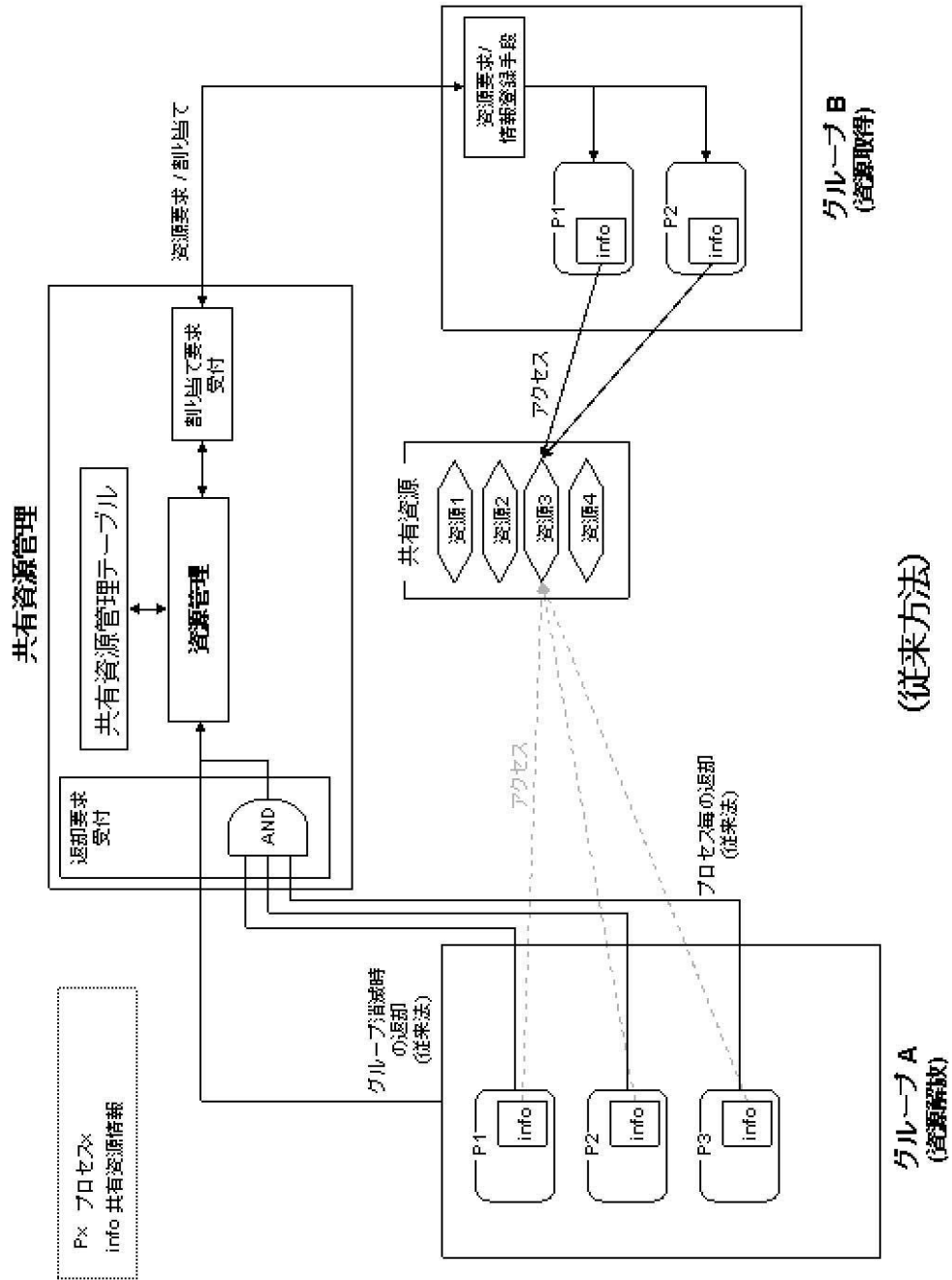
○利点

- ・従来の一般的な方法より資源返却コストが低い。
- ・システムの運用状況に応じた資源運用ができる。
- ・管理をわけることなく、従来法との共存が可能である。

【図 37】



【 図 3 8 】



【 図 3 9 】

