(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3778593号
(P3778593)
(45)発行日　平成18年5月24日(2006.5.24)　(24)登録日　平成18年3月10日(2006.3.10)

(51)Int.Cl.　F I
G06F 12/00 (2006.01)　G06F 12/00 535Z

請求項の数 12 (全 17 頁)

(21)出願番号　特願平7-157452
(22)出願日　平成7年6月23日(1995.6.23)
(65)公開番号　特開平9-6661
(43)公開日　平成9年1月10日(1997.1.10)
審査請求日　平成14年1月31日(2002.1.31)

(73)特許権者　000005223
富士通株式会社
神奈川県川崎市中原区上小田中４丁目１番１号
(73)特許権者　000136136
株式会社ＰＦＵ
石川県かほく市宇野気ヌ９８番地の２
(74)代理人　100077517
弁理士　石田　敬
(74)代理人　100100871
弁理士　土屋　繁
(74)代理人　100088269
弁理士　戸田　利雄
(74)代理人　100082898
弁理士　西山　雅也

最終頁に続く

(54)【発明の名称】システム資源管理装置および管理方法

(57)【特許請求の範囲】

【請求項１】

ネットワーク（Ｌ）に接続された複数のコンピュータシステム（１－１、１－２、…、１－ｎ）中の特定のコンピュータシステム（１－１）に設けられ、かつ、該複数のコンピュータシステム（１－１、１－２、…、１－ｎ）上の複数の実行体により共通に使用される共用資源（３）を直接管理する資源管理部（４）を有し、

該資源管理部（４）は、任意の実行体に関し前記共用資源（３）の入出力処理を行う入出力処理部（５）と、他の任意の実行体に対する前記共用資源（３）のロック処理を行うロック処理部（６）とを含むシステム資源管理装置において、

前記特定のコンピュータシステム（１－１）上のプロセス手段は、

前記入出力処理部（５）に対し、前記コンピュータシステム（１－１、１－２、…、１－ｎ）上の任意の実行体による前記共用資源（３）の入出力処理の要求を呈示する入出力処理要求部（７）を含む第１の実行体（ＡＵ）と、

該第１の実行体（ＡＵ）を識別するための識別子を使用し、前記第１の実行体（ＡＵ）を代行して前記ロック処理部（６）に対し前記共用資源（３）のロック処理を要求する代行ロック処理要求部（１０）を含む第２の実行体（ＢＵ）とに分割して配置されることを特徴とするシステム資源管理装置。

【請求項２】

前記識別子が、前記特定のコンピュータシステム（１－１）内で前記第１の実行体（ＡＵ）を一意に識別することが可能な実行体識別子と、前記第１の実行体（ＡＵ）により前記

共用資源（３）を一意に識別することが可能なファイル識別子とからなり、前記第２の実行体（ＢＵ）は、前記実行体識別子およびファイル識別子を用いて前記共用資源（３）のロック処理を要求する請求項１記載の管理装置。

【請求項３】

前記共用資源（３）に対するロック処理として、前記共用資源（３）内のオフセットと長さを指定することにより前記共用資源（３）をロック状態にし、該ロック状態になった領域のみに対して他の実行体からのアクセスを制限する請求項１記載の管理装置。

【請求項４】

前記共用資源（３）に対するロック処理として、前記共用資源（３）をオープン状態にするときの指定により、前記共用資源（３）全体に対して他の実行体からのアクセスを制限する請求項１記載の管理装置。

【請求項５】

前記共用資源（３）の入出力処理を要求する前記第１の実行体（ＡＵ）が複数存在する場合、一つの独立した前記第２の実行体（ＢＵ）が、すべての前記第１の実行体（ＡＵ）を代行して前記ロック処理を要求する請求項１記載の管理装置。

【請求項６】

複数の他のコンピュータシステム（１－２～１－ｎ）上の実行体からの前記共用資源（３）のロック処理の要求を受け付けるために、前記第２の実行体（ＢＵ）が、さらに、前記他のコンピュータシステム（１－２～１－ｎ）上の実行体からの前記ロック処理の要求に対し仮想的な識別子を自動的に割り当てる識別子自動割り当て処理部（９）を含み、

前記第２の実行体（ＢＵ）は、前記他のコンピュータシステム（１－２～１－ｎ）からのロック処理の要求に対し自動的に割り当てられた仮想的な識別子を用いて、前記共用資源（３）のロック処理を要求することにより、すべての前記コンピュータシステム（１－１、１－２、…、１－ｎ）における任意の実行体間での排他制御を行う請求項１記載の管理装置。

【請求項７】

ネットワークに接続された複数のコンピュータシステム中の特定のコンピュータシステムに設けられ、かつ、該複数のコンピュータシステム上の各種の処理により共通に使用される情報記憶部内の共用資源の入出力処理、および、ロック処理を行うためのシステム資源管理方法であって、

前記特定のコンピュータシステム上の処理を、

前記複数のコンピュータシステム上の任意の処理に関する前記共用資源の入出力処理の要求を処理する第１の処理と、

前記共用資源のロック処理の要求を処理する第２の処理とに分割し、

該第２の処理により、前記第１の処理を識別するための識別子に基づき前記第１の処理を代行して前記共用資源のロック処理を行うことを特徴とするシステム資源管理方法。

【請求項８】

前記識別子として、前記特定のコンピュータシステム内で前記第１の処理を一意に識別することが可能な処理識別子と、前記第１の処理により前記情報記憶部内で前記共用資源を一意に識別することが可能なファイル識別子とを用いて前記共用資源のロック処理を行う請求項７記載の管理方法。

【請求項９】

前記共用資源に対するロック処理として、前記共用資源内のオフセットと長さを指定することにより前記共用資源をロック状態にし、該ロック状態になった領域のみに対して他の処理からのアクセスを制限する請求項７記載の管理方法。

【請求項１０】

前記共用資源に対するロック処理として、前記共用資源をオープン状態にするときの指定により、前記共用資源全体に対して他の処理からのアクセスを制限する請求項７記載の管理方法。

【請求項１１】

前記共用資源の入出力処理を行う前記第１の処理が複数存在する場合、一つの独立した前記第２の処理が、すべての前記第１の処理を代行して前記ロック処理を行う請求項７記載の管理方法。

【請求項１２】

複数の他のコンピュータシステム上の処理からの前記共用資源のロック処理の要求を受け付けるために、前記第２の処理内で、前記他のコンピュータシステム上の処理からの前記ロック処理の要求に対し仮想的な識別子を自動的に割り当て、該割り当てられた仮想的な識別子を用いて、前記共用資源のロック処理を行うことにより、すべての前記コンピュータシステムにおける任意の処理間での排他制御を行う請求項７記載の管理方法。

【発明の詳細な説明】

【０００１】

【産業上の利用分野】

本発明は、ＬＡＮ（ローカルエリア・ネットワーク）等のネットワークを通して接続されている複数のコンピュータシステム中の特定のコンピュータシステムに設けられた共用のシステム資源を管理するためのシステム資源管理装置および管理方法に関する。

【０００２】

さらに詳しくいえば、本発明は、上記コンピュータシステムが、入出力処理や編集等の処理を実行するための各種の実行体（ここでいう実行体には、プロセス、ジョブ等が含まれる）を含む複数のマシンである場合に、これらのマシン間で共有する共用のシステム資源｛以下、共用資源（Shared Resource ）と略記する｝、例えば、情報記憶装置内の共用ファイルをあるマシン内の実行体により使用することを明示する処理、すなわち、共用ファイルのロック処理を効率良く行うための一手法について言及するものである。

【０００３】

ここで、ロック処理とは、特定のマシンに設けられた共用ファイルを他のマシン中の２種以上の実行体から同時にアクセスすることにより共用ファイルの内容が破壊されるのを防止するために、ある一つの実行体が共用ファイルの一部または全体を現在使用していることを明示して他の実行体の共用ファイルへのアクセスを制限する処理を指している。

【０００４】

【従来の技術】

図７および図８は従来のシステム資源管理装置の構成を示す図であり、図７は、従来のシステム資源管理装置の第１の例を示すブロック図、図８は、従来のシステム資源管理装置の第２の例を示すブロック図である。ただし、ここでは、特に、複数のコンピュータシステム上の複数の実行体により共通に使用される共用ファイル等の共用資源に対する入出力処理やロック処理を管理するための装置構成を例示することとする。

【０００５】

図７および図８においては、複数のコンピュータシステムとして、３種のマシンにそれぞれ対応するコンピュータシステム１００－１、１００－２および１００－３が、ＬＡＮ等のネットワークＬを通して互いに接続されている。これらのコンピュータシステム中の特定のコンピュータシステム（例えば、コンピュータシステム１００－１）には、情報を前もって記憶しておくためのディスク装置等からなる情報記憶部２が設けられている。上記コンピュータシステム１００－１の情報記憶部２は、ネットワークＬを通して他のコンピュータシステム１００－２、１００－３により共用されている。

【０００６】

さらに詳しくいえば、上記の情報記憶部２には、複数のコンピュータシステム１００－１、１００－２および１００－３上の各種の実行体により共通に使用することが可能な共用ファイル等の共用資源３が含まれている。さらに、情報記憶部２内の共用資源３に対する入出力処理やロック処理は、コンピュータシステム１００－１内で共用資源３の管理機能を有する資源管理部（例えば、ファイル管理部）４に設けられた入出力処理部５やロック処理部６に要求を呈示することにより行われる。

【０００７】

図７に示す従来の第１の例の構成では、コンピュータシステム１００－１には、実行体の一種であるプロセスＥ２およびプロセスＥ３が存在し、コンピュータシステム１００－２には、実行体の一種であるプロセスＣ２が存在し、コンピュータシステム１００－３には、実行体の一種であるプロセスＣ３が存在している。

【０００８】

プロセスＥ２は、このプロセスＥ２内の入出力処理要求部７およびロック処理要求部８から、同プロセスＥ２が属するコンピュータシステム１００－１内の共用資源管理部４の入出力処理部５やロック処理部６に対して、それぞれ共用資源３の入出力処理要求およびロック処理要求を行う。さらに、コンピュータシステム１００－２内のプロセスＣ２の入出力処理要求部７－２およびロック処理要求部８－２から、共用資源管理部４の入出力処理部５やロック処理部６に対して、それぞれ共用資源３の入出力処理要求およびロック処理要求を受け付けて処理する。

【０００９】

同様に、プロセスＥ３は、このプロセスＥ３内の入出力処理要求部７－１およびロック処理要求部８－１から、同プロセスＥ３が属するコンピュータシステム１００－１内の共用資源管理部４の入出力処理部５およびロック処理部６に対して、それぞれ共用資源３の入出力処理要求およびロック処理要求を行う。さらに、コンピュータシステム１００－３内のプロセスＣ３の入出力処理要求部７－３およびロック処理要求部８－３から、共用資源管理部４の入出力処理部５やロック処理部６に対して、それぞれ共用資源３の入出力処理要求およびロック処理要求を受け付けて処理する。

【００１０】

このように、図７のシステム資源管理装置によれば、他のコンピュータシステム（例えば、コンピュータシステム１００－２、１００－３）からの共用資源３への入出力処理要求およびロック処理要求に対する受付処理は、共用資源３の存在するコンピュータシステム１００－１内で他のコンピュータシステムのプロセス（例えば、プロセスＣ２、Ｃ３）にそれぞれ対応するようなプロセス（例えば、プロセスＥ２、Ｅ３）にて行われていた。

【００１１】

次に、図８に示す従来の第２の例の構成では、コンピュータシステム１００－１には、実行体の一種であるプロセスＦとプロセスＧとプロセスＣ１とが存在し、コンピュータシステム１００－２には、前述の図７の場合と同様のプロセスＣ２が存在し、コンピュータシステム１００－３には、前述の図７の場合と同様のプロセスＣ３が存在している。この場合、コンピュータシステム１００－１のプロセスＦは、一つでも複数でも構わない。

【００１２】

図８では、プロセスＣ２の入出力処理要求部７－２、および、プロセスＣ３の入出力処理要求部７－３からネットワークＬを通してコンピュータシステム１００－１に送られてくる共用資源３の入出力処理要求は、プロセスＦが受け取る。さらに、このプロセスＦは、同プロセスＦ内の入出力処理要求部７から資源管理部４に対して共用資源３の入出力処理要求を行う。

【００１３】

また一方で、プロセスＣ２のロック処理要求部８－２、および、プロセスＣ３のロック処理要求部８－３からネットワークＬを通してコンピュータシステム１００－１に送られてくる共用資源３のロック処理要求は、プロセスＧが受け取る。さらに、このプロセスＧは、コンピュータシステム１００－１内の資源管理部４に対して共用資源３のロック処理要求は行わずに、プロセスＧに設けられたプロセス内ロック処理要求受付部２１により同プロセスＧ内のみでの排他制御を行っていた。

【００１４】

【発明が解決しようとする課題】

しかしながら、従来の第１の例（図７）のように、共用資源３の入出力処理およびロック処理を一つの実行体、例えば、一つのプロセスとして実現しようとすると、入出力処理とロック処理が並列して行えず、コンピュータシステム内の処理の効率が悪くなるという問

題が生ずる。さらに、一つのプロセスで入出力処理要求に対する処理とロック処理要求に対する処理の両方を行うため、システム資源管理装置自体の構造が複雑になるという問題も生ずる。

【００１５】

また一方で、図８のように、共用資源３の入出力処理要求を受け付けて処理するプロセスＦと、ロック処理要求を受け付けて処理するプロセスＧとを互いに独立したプロセスに分割し、プロセスＧのみでの排他制御をする場合は、資源管理部４に対して共用資源３のロック処理要求は行わない。このため、コンピュータシステム１００－１内のプロセスＣ１（図８では省略、後述の図１参照）と、コンピュータシステム１００－２内のプロセスＣ２やコンピュータシステム１００－３の内のプロセスＣ３との間ではロック処理要求が無効になるという問題が生ずる。

【００１６】

コンピュータシステム１００－１内のプロセスＣ１との間でロック処理要求が有効になるように、コンピュータシステム１００－２内のプロセスＣ２のロック処理要求部８－２からのロック処理要求により、プロセスＧのロック処理要求部（図８では省略）が資源管理部４に対してロック処理要求を行うことも考えられる。しかしながら、この場合、その後に、プロセスＣ２の入出力処理要求部７－２からの入出力処理要求に応じてプロセスＦの入出力処理要求部７から資源管理部４に呈示される入出力処理要求が、上記のプロセスＧによるロック処理要求によって制限されてしまう。このようなプロセスＦの入出力処理要求の制限を回避しようすると、プロセスＧは、資源管理部４に対して共用資源３のロック処理要求をすることが不可能になる。この結果、上記のコンピュータシステム１００－１内のプロセスと、他のコンピュータシステム１００－２、１００－３内のプロセスＣ３との間でロック処理要求が無効になるという問題は以前として残ることのなる。

【００１７】

本発明は、上記問題点に鑑みてなされたものであり、複数のコンピュータシステムシステム上に存在するようなある一つのプロセス等の実行体により共用ファイル等の共用資源のロック処理要求をした後に、この共用資源の入出力処理を行う他の実行体からの入出力処理要求が制限されたり、入出力処理要求およびロック処理要求の両方の受付処理によって装置自体の構造が複雑になったりすることなく、共用資源の入出力処理およびロック処理を効率良く行うことが可能なシステム資源管理装置および管理方法を提供することを目的とするものである。

【００１８】

【課題を解決するための手段】

図１は、本発明の原理構成を示すブロック図である。なお、これ以降、前述した構成要素と同様のものについては、同一の参照番号を付して表すこととする。

図１において、本発明のシステム資源管理装置は、前述の従来の資源管理装置（図７および図８）と同じように、ネットワークＬに接続された複数のコンピュータシステム１－１、１－２、…、１－ｎ（ｎは任意の正の整数）中の特定のコンピュータシステム１－１に設けられた資源管理部４を有している。この資源管理部４は、複数のコンピュータシステム１－１、１－２、…、１－ｎ上の複数の実行体により共通に使用される共用資源３を直接管理するためのものである。さらに、この資源管理部４は、従来の場合と同じように、任意の実行体に関し共用資源３の入出力処理を行う入出力処理部５と、他の任意の実行体に対する共用資源３のロック処理を行うロック処理部６とを含む。

【００１９】

さらに、上記問題点を解決するために、図１に示すような本発明のシステム資源管理装置における特定のコンピュータシステム１－１上の実行体は、入出力処理部５に対し、上記コンピュータシステム１－１、１－２、…、１－ｎ上の任意の実行体による共用資源３の入出力処理の要求を呈示する入出力処理要求部７を含む第１の実行体ＡＵと、この第１の実行体ＡＵを識別するための識別子を使用し、上記第１の実行体ＡＵを代行してロック処理部６に対し共用資源３のロック処理を要求する代行ロック処理要求部１０を含む第２の

実行体ＢＵとに分割して配置される。

【００２０】

好ましくは、本発明のシステム資源管理装置に使用される識別子は、コンピュータシステム１－１内で第１の実行体ＡＵを一意に識別することが可能な実行体識別子と、第１の実行体ＡＵにより共用資源３を一意に識別することが可能なファイル識別子とからなり、第２の実行体ＢＵは、上記実行体識別子およびファイル識別子を用いて共用資源３のロック処理を要求する。

【００２１】

さらに、好ましくは、本発明のシステム資源管理装置は、共用資源３に対するロック処理として、共用資源３内のオフセットと長さを指定することにより共用資源３をロック状態にし、このロック状態になった領域のみに対して他の実行体からのアクセスを制限する構成になっている。

さらに、好ましくは、本発明のシステム資源管理装置は、共用資源３に対するロック処理として、共用資源３をオープン状態にするときの指定により、共用資源３全体に対して他の実行体からのアクセスを制限する構成になっている。

【００２２】

さらに、好ましくは、本発明のシステム資源管理装置は、共用資源３の入出力処理を要求する第１の実行体ＡＵが複数存在する場合、一つの独立した上記第２の実行体ＢＵが、すべての第１の実行体ＡＵを代行してロック処理を要求する構成になっている。

さらに、好ましくは、特定のコンピュータシステム１－１以外の他のコンピュータシステム１－２～１－ｎにおける実行体（例えば、実行体ＣＵ２～ＣＵｎ）からの共用資源３のロック処理の要求を受け付けるために、第２の実行体ＢＵは、さらに、上記の他のコンピュータシステム１－２～１－ｎにおけるプロセス手段からのロック処理の要求に対し仮想的な識別子を自動的に割り当てる識別子自動割り当て処理部９を備えている。上記第２の実行体ＢＵは、他のコンピュータシステム１－２～１－ｎからのロック処理の要求に対し自動的に割り当てられた仮想的な識別子を用いて、共用資源３のロック処理を要求することにより、すべてのコンピュータシステム１－１、１－２、…、１－ｎにおける任意の実行体間（例えば、実行体ＣＵ１と実行体ＣＵ２～ＣＵｎとの間）での排他制御を行う構成になっている。

【００２３】

また一方で、例えば図１に示すようなシステム資源管理装置に遂行される本発明のシステム資源管理方法は、ネットワークに接続された複数のコンピュータシステム中の特定のコンピュータシステムに設けられ、かつ、これらの複数のコンピュータシステムにおける各種の処理（プロセス等の実行体により実行される種々のタイプの処理という意味で用いられる）により共通に使用される情報記憶部内の共用資源の入出力処理、および、ロック処理を行うための管理方法に関係する。

【００２４】

さらに、本発明のシステム資源管理方法では、上記特定のコンピュータシステム上の処理を、複数のコンピュータシステム上の任意の処理に関する共用資源の入出力処理の要求を処理する第１の処理と、共用資源のロック処理の要求を処理する第２の処理とに分割する。さらにまた、この第２の処理により、第１の処理を識別するための識別子に基づき上記第１の処理を代行して共用資源のロック処理を行うようにしている。

【００２５】

好ましくは、本発明のシステム資源管理方法では、上記の識別子として、特定のコンピュータシステム内で第１の処理を一意に識別することが可能な処理識別子（前述の実行体識別子とほぼ同じ意味で用いられる）と、第１の処理により情報記憶部内で共用資源を一意に識別することが可能なファイル識別子とを用いて共用資源のロック処理を行うようにしている。

【００２６】

さらに、好ましくは、本発明のシステム資源管理方法では、共用資源に対するロック処理

として、共用資源内のオフセットと長さを指定することにより共用資源をロック状態にし、このロック状態になった領域のみに対して他の処理からのアクセスを制限するようにしている。
さらに、好ましくは、本発明のシステム資源管理方法では、共用資源に対するロック処理として、共用資源をオープン状態にするときの指定により、共用資源全体に対して他の処理からのアクセスを制限するようにしている。
【００２７】
さらに、好ましくは、本発明のシステム資源管理方法では、共用資源の入出力処理を行う第１の処理が複数存在する場合、一つの独立した第２の処理は、すべての第１の処理を代行してロック処理を行うようにしている。
さらに、好ましくは、本発明のシステム資源管理方法では、特定のコンピュータシステム以外の他のコンピュータシステムにおける処理からの共用資源のロック処理の要求を受け付けるために、第２の処理内で、上記の他のコンピュータシステム上の処理からのロック処理の要求に対し仮想的な識別子を自動的に割り当て、この割り当てられた仮想的な識別子を用いて、共用資源のロック処理を行うことにより、すべてのコンピュータシステムにおける任意の処理間での排他制御を行うようにしている。
【００２８】
さらに詳しく説明すると、図１において、共用資源３を有する特定のコンピュータシステム１－１以外の他のコンピュータシステム１－２～１－ｎにおける共用資源３の入出力処理要求に対する受付処理と、ロック処理要求に対する受付処理が並列して効率良く行えるように、コンピュータシステム１－１上の共用資源３の入出力処理要求を受け付けて処理する第１の実行体ＡＵと、上記共用資源３のロック処理要求を受け付けて処理する第２の実行体ＢＵとを互いに独立した実行体に分割して配置する。ここで、第１の実行体ＡＵは、複数の他のコンピュータシステム１－２～１－ｎ上の実行体からの共用資源３に対する入出力処理要求を並列に受け付けられるようにコンピュータシステム１－１内に複数存在しても構わないが、第２の実行体ＢＵは、ロック処理の関係上コンピュータシステム１－１内に一つだけ存在することが必要である。
【００２９】
第１の実行体ＡＵは、他のコンピュータシステム１－２～１－ｎにおける実行体の入出力処理要求部７－２～７－ｎからの共用ファイル３の入出力処理要求を受け取り、上記第１の実行体ＡＵの入出力処理要求部７から資源管理部４に対して共用資源３の入出力処理要求をする。
また一方で、第２の実行体ＢＵは、他のコンピュータシステム１－２～１－ｎにおける実行体のロック処理要求部８－２～８－ｎからの共用資源３のロック処理要求を受け取り、上記第２の実行体ＢＵのロック処理要求部８から資源管理部４に対して共用資源３のロック処理要求をする。
【００３０】
しかし、この場合、第２の実行体ＢＵは、ロック処理対象の共用資源３のファイル識別子を使用し、さらに、実行体識別子として第２の実行体ＢＵ自身の識別子ではなく、第１の実行体ＡＵの実行体識別子を使用して、第２の実行体ＢＵの代行ロック処理要求部１０から資源管理部４に対して共用資源３のロック処理要求をする。このとき、第１の実行体ＡＵが複数存在すると、第２の実行体ＢＵは、ロック処理要求を行った他のコンピュータシステムの実行体に対応する第１の実行体ＡＵのプロセス識別子を使用する。
【００３１】
すなわち、第２の実行体ＢＵは、この第２の実行体ＢＵ自身の識別子ではなく、第１の実行体ＡＵの実行体識別子を使用して、資源管理部４に対して共用資源３のロック処理要求をするため、資源管理部４は第１の実行体ＡＵがロック処理要求をしたものとみなして共用資源３のロック処理を実行する。
さらに、一つの第２の実行体ＢＵが複数の他のコンピュータシステム上の実行体からのロック処理要求を受け付けて処理する際に、この第２の実行体ＢＵの識別子自動割当て処理

部９により自動的に他のコンピュータシステム上の実行体からのロック処理要求に対して仮想的な識別子（例えば、代行ロック識別子）を割り当てる。さらに、第１の実行体ＡＵの実行体識別子とロック処理対象の共用資源３のファイル識別子に加えて、上記の代行ロック識別子を使用することにより、資源管理部４に対して共用資源３のロック処理要求をする。

【００３２】

【作用】

本発明のシステム資源管理の手法によれば、共用資源３を有する特定のコンピュータシステム１－１上の第２の実行体ＢＵが、この第２の実行体ＢＵ自身の識別子ではなく、コンピュータシステム１－１上の第１の実行体ＡＵの実行体識別子等を使用して、資源管理部４に対して共用資源３のロック処理要求をし、第１の実行体ＡＵがロック処理要求をしたように見せかける。このため、第１の実行体ＡＵは、第２の実行体ＢＵが共用資源３のロック処理要求をした後も、ロック処理がなされた状態の共用資源３に対する入出力処理の影響を受けない。

【００３３】

さらに、第２の実行体ＢＵは、実際に、資源管理部４に対して共用資源３のロック処理要求をしているため、特定のコンピュータシステム１－１における他の実行体ＣＵ１と、他のコンピュータシステム１－２～１－ｎにおける実行体ＣＵ２～ＣＵｎとの間のロック処理要求も有効になる。

さらにまた、第２の実行体ＢＵは、資源管理部４に対して共用資源３のロック処理要求をするときに、他のコンピュータシステム１－２～１－ｎ上の実行体ＣＵ２～ＣＵｎからのロック処理要求に対して自動的に割り当てた仮想的な代行ロック識別子も使用しているため、コンピュータシステム１－１上の実行体ＣＵ１と、他のコンピュータシステム１－２～１－ｎにおける実行体ＣＵ２～ＣＵｎとの間のロック処理要求が可能になり、すべてのコンピュータシステムにおける任意の実行体間での排他制御を行えるようになる。

【００３４】

かくして、本発明では、共用資源の入出力処理要求を受け付けて処理する第１の実行体と、ロック処理要求を受け付けて処理する第２の実行体とを互いに独立した実行体に分割して、第２の実行体が第１の実行体を代行してロック処理要求をすることにより、第２の実行体により共用資源のロック処理要求をした後に、この共用資源の入出力処理を行う他の実行体からの入出力処理要求が制限されることはなくなる。さらに、コンピュータシステムシステム上の異なる実行体による入出力処理要求およびロック処理要求の受付処理を独立に実行することにより、各実行体の負担を軽くすることができるので、従来よりも簡単な装置構成でもって共用資源の入出力処理およびロック処理等を効率良く行うことが可能になる。

【００３５】

さらにまた、一つのコンピュータシステム上の実行体から同コンピュータシステムの資源管理部に対して共用資源のロック処理要求が行われるため、すべてのコンピュータシステムにおける任意の実行体間でのロック処理要求が有効になるので、共用資源の存在するコンピュータシステム上の実行体と、ネットワークに接続される他のコンピュータシステムにおける実行体との間のロック処理要求による排他制御が可能になる。

【００３６】

【実施例】

以下添付図面（図２～図６）を用いて本発明の実施例を詳細に説明する。

図２は、本発明の一実施例によるシステム資源管理装置の構成を示すブロック図、図３は、本発明の一実施例で使用する識別子情報テーブルを示す図、および、図４は、本発明の一実施例で使用する識別子自動割り当て処理テーブルを示す図である。ただし、ここでは、複数のコンピュータシステム１－１～１－ｎ（図１）として、３種のマシンＭ－１、Ｍ－２およびＭ－３がネットワークＬに接続されている場合を例示することとする。さらに、本実施例においては、実行体としてプロセスを用いることとする。

【００３７】
図２において、３種のマシンＭ－１、Ｍ－２およびＭ－３は、お互いにネットワークＬを介して通信することができるように構成されている。特定のコンピュータシステムに対応するマシンＭ－１の情報記憶部２は、共用資源３としての共用ファイル３０を備えている。この共用ファイル３０は、資源管理部４（図１）として機能するファイル管理部４０内の入出力処理部５やロック処理部６を通してアクセスすることができる。すなわち、マシンＭ－１内に設けられた情報記憶部２に存在する共用ファイル３０は、ネットワークＬを通して他のマシンＭ－２、Ｍ－３により共用される構成になっている。
【００３８】
さらに、図２において、マシンＭ－１には第１のプロセスＡ（図１の第１の実行体ＡＵの一例）と第２のプロセスＢ（図１の第２の実行体ＢＵの一例）と他のプロセスＣ１（図１の他の実行体ＣＵ１の一例）とが存在し、マシンＭ－２およびマシンＭ－３には、それぞれ、プロセスＣ２およびプロセスＣ３（図１の他の実行体ＣＵ２，ＣＵ３の一例）が存在する。
【００３９】
前述の図１の場合と同様に、第１のプロセスＡには、入出力処理要求部７が設けられており、第２のプロセスＢには、識別子自動割り当て処理部９と代行ロック処理要求部１０が設けられている。さらに、プロセスＣ１～プロセスＣ３には、それぞれ、入出力処理要求部７－１～７－３とロック処理要求部８－１～８－３が設けられている。
【００４０】
上記の各プロセスは、マシン内で当該プロセスを一意に識別できるプロセス識別子（実行体識別子の一例）１３を有している。さらに詳しくいえば、第１のプロセスＡは「ａ」というプロセス識別子を、第２のプロセスＢは「ｂ」というプロセス識別子を、プロセスＣ１、プロセスＣ２およびプロセスＣ３は「ｃ」というプロセス識別子をもっている。これらのプロセス識別子１３の各々は、各プロセス識別子に対応するプロセスが存在するマシン内でのみ有効である。識別子情報テーブル１１は、第１のプロセスＡと第２のプロセスＢの共用域に存在し、第１のプロセスＡと第２のプロセスＢのどちらからでもアクセスすることができる。
【００４１】
他のマシンＭ－２に存在するプロセスＣ２が、マシンＭ－１の情報記憶部２の共用ファイル３０を利用する場合には、ネットワークＬを通してマシンＭ－１上の第１のプロセスＡに対し、共用ファイル３０をオープン状態にすること等の要求を呈示することによって、プロセスＣ２が利用対象とする共用ファイル３０の共用ファイル識別子１４（図３参照）を要求する。
【００４２】
第１のプロセスＡは、上記の要求を受け取ると、図３の識別子情報テーブル１１の中から、受け取った共用ファイル名１６（例えば、“ＡＢＣＤ”）に基づきプロセスＣ２が利用対象とする共用ファイル３０を探す。利用対象の共用ファイル３０が見つかった場合は、識別子情報テーブル１１のインデックスとしての共用ファイル識別子１４をプロセスＣ２に返す。
【００４３】
もし、対象の共用ファイル３０が見つからなければ、第１のプロセスＡがプロセスＣ２の利用対象の共用ファイル３０をオープン状態にしてから、第１のプロセスＡのプロセス識別子である「ａ」と、プロセスＣ２の利用対象、例えば、ロック処理対象の共用ファイル３０のファイル識別子１５と、共用ファイル名１６とを識別子情報テーブル１１に登録し、共用ファイル識別子１４をプロセスＣ２に返す。
【００４４】
なお、共用ファイル識別子１４として、図３では、レコードのインデックスを使用しているが、別のケースとしてレコード全体を利用することも可能である。
さらに詳しく説明すると、プロセスＣ２が、共用ファイル３０の入出力処理をする場合は

、処理対象の共用ファイル３０の共用ファイル識別子１４を指定し、プロセスＣ２内の入出力処理要求部７－２からネットワークＬを通して、第１のプロセスＡに対し入出力処理要求を送る。この入出力処理要求が送られてきた時点で、第１のプロセスＡは、共用ファイル識別子１４をもとに識別子情報テーブル１１を参照してファイル識別子１５を獲得する。さらに、第１のプロセスＡは、上記の獲得したファイル識別子１５を使用することにより、入出力処理要求部７からファイル管理部４０に対して共用ファイル３０の入出力処理要求を行い、その結果をプロセスＣ２に返す。

【００４５】

また一方で、プロセスＣ２が、共用ファイル３０のロック処理をする場合は、処理対象の共用ファイル３０の共用ファイル識別子１４を指定し、プロセスＣ２内のロック処理要求部８－２からネットワークＬを通して、第２のプロセスＢに対しロック処理要求を送る。このロック処理要求には、共用ファイル識別子１４とロック処理の種類や範囲の他に、ロック処理要求元のマシン名と、プロセスＣ２のプロセス識別子である「ｃ」とが含まれている。この場合、「ｃ」のプロセス識別子の代わりに、マシンＭ－２内のファイル識別子１５を使用することも可能である。

【００４６】

第２のプロセスＢは、上記のロック処理要求を受け取ると、識別子自動割り当て処理部９において図４の識別子自動割り当て処理テーブル１２を参照しながら、ロック処理要求元のマシン名１８とプロセス識別子１３とが一致するレコードを探す。もし、このレコードが見つかれば、当該レコードから代行ロック識別子１９（例えば、「Ｌ１」）を獲得する。もし見つからなければ、識別子自動割り当て処理テーブル１２に対しロック処理要求元のマシン名１８とプロセス識別子１３を新たに登録し、既に登録済みの代行ロック識別子１９と重複しない任意の代行ロック識別子１９を割り当てる。この種の代行ロック識別子１９として、識別子自動割り当て処理テーブル１２のインデックス１７等が利用可能である。

次に、第２のプロセスＢは、識別子情報テーブル１１の中から、ロック処理対象の共用ファイル３０の共用ファイル識別子１４に対応するレコードを探し、その中の入出力処理要求元のプロセスのプロセス識別子１３である「ａ」と、ロック処理対象の共用ファイル３０のファイル識別子１５とを得る。

【００４７】

さらに、第２のプロセスＢは、代行ロック処理要求部１０において、入出力処理要求元のプロセスのプロセス識別子１３である「ａ」と、ファイル識別子１５と、代行ロック識別子１９とを使用して、上記代行ロック処理要求部１０からファイル管理部４０に対し共用ファイル３０のロック処理要求を行い、その結果をプロセスＣ２に返す。

【００４８】

なお、この場合、共用ファイル３０に対するロック処理の一つの種類として、共用ファイル３０内のオフセットと長さを指定することにより共用ファイル３０をロック状態にし、このロック状態になった領域のみに対して他のプロセスからのアクセスを制限するようなロック処理が挙げられる。

さらに、共用ファイル３０に対するロック処理の別の種類として、共用ファイル３０をオープン状態にするときの指定により、共用ファイル３０全体に対して他のプロセスからのアクセスを制限するようなロック処理が挙げられる。

【００４９】

上記の実施例では、第２のプロセスＢが、第１のプロセスＡのプロセス識別子１３である「ａ」を使用して、この第１のプロセスＡがファイル管理部４０に対しロック処理要求をしたように見せかけている。したがって、その後に、前述のように、第２のプロセスＢが代行してロック処理を行った共用ファイル３０の入出力処理要求を再び第１のプロセスＡに送り、この第１のプロセスＡがファイル管理部４０に対して共用ファイル３０の入出力処理要求を行うときに、第１のプロセスＡは、共用ファイル３０の入出力処理に対して上記の第２のプロセスＢのロック処理による制限を受けないことになる。

【００５０】
さらに他のマシンＭ－３に存在するプロセスＣ３が、共用ファイル３０のロック処理要求をする場合には、前述のプロセスＣ２の場合と同様に、マシンＭ－１上の第１のプロセスＡに対しロック処理対象の共用ファイル３０の共用ファイル識別子１４を要求して、第２のプロセスＢにロック処理要求をする。
第２のプロセスＢは、識別子自動割り当て処理部９において、識別子自動割り当て処理テーブル１２を利用することにより、プロセスＣ２に対して割り当てた代行ロック識別子１９とは異なる代行ロック識別子１９を割り当てる。
【００５１】
さらに、代行ロック処理要求部１０において、第１のプロセスＡのプロセス識別子１３である「ａ」と、ファイル識別子１５と、代行ロック識別子１９とを使用することにより、ファイル管理部４０に対して共用ファイル３０のロック処理要求を行う。このように、プロセスＣ２とプロセスＣ３に対する代行ロック処理要求では、互いに異なる代行ロック識別子１９を使用しているため、ファイル管理部４０によってプロセスＣ２とプロセスＣ３との間でロック処理要求による排他制御が行われる。
【００５２】
また一方で、マシンＭ－１上の他のプロセスＣ１が共用ファイル３０のロック処理要求をする場合は、プロセスＣ１のロック処理要求部８－１から直接、ファイル管理部４０に対して共用ファイル３０のロック処理要求を行う。この場合は、プロセス識別子１３として、プロセスＣ１のプロセス識別子「ｃ」を使用し、代行ロック識別子１９は“０”を使用する。すなわち、この場合は、第１のプロセスＡのプロセス識別子「ａ」とは異なるプロセス識別子「ｃ」を使用しているので、ファイル管理部４０により、マシンＭ－１上のプロセスＣ１と他のマシン上のプロセスＣ２、Ｃ３との間でロック処理要求による排他制御が行われる。
【００５３】
さらに、上記の実施例において、ファイル管理部４０内の共用ファイル３０の入出力処理を要求する第１のプロセスＡが複数存在する場合でも、一つの第２のプロセスＢが、すべての第１のプロセスＡを代行して共用ファイル３０に対しロック処理要求を行う。この場合、複数の第１のプロセスＡのプロセス識別子として、「ａ」以外に「ａ１」、「ａ２」…が割り当てられ、他のマシンのロック処理要求元のプロセスのプロセス識別子として、「ｃ」以外に「ｄ」、「ｅ」…が割り当てられる。
【００５４】
図５は、図２の識別子自動割り当て処理部の動作を説明するためのフローチャートであり、図６は、図２の代行ロック処理要求部の動作を説明するためのフローチャートである。これらのフローチャートは、第１のプロセスＡや第２のプロセスＢの機能がコンピュータシステムのＣＰＵ（中央処理装置）等により実現される場合の識別子自動割り当て処理部９および代行ロック処理要求部１０の動作を示すものである。上記のフローチャートに従い、図２の識別子自動割り当て処理部９および代行ロック処理要求部１０の一連の動作を詳しく説明することとする。
【００５５】
まず初めに、図５のステップＳ５１に示すように、第２のプロセスＢの識別子自動割り当て処理部９は、他のマシン上のプロセスから受け付けたロック処理要求をもとに、共用ファイル識別子、ロック処理要求元のマシン名、および、ロック処理要求元のプロセスのプロセス識別子を認識する。
次に、ステップＳ５２およびステップＳ５３において、識別子自動割り当て処理部９は、上記のロック処理要求元のマシン名と、ロック処理要求元のプロセスのプロセス識別子とを使用して識別子自動割り当て処理テーブルを検索することにより、共用ファイル識別子、ロック処理要求元のマシン名、および、ロック処理要求元のプロセスのプロセス識別子が一致するレコードを探す。
【００５６】

目的とするレコードが見つかった場合、ステップＳ５４において、当該レコードに基づき代行ロック識別子を得る。
目的とするレコードが見つからなかった場合、ステップＳ５５において、ロック処理要求元のマシン名１８とロック処理要求元のプロセスのプロセス識別子１３とを識別子自動割り当て処理テーブルに代入することにより、新しいレコードを追加する。さらに、ステップＳ５６において、既に登録済みの代行ロック識別子と重複しない任意の代行ロック識別子を割り当て、上記の新しいレコードに代入する。

【００５７】

その後、代行ロック処理要求部１０は、図６のステップＳ６１およびステップＳ６２に示すように、前述のステップＳ５１の共用ファイル識別子を使用して、識別子情報テーブルを検索することにより、ロック処理対象の共用ファイルを探す。
ロック処理対象の共用ファイルに対応するレコードが見つからなかった場合、代行ロック処理要求部１０は、ロック処理要求元のプロセスに対しエラーメッセージを送り、上記共用ファイルが現在アクセスできない旨を通知する。

【００５８】

ロック処理対象の共用ファイルに対応するレコードが見つかった場合、ステップＳ６３において、当該レコード中の入出力処理要求元のプロセスのプロセス識別子と、ロック処理対象の共用ファイルのファイル識別子とが得られる。
さらに、代行ロック処理要求部１０は、ステップＳ６４において、代行ロック識別子と、入出力処理要求元のプロセスのプロセス識別子と、ロック処理対象の共用ファイルのファイル識別子とを使用して、共用ファイルのロック処理要求を行う。

【００５９】

最終的に、第２のプロセスＢは、共用ファイルのロック処理要求の結果をロック処理要求元のプロセスに通知する。
上記の識別子自動割り当て処理部および代行ロック処理要求部の動作は、コンピュータシステムのソフトウェアに基づいて提供されるプログラムにより実行することができるので、システム資源管理装置のハードウェア構成を複雑にすることなく、任意のロック処理要求元のプロセスの代行ロック処理を効率良く遂行することが可能になる。

【００６０】

【発明の効果】

以上説明したように本発明のシステム資源管理装置によれば、第１に、共用ファイル等の共用資源の入出力処理要求を受け付けて処理するプロセス等の実行体と、ロック処理要求を受け付けて処理するプロセス等の実行体とを互いに独立した実行体に分割して配置し、後者の実行体が前者の実行体を代行してロック処理要求をすることにより、各実行体の負担を軽くすることができる。この結果、効率の良い入出力処理およびロック処理が可能になり、システム全体の性能が向上する。

【００６１】

さらに、本発明のシステム資源管理装置によれば、第２に、このシステム資源管理装置に使用される識別子が、上記の入出力処理要求を受付処理する実行体を一意に識別できる実行体識別子と、この実行体により共用資源を一意に識別できるファイル識別子とから構成されているので、これらの実行体識別子およびファイル識別子を検索することにより、他の任意の実行体による共用資源のロック処理要求を効率良く処理することが可能になる。

【００６２】

さらに、本発明のシステム資源管理装置によれば、第３に、共用資源内のオフセットと長さを指定することにより共用資源をロック状態にし、このロック状態になった領域のみに対して他の実行体に対するロック処理を行っているので、他の実行体は、共用資源内の残りの領域を選択して使用することが可能になり、共用資源の使用効率が向上する。

【００６３】

さらに、本発明のシステム資源管理装置によれば、第４に、共用資源をオープン状態にするときの指定により、共用資源全体に対して他の実行体からのアクセスを制限することも

できるので、このような種類のロック処理を採用すれば、他の実行体のアクセスにより共用資源の内容が破壊されるおそれがなくなる。
さらに、本発明のシステム資源管理装置によれば、第５に、共用資源の入出力処理要求を受け付けて処理する実行体が複数存在する場合でも、これらの実行体を代行してロック処理を要求する機能を一つの独立した実行体により実行するようにしているので、代行ロック処理の構成を複雑にすることなく、任意の実行体による共用資源のロック処理を効率良く行うと共に、複数の他の実行体による入出力処理を効率良く行うことが可能になる。
【００６４】
さらに、好ましくは、本発明のシステム資源管理装置によれば、第６に、共用資源のロック処理処理要求を受け付けて処理する実行体に対し仮想的な識別子を自動的に割り当てることにより、共用資源の存在するコンピュータシステム以外の他のコンピュータシステムにおける実行体からの共用資源のロック処理の要求を受け付けるようにしているので、共用資源の存在するコンピュータシステム上の実行体と、ネットワークに接続された他のコンピュータシステム上の実行体との間のロック処理による排他制御が可能になり、ネットワーク上のすべてのコンピュータシステム上の実行体に対し同等なロック処理が行える。
【００６５】
また一方で、本発明のシステム資源管理方法によれば、第１に、共用資源の入出力処理要求を受け付けて処理するタイプの処理と、ロック処理要求を受け付けて処理するタイプの処理とを互いに独立した処理に分割し、後者の処理が前者の処理を代行してロック処理要求をすることにより、任意の処理によるロック処理をした後も、他の任意の処理による入出力処理要求が制限されることはなくなり、共用資源の入出力処理およびロック処理を並列して効率良く遂行することが可能になる。
【００６６】
さらに、本発明のシステム資源管理方法によれば、第２に、代行ロック処理に使用される識別子が、上記の入出力処理要求を受付処理するタイプの処理を一意に識別できる処理識別子と、上記タイプの処理により共用資源を一意に識別できるファイル識別子とを含んでいるので、これらの処理識別子（例えば、プロセス識別子）およびファイル識別子を検索することにより、他の任意の処理による共用資源のロック処理を効率良く代行することが可能になる。
【００６７】
さらに、本発明のシステム資源管理方法によれば、第３に、共用資源内のオフセットと長さを指定することにより共用資源をロック状態にし、このロック状態になった領域のみに対して他の処理に対するロック処理を行っているので、他のプロセスからも、共用資源内の残りの領域に対してアクセスすることが可能になり、複数種の処理、すなわち、複数のプロセスにより効率良く共用資源を使用することができる。
【００６８】
さらに、本発明のシステム資源管理方法によれば、第４に、共用資源をオープン状態にするときの指定により、共用資源全体に対して他の処理、すなわち、他のプロセスからのアクセスが制限されるので、このような種類のロック処理を実行することにより、共用資源の内容が破壊されるのを防止することが可能になる。
【００６９】
さらに、本発明のシステム資源管理方法によれば、第５に、共用資源の入出力処理要求を受け付けて処理するタイプの処理が複数存在する場合でも、これらの処理を代行してロック処理を要求する機能を一つの独立した処理により実行するようにしているので、任意の処理、すなわち、任意のプロセスによる共用資源のロック処理を容易にかつ効率良く行うと共に、複数の他の処理、すなわち、他のプロセスによる入出力処理を誤りなく遂行することが可能になる。
【００７０】
さらに、本発明のシステム資源管理方法によれば、第６に、共用資源のロック処理要求を受付処理するタイプの処理にて自動的に割り当てられた仮想的な識別子を使用することに

より、共用資源の存在するコンピュータシステム以外の他のコンピュータシステムにおける処理からの共用資源のロック処理の要求を受け付けるようにしているので、共用資源の存在するコンピュータシステム上の処理と、ネットワークに接続された他のコンピュータシステム上の処理との間で排他制御を利用したフレキシブルなロック処理を実行することが可能になる。

【図面の簡単な説明】

【図１】本発明の原理構成を示すブロック図である。

【図２】本発明の一実施例によるシステム資源管理装置の構成を示すブロック図である。

【図３】本発明の一実施例で使用する識別子情報テーブルを示す図である。

【図４】本発明の一実施例で使用する識別子自動割り当て処理テーブルを示す図である。

【図５】図２の識別子自動割り当て処理部の動作を説明するためのフローチャートである。

【図６】図２の代行ロック処理要求部の動作を説明するためのフローチャートである。

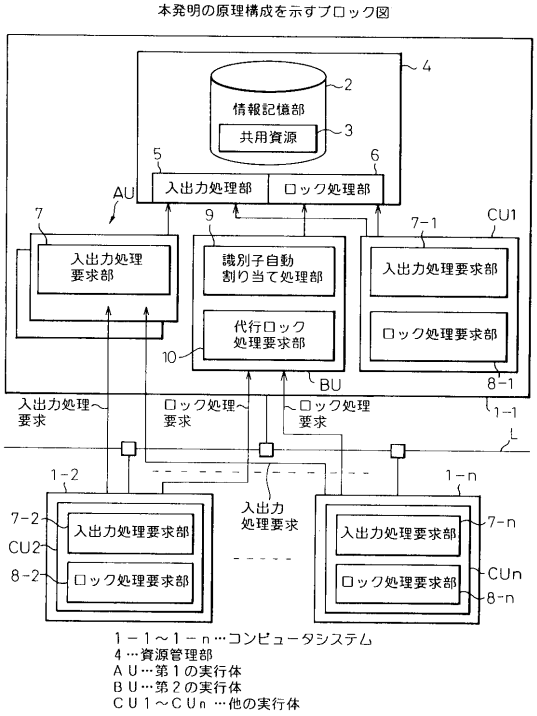
【図７】従来のシステム資源管理装置の第１の例を示すブロック図である。

【図８】従来のシステム資源管理装置の第２の例を示すブロック図である。

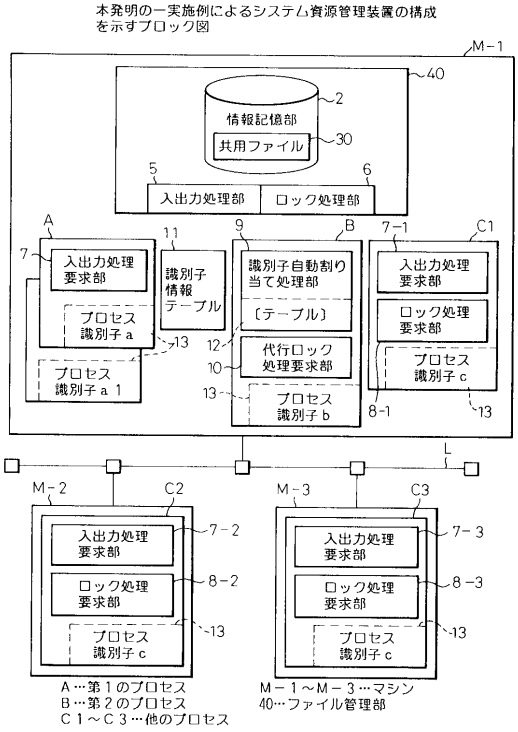
【符号の説明】

１－１～１－ｎ…コンピュータシステム
２…情報記憶部
３…共用資源
４…資源管理部
５…入出力処理部
６…ロック処理部
７…入出力処理要求部
８…ロック処理要求部
９…識別子自動割り当て処理部
１０…代行ロック処理要求部
１１…識別子情報テーブル
１２…識別子自動割り当て処理テーブル
１３…プロセス識別子
３０…共用ファイル
４０…ファイル管理部
ＡＵ…第１の実行体
ＢＵ…第２の実行体
Ａ…第１のプロセス
Ｂ…第２のプロセス
Ｍ－１～Ｍ－３…マシン

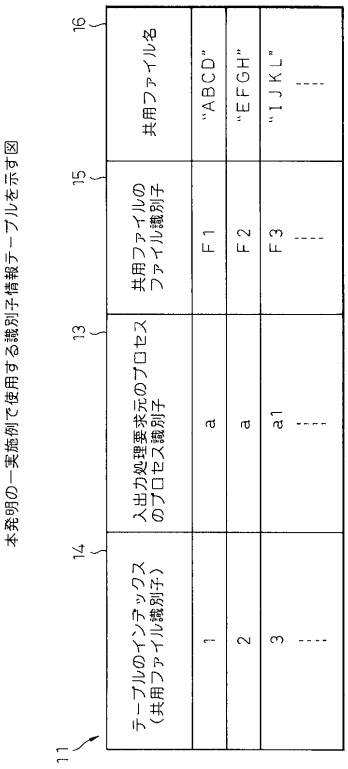
【図１】


本発明の原理構成を示すブロック図
情報記憶部
共用資源
入出力処理部
ロック処理部
AU
入出力処理要求部
識別子自動割り当て処理部
代行ロック処理要求部
CU1
ロック処理要求部
BU
入出力処理要求
ロック処理要求
1－1～1－n…コンピュータシステム
4…資源管理部
AU…第１の実行体
BU…第２の実行体
CU1～CUn …他の実行体

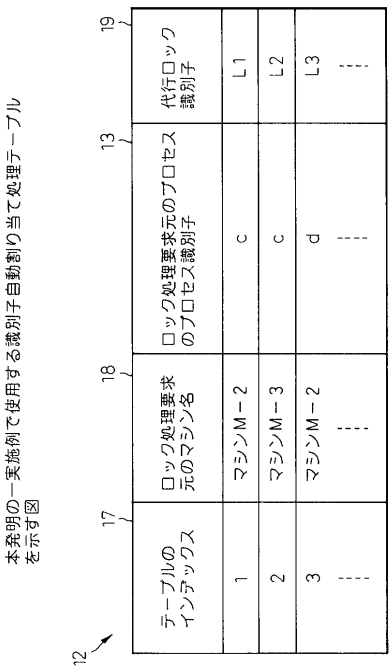
【図２】


本発明の一実施例によるシステム資源管理装置の構成を示すブロック図
情報記憶部
共用ファイル
入出力処理部
ロック処理部
入出力処理要求部
識別子情報テーブル
識別子自動割り当て処理部
〔テーブル〕
代行ロック処理要求部
ロック処理要求部
プロセス識別子a
プロセス識別子a1
プロセス識別子b
プロセス識別子c
A…第１のプロセス
B…第２のプロセス
C1～C3…他のプロセス
M－1～M－3…マシン
40…ファイル管理部

【図３】

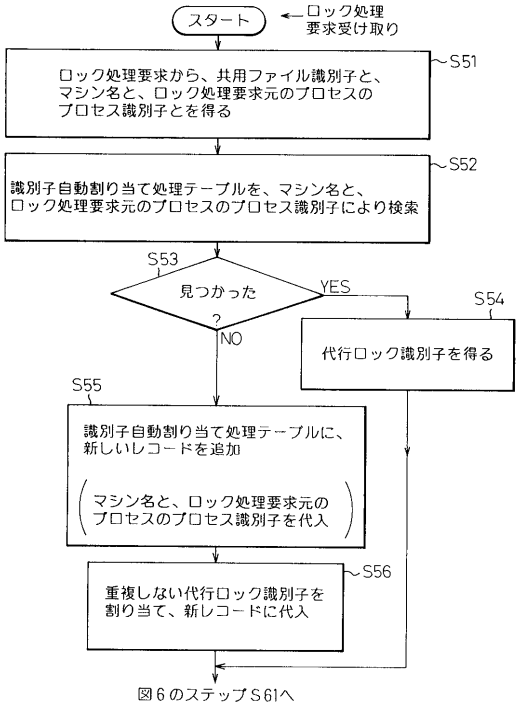

本発明の一実施例で使用する識別子情報テーブルを示す図
テーブルのインデックス（共用ファイル識別子）
入出力処理要求元のプロセス識別子
共用ファイルのファイル識別子
共用ファイル名
"ABCD"
"EFGH"
"IJKL"

【図４】


本発明の一実施例で使用する識別子自動割り当て処理テーブルを示す図
テーブルのインデックス
ロック処理要求元のマシン名
ロック処理要求元のプロセス識別子
代行ロック識別子
マシンM－2
マシンM－3
L1
L2
L3

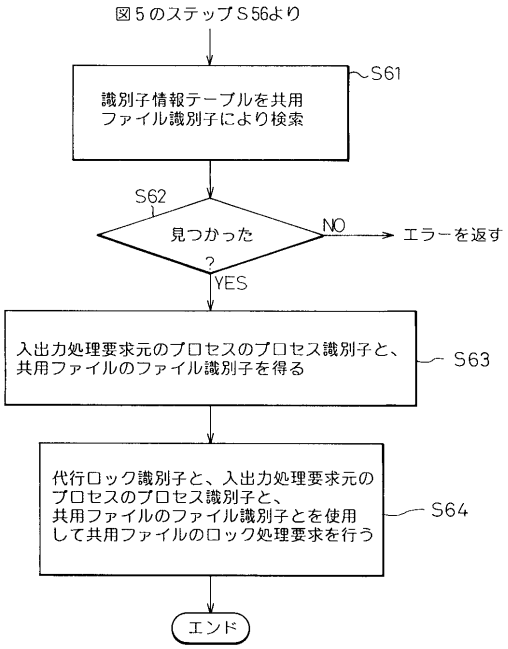
【図５】

図２の識別子自動割り当て処理部の動作を説明するためのフローチャート



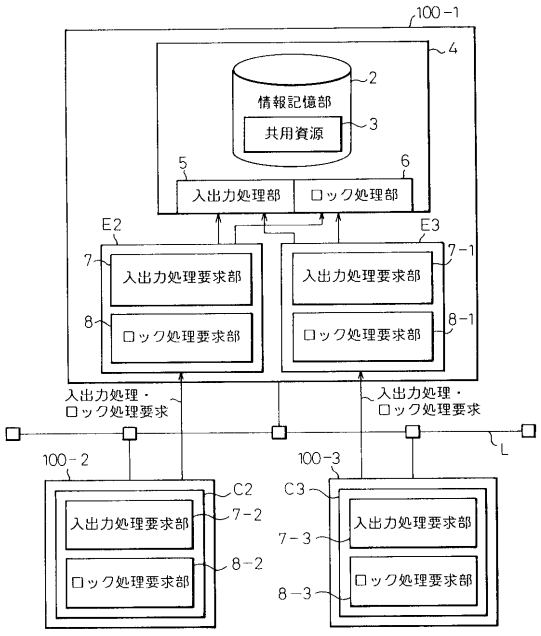
【図６】

図２の代行ロック処理要求部の動作を説明するためのフローチャート



【図７】

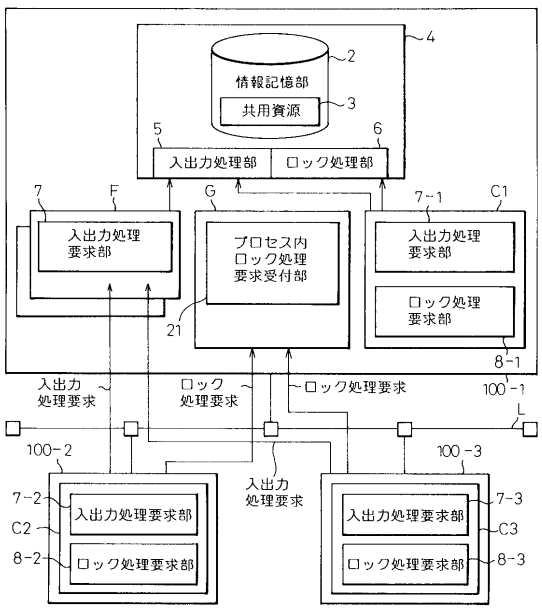
従来のシステム資源管理装置の第１の例を示すブロック図



100－１～100－３…コンピュータシステム
Ｅ２，Ｅ３…プロセス

【図８】

従来のシステム資源管理装置の第２の例を示すブロック図



Ｆ，Ｇ…プロセス

フロントページの続き

(72)発明者　原田　哲也
神奈川県川崎市中原区上小田中１０１５番地　富士通株式会社内
(72)発明者　荒井　悟
石川県河北郡宇ノ気町字宇野気ヌ９８番地の２　株式会社ピーエフユー内

審査官　相崎　裕恒

(56)参考文献　特開平０３－１４７１５１（ＪＰ，Ａ）
特開平０４－２６０９４７（ＪＰ，Ａ）
特開平０４－０１７０４１（ＪＰ，Ａ）
特開平０６－１６１８６１（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
G06F 12/00-13/00
G06F　9/46-54
G06F 15/16-177