B-47 Global File Virtual Memoryを用いたプロセスマイグレーション機能の設計
Design of Process Migration Mechanism using Global File Virtual Memory

佐藤 記代子† 矢野 浩邦† 前田 誠司†
Kiyoko Sato Hirokuni Yano Seiji Maeda

1 はじめに

近年の情報化社会において，情報システムの担う役割は非常に重要なものである．我々は，情報システムとしての役割を十分に果たす，簡単で柔軟な高信頼計算機クラスタシステムの研究を行っている．

プロセスマイグレーションは，計算機クラスタシステムにおける耐故障性の向上や動的負荷分散を実現するための技術である．しかし，プロセスマイグレーションの開始から完了までの間，プロセスが停止するため，プロセスマイグーレションを高速に行い，プロセス停止時間を短くすることが求められる．

そこで，我々は，プロセスマイグレーションを高速に行うために，Global File Virtual Memory(GFVM) を開発した．GFVM とは，システム内の全計算機ノードから透過的にアクセス可能であり，かつ，複数の計算機ノードのディスクに多重化されて存在するファイルを退避領域として使用する仮想記憶である．GFVM を用いることによって，プロセスマイグレーション処理を高速化すると供に，プロセスマイグレーション後のプロセス実行も高速化することが可能となる．

2 システム構成

図 1 に本実験システムの構成を示す．

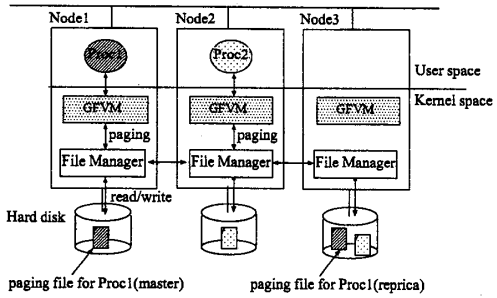

図 1: システム構成

本実験システムは，ネットワークで接続された複数の計算機ノードで構成され，各計算機ノードは，それぞれハードディスクを保持する．各計算機ノード上には，カーネルモジュールとして，Global File Virtual Memory(GFVM) と File Manager が存在する．File Manager は，ファイルを，システム内の全ノードから透過的なアクセスを可能にし，システム内の複数の計算機ノードのハードディスクに多重化して保存する [1].

†(株) 東芝 研究開発センター コンピュータ・ネットワークラボラトリー
Computer & Network Systems Laboratory, Corporate Research & Development Center,Toshiba Corporation

3 従来のプロセスマイグレーションとその問題点

従来のプロセスマイグレーションの手順は次のようになる．

プロセス移送元の計算機ノード
　プロセスを停止し，プロセス状態を取得して，プロセス移送先計算機ノードへ取得したプロセス状態を移送する．

プロセス移送先の計算機ノード
　受け取ったプロセス状態を元に，プロセスの複製を作成し，複製プロセスを実行する．

プロセスマイグレーションにおけるプロセスの停止時間は，移送元計算機ノードでの停止後，移送先計算機ノードで複製プロセスを実行するまでの時間である．プロセス状態には，プロセスのアドレス空間の全内容,CPU レジスタ値，プロセスが使用していたファイル情報等が含まれる．このうち，アドレス空間のサイズは非常に大きく，移送に時間がかかる．そのため，プロセスの停止時間が長いという問題がある．長時間のプロセス停止は，ユーザに影響を及ぼすため，迅速なサービス提供が求められる情報システムでは，致命的な問題となる．

4 Global File Virtual Memory 用いたプロセスマイグレーション

本研究は，プロセスマイグレーションに適した仮想記憶である Global File Virtual Memory を用いることで，プロセスマイグレーション時のアドレス空間の移送時間を短縮すると供に，プロセスマイグレーション後のプロセス実行を高速化する．

4.1 Global File Virtual Memory

Global File Virtual Memory(GFVM) は，退避領域として，各プロセス毎に個別のファイル (ページングファイル) を使用する仮想記憶である．GFVM は，通常のプロセス実行に伴うページングの他に，プロセスマイグレーション時にもページアウトを行う．GFVM によってページアウトされるページは，ページインされてから変更のあったページ (ダーティーページ) である．

図 1 は,Node1 でプロセス Proc1 を実行している時の GFVM と File Manager，そして，ページングファイルの関係を表す．GFVM は,File Manager を通じて Proc1 用のページングファイルに対してページングを行う．File Manager は，Proc1 用のページングファイルのマスターを Node1 のローカルディスク上に保持すると供に,Node3 のディスク上に Proc1 用のページングファイルのレプリカを保持する．

本研究では，プロセスのページングファイルのレプリカを保持する計算機ノードの中から，プロセスの移送先計算機ノードを選択する．この結果，移送後のプロセスとページングファイルは同一の計算機ノードに存在するので，プロセスマイグレーション

後のページングは，ローカルディスクへのアクセスとなり，ページングを高速化できる．

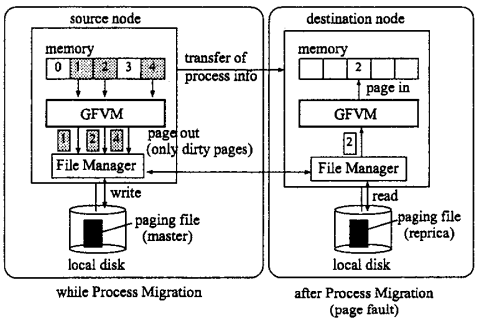

図 2: GFVM による Process Migration 手法

4.2 プロセスマイグレーション処理

図 2 の左図は，プロセスマイグレーション時のプロセス移送元計算機ノードでの GFVM の動作を表す．移送するプロセスのアドレス空間の内容のうち，ページ 0,1,2,3,4 がメモリ上にページインされており，ページ 1,2,4 はダーティーページである．

移送元計算機ノードでは，プロセスを停止した後,GFVM が，ダーティーページ 1,2,4 のみを,File Manager を介してプロセス固有のページングファイルにページアウトする．File Manager は，ページアウトされたダーティーページ 1,2,4 の内容を，ローカルディスク上のページングファイルに書き，並行して，他計算機ノードのディスク上にあるレプリカのページングファイルにも同様に書く．

本研究によると，プロセスマイグレーション時のアドレス空間の移送時間は，ダーティーページのページアウト時間となる．従って，全アドレス空間を移送先計算機ノードへ移送する場合に比べて，移送時間を短縮することができる．

さらに，GFVM が，プロセス実行中の通常処理として，更新頻度の少ないダーティーページを定期的にページアウトすることで，プロセスマイグレーション時のページアウトを削減することも可能となる．

4.3 プロセスマイグレーション後のプロセス実行

本研究では，前述の通り，プロセスの移送先計算機ノードを選択する際に，プロセスのページングファイルのレプリカを保持する計算機ノードから選択する．この結果，移送後のプロセスとページングファイルは同一の計算機ノードに存在する．

図 2 の右図は，プロセスマイグレーション後にプロセスを実行中の，プロセス移送先計算機ノードの動作である．プロセス実行中にページフォルトが発生し，ページ 2 をページインする際，GFVM は,File Manager を介して，ローカルディスク上のページングファイルからページ 2 をページインする．

本研究によると，プロセスマイグレーション後のページングは，常にローカルディスクに対して行うことになり，プロセスマイグレーション後のプロセス実行を高速化することができる．

また,File Manager がレプリカファイルを作成した際のファイルキャッシュが残っていれば，ローカルディスクへアクセスする必要もなく，ページング処理を高速化できる可能性も高い．

5 関連研究

プロセスマイグレーションの高速化を目的とした研究には，PreCopy[2],Copy On Reference[3], そして，Flushing[4] 等がある．

PreCopy は，プロセス実行中に，プロセス移送先計算機ノードへプロセス状態を全て移送するため，プロセス実行中の通信負荷が高くなる．また，プロセス移送先計算機ノードを事前に決定する必要があり，動的な負荷分散に柔軟に対応できない．

Copy On Reference は，移送先計算機ノードでプロセスの実行を再開した後に，移送元計算機ノードからページインを行うため，移送元計算機の停止や故障の際に，プロセスの実行を継続できない．

Flushing は，プロセスマイグレーション時に，移送元計算機ノードでダーティページのみを File Server に対してページアウトし，移送先計算機ノードで File Server からページインする．File Server からのページインは，ネットワークを介してリモートディスクからページを読み込むことになるので，ページイン処理に時間がかかり，プロセスマイグレーション後のプロセス実行に影響を与えてしまう．

6 まとめ

本稿では，プロセスマイグレーション処理を高速化すると供に，プロセスマイグレーション後のプロセス実行を高速化する，プロセスマイグレーション機能の設計について述べた．本研究は，プロセスマイグレーション時に,GFVM のページング機能を用いて，プロセスのアドレス空間のダーティーページのみを，プロセス固有のページングファイルへページアウトする．従って，プロセスの全アドレス空間を移送する場合に比べて，プロセスマイグレーションを高速に行うことが可能になる．そして，ページングファイルのレプリカを保持する計算機ノードから，プロセスの移送先計算機ノードを選択するため，プロセスマイグレーション後のページング処理は，常にローカルディスクに対して行われる．従って，プロセスマイグレーション後のプロセス実行を高速に行うことが可能になる．

GFVM を用いたプロセスマイグレーション機能は、現在、実験システム上での実装を行っている。実装を終え次第、プロセスマイグレーションの処理時間を測定し、評価を行う予定である．

参考文献

[1] 矢野他：“ シングルイメージクラスタシステムにおける高信頼ストレージ機能の設計 ”,FIT2002(2002)

[2] M.Theimer, K.Lantz, and D.Cheriton: “ Preemptable remote execution facilities for the V-system ”, In 10th ACM Symposium on Operating Systems Principles,pp.2-12(1985)

[3] E.Zayas: “ Attacking the process migration bottleneck ”, In 11th ACM Symposium on Operating System Principles,pp.13-22(1987)

[4] F.Douglis and J.Ousterhout: “ Transparent process migration: Design alternatives and the Sprite implementation ”, Software-Practice and Experience,21(8),pp757-786(1991)