WIDE Technical-Report in 2009

NEMO BSを用いたXenゲスト計算機のオフリンクライブマイグレーション

wide-tr-keiichi-xen-offlink-migration-00.pdf

WIDE Project : http://www.wide.ad.jp/

If you have any comments on WIDE documents, please contact to ad@wide.ad.jp

Title: NEMO BS を用いた Xen ゲスト計算機のオフリンクライブマイグレーション

Author(s): 島 慶一 (keiichi@iijlab.net)

Date: 2009-9-1

NEMO BSを用いたXenゲスト計算機のオフリンクライブマイグレーション

島 慶一*

1 背景

計算機の処理能力、ネットワークやデータストアなどの技術進歩により、個々の計算機の能力が飛躍的に進歩している。ただ、単体技術の発展速度は今後鈍化するという見方もあり、複数の計算機をひとつの計算機資源として取り扱うクラウド技術への期待が高まっている [1, 2]。一方、ひとつの計算機資源を仮想的に分割し、複数の異なる計算機として利用する仮想計算機技術も長年研究されてきた [3]。一見異なる方向を目指しているように見えるこの二つの技術だが、お互いに補い合うことでより効率的な計算機資源の活用を実現できると考えられている。Google が展開している Google App Engine[4] の仕組みや、Amazon が提供している EC2[5] などは、複数の計算機資源を組み合わせてひとつのサービスとするクラウド環境を提供しているが、ひとつひとつの計算機資源は仮想計算機技術を用いられている。一台の物理的な計算機、という大きな単位での資源管理をやめ、小さく切り出された仮想計算機をひとつの単位として利用することで、結果的に効率的かつ柔軟な計算機資源の利用を実現しつつ、仮想計算機に分割したことによる負荷を上回る計算能力をクラウド環境として提供している。

このような仮想環境が発展していくためには、仮想計算機を柔軟に管理する仕組みが重要となる。クラウド環境からの要求に従って、必要な場所に必要な量の仮想計算機を配置できることが、全体の性能向上や資源の効率利用に貢献するためである。本論文では、仮想計算機の再配置の仕組みに注目し、稼働中の仮想計算機を異なるネットワーク (オフリンクネットワーク) に移動させる仕組みを提案する。

*株式会社 IIJ イノベーションインスティテュート 技術研究所

2 ライブマイグレーションの課題

現在、多くの仮想計算機技術が実用レベルで提供されており、その中には仮想計算機を、その親となっている計算機 (ホスト計算機) から別のホスト計算機に移動させる機能を提供しているものもある。VMware の VMotion や、Xen の Live Migration がその代表例である。以後、本稿ではホスト計算機から切り出された仮想計算機をゲスト計算機、ゲスト計算機のホスト計算機間移動機能をライブマイグレーションと呼称する。ライブマイグレーション機能を利用すると、稼働中のゲスト計算機をほぼ無停止で別の機材に移動できる。ただし、現在提供されている機能は、ホスト計算機の配置に制限があり、移動元のホスト計算機と移動先のホスト計算機が同一のサブネットワークに属している必要がある。

この制限は、ホスト計算機とゲスト計算機のネットワーク構成に原因がある。仮想計算機環境で稼働する各計算機は対等な関係ではない。通常はホスト計算機がすべての資源を管轄し、ゲスト計算機に配分する形式となっている。ゲスト計算機をネットワークに接続する場合、図 1 に示すような構成が用いられる。図 1 (a) の場合、ゲスト計算機はホスト計算機と同じネットワークに、ホスト計算機が提供する仮想ブリッジを経由して接続される。(b) の場合、ホスト計算機とゲスト計算機の間に仮想的なスイッチが設けられ、ゲスト計算機はホスト計算機を上流ルー

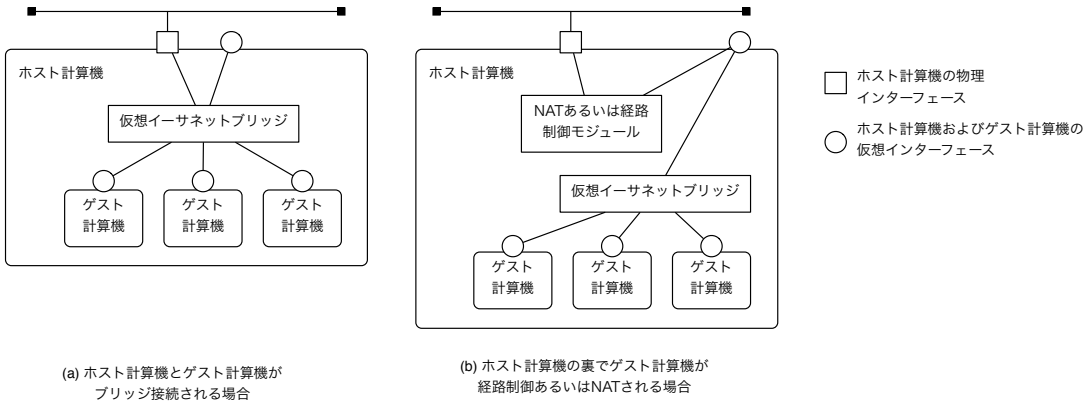

図 1: ゲスト計算機のネットワーク環境

タとしてネットワークに接続する。この構成から明らかなように、ホスト計算機がどのようにネットワークに接続しているかが、ゲスト計算機のネットワーク構成に大きく関わることになる。ライブマイグレーションが発生するとき、ゲスト計算機が動作環境を変更することはないため、実質、図 1 (a) の構成でしかライブマイグレーションは実現できない。それ以外の構成の場合、例えば図 1 (b) の構成の場合、ゲスト計算機が他のホスト計算機に移動した瞬間、ゲスト計算機が接続している仮想イーサネットブリッジで用いられているネットワークアドレスブロックが変化するため、ゲスト計算機のネットワーク環境を適切に変更しない限り通信を継続することができなくなる。また、図 1 (a) の構成であっても、移動元と移動先のホスト計算機が異なるサブネットワークに接続している場合は同様の問題が発生する。

ライブマイグレーションによって、仮想計算機がひとつのホスト計算機に集中することを回避できるが、同一サブネットワークに限られてしまい、他の位置に資源に余裕のあるホスト計算機が配置されていても活用ができない。また、計算資源を提供するクラウド環境と計算資源を活用するクライアントの間の通信遅延を抑えるためには、クラウド環境とクライアントがネットワーク的に近いのが理想である。クラウド環境の一部としてゲスト計算機を利用する場合、ゲスト計算機をより近いホスト計算機に移動させたいという要求が発生したとしても、前述した理由によりネットワークを越えるゲスト計算機の移動ができないのが現状である。

3 NEMO BS の概要

NEMO Basic Support (NEMO BS)[6] は IPv6 ルータに移動通信機能を追加する。NEMO BS 環境では、NEMO BS をサポートした移動ルータ (MR) が不変のネットワークプレフィックス (モバイルネットワークプレフィックス、MNP) を管理する。MR が提供するサブネットワークに接続した IPv6 ホストは、MNP の範囲からアドレスを取得する。MR はインターネット上の様々なサブネットワークに接続し、接続先のネットワーク環境に応じたネットワーク構成に動的に対応するが、MR が管理する MNP が変化することはない。MR 配下のホストは MR の位置に関わらず常に同じネットワーク環境を維持できる。

移動透過性は、MR と対をなして動作するホームエージェント (HA) の助けをかりて実現されている (図 2)。MR は移動先のネットワークで、ネットワーク環境に応じたアドレス (気付アドレス) を取得し、HA との間に双方向 IPv6 over IPv6 トンネルを確立する。すべての MNP からのトラフィックは、そのトンネルを使って一旦 HA に転送され、そこから再

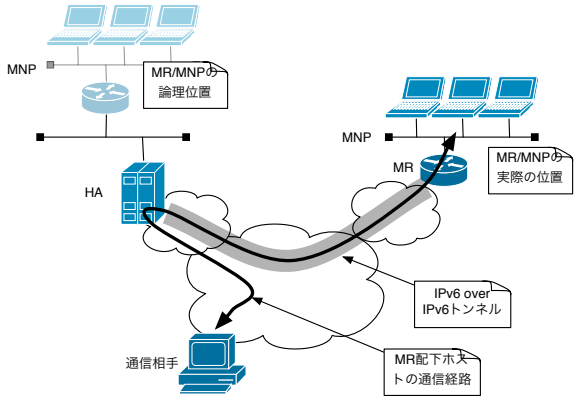

図 2: NEMO BS 動作の概要

転送される。逆に、MNP 宛のトラフィックは HA が一旦受け取り、トンネルを介して MR に配送され、その後最終宛先ホストに転送される。

4 設計

2 章で述べた通り、ライブマイグレーションが同一サブネットワーク内での移動に限定されるのは、ゲスト計算機のネットワーク環境がホスト計算機に依存しているためである。よって、ゲスト計算機のネットワーク環境がホスト計算機によらず一定になるように設計すれば、異なるサブネットワークに接続しているホスト計算機間でもゲスト計算機を移動することができるようになる。一定のネットワーク環境を提供する手段として、本論文では IP モビリティ技術を用いる。IP モビリティ技術を使ってゲスト計算機のネットワーク環境を一定に保つ方法として、ゲスト計算機自体が Mobile IP など [7, 8] のホスト移動通信機能を備える方式と、ホスト計算機が NEMO BS などを活用して透過的なネットワークをゲスト計算機に提供する方法が考えられる。前者は、ゲスト計算機の改変 (IP モビリティ機能の導入) が必要になる代わりに、ゲスト計算機単体での移動が可能になり、計算機移動の細かい制御が期待できる。後者は、これまで利用してきたゲスト計算機がそのまま継続利用できる代わりに、ゲスト計算機の移動

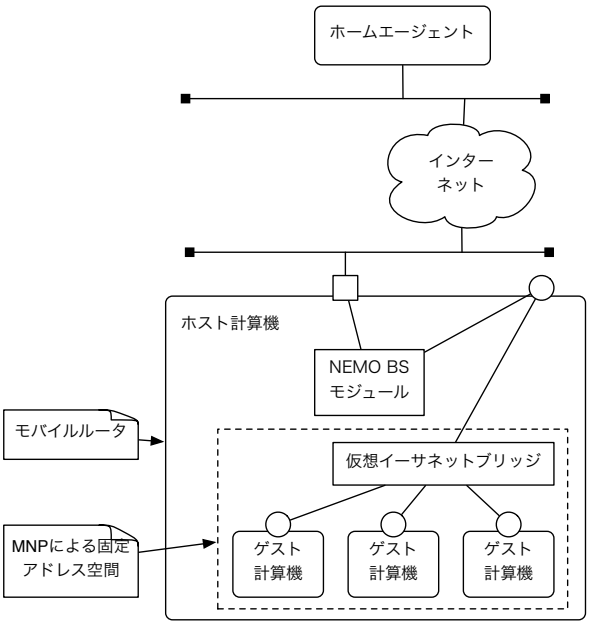

図 3: Xen を用いたゲスト計算機オフリンクマイグレーション設計

とホスト計算機の NEMO BS 移動ルータとしての移動が同期しなければならないという制限が発生する。今回は、既存のシステムで用いられているゲスト計算機をそのまま利用し続けるというシナリオを前提として、後者の方式を選択した。

システムの設計はホスト計算機の資源管理の方法に依存する。そのため、システム構成に使用する仮想計算機環境毎に異なる可能性がある。本論文では、Xen[9] が提供する仮想計算機環境を用いた設計を提示するが、他のシステムでも大きな変更は必要ないと考えている。図 3 に概要を示す。構成的には図 1 (b) を拡張したものとなっている。ホスト計算機が MR としての機能を有し、インターネット接続用と、MNP 提供用のふたつのインターフェースを持つ。MNP 接続用インターフェースは、実際には物理的なインターフェースではなく、ホスト計算機が提供するゲスト計算機用仮想ブリッジに接続された仮想インターフェースとして構成する。仮想インターフェースおよび仮想ブリッジに接続されたホストには、MR が管理する MNP 空間のアドレスが割り当

てられる。NEMO BS の機能により、ホスト計算機が物理的にどこに接続されていても、MNP 内のアドレスが変更されることはない。

ゲスト計算機のライブマイグレーションのために、複数のホスト計算機がネットワーク上に配置される。これらのホスト計算機は、NEMO BS の MR として同じ設定 (ホームアドレス、MNP) を保持するが、稼働中のゲスト計算機を有していない場合は MR としては動作しない。ゲスト計算機がマイグレーションする場合、通常のライブマイグレーションの手順を用いて、各ゲスト計算機がマイグレーション先のホスト計算機に移動する。この時点では、まだゲスト計算機はネットワークから切り離された状態にある。その後、マイグレーション元のホスト計算機がマイグレーション先のホスト計算機に移動完了通知を送り、自身の NEMO BS 機能を停止する。移動完了通知を受けたホスト計算機は、HA に対して現在位置を登録し、NEMO BS の MR として動作を開始する。HA への登録が完了した段階で、ゲスト計算機が接続している仮想ブリッジの MNP が有効となり、マイグレーションが完了する。

5　実験による検証

提案した設計が実現可能であることを確認するため、プロトタイプを実装して稼働実験を実施した。

5.1　実験環境

実験環境は、ホームエージェント、NEMO ルータ機能を備えた 2 台の Xen ホスト計算機、テスト用のストリーミングデータを受信する計算機の合計 4 台の計算機で構成される。それぞれの計算機はインターネットに接続され、相互に通信が可能な状態となっている (図 4)。Xen ホスト計算機には 1 台のゲスト計算機が構築され、ストリーミングサーバとして動作する。

オペレーティングシステムは Debian Linux 5.0、CentOS 5.2 が用いられ、必要な拡張機能およびソフトウェアが導入されている。表 1 にソフトウェア情報を示す。

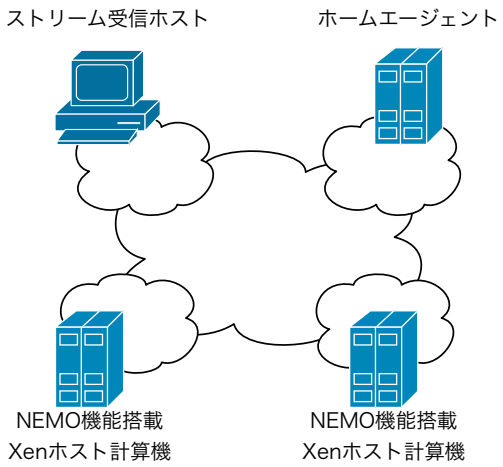

図 4: 実験環境概要

実験環境をより現実の環境に近づけるため、上述の機材は実際のインターネット環境上に構築した。ホームエージェント、およびホームエージェントが管理するモバイルネットワークは Interop Tokyo 2009[10] のために構築された運用ネットワークの一部に設置し、Xen ホスト計算機とストリーム受信ホストは IIJ のネットワーク上に配置された。それぞれの機材は Internet を介して相互接続されている。図 5 にトポロジを示す。

ストリーム送信ホスト (ゲスト計算機) からは、15M バイト、520K ビット/秒の MPEG4 ストリームデータを UDP を使って継続的に送信した。ゲスト計算機のライブマイグレーション、およびホスト計算機の NEMO 移動ルータ機能の停止作業および開始作業は別途用意した制御用計算機から ssh を用いた遠隔コマンド実行によって操作した。ホスト計算機が接続しているネットワークの IPv6 ルータ広告の間隔は、IPv6 で定義されている最小間隔 (3 から 4 秒) に設定されている。

制御用計算機からは、ゲスト計算機のライブマイグレーションを実行し、ライブマイグレーションが完了した直後にホスト計算機の移動処理を開始する

計算機	導入ソフトウェア
ホームエージェント	Debian Linux 5.0、NEPL
NEMO 機能搭載 Xen ホスト計算機	Debian Linux 5.0、NEPL、Xen 3.2
Xen ゲスト計算機	Debian Linux 5.0、Video Lan Server
ストリーム受信ホスト	CentOS 5.2、Video Lan Client

表 1: 利用したソフトウェア

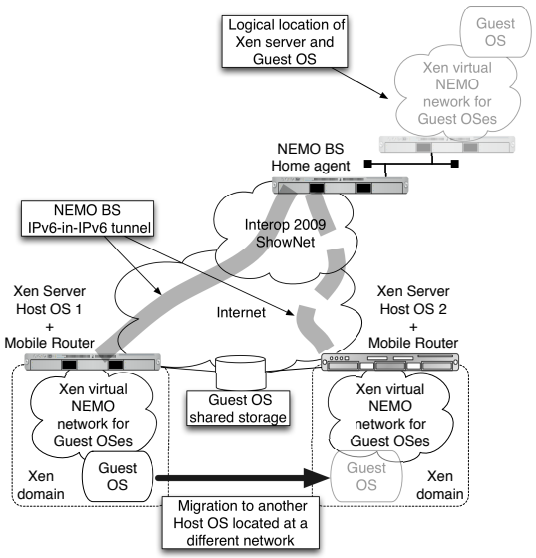

図 5: 実験ネットワークトポロジ

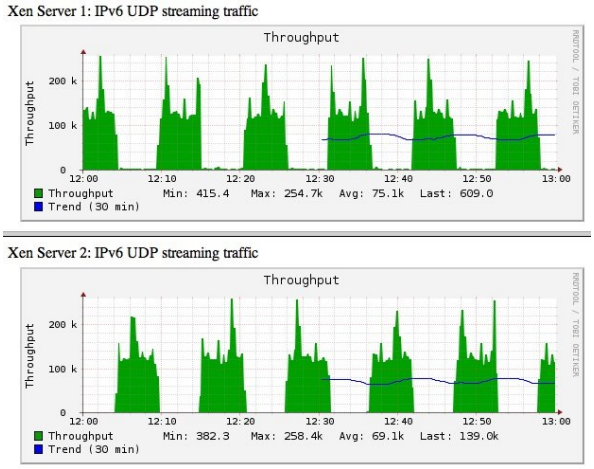

図 6: ストリームトラフィックのの推移

スクリプトを準備し、5 分間隔で二つの計算機間の移動を繰り返した。

6 評価と課題

ストリームトラフィックをホスト計算機の外足 (インターネット側インターフェース) でモニタした結果を図 6 に示す。5 分毎にゲスト計算機がライブマイグレーションされ、同時にホスト計算機が NEMO 移動ルータとして移動しているため、トラフィックグラフが 5 分毎にふたつのホスト計算機の間を移動しているのが確認できる。

ライブマイグレーションでは、仮想計算機のデータを複製している段階でも複製元の仮想計算機が動作し続けるため、切り換え時に瞬間的に停止 (数百ミリ秒) するだけで動作を継続できる。しかし、今回の構成ではホスト計算機をネットワーク層の技術を用いて移動させているため、ゲスト計算機のマイグレーションの後のホスト計算機の移動処理が開始される。観測したトラフィックデータから、片方のホスト計算機がストリームデータの配信を停止し、移動先のホスト計算機がストリームデータを送信し始めるまでに 6 秒程度の時間がかかる事が判明した。

ホスト計算機が接続したネットワークの IPv6 ルータ広告の間隔は 3 から 4 秒となっているので、ホスト計算機は平均すれば 1.75 秒でルータ広告を受信できる環境にある。NEMO BS の処理の一部である、気付アドレスの重複確認に 1 秒かかることを考慮すると、この環境では通常 3 秒程度で移動検出が可能となる。今回、それ以上に時間がかかっているのは以下の 2 つの理由によると考えられる。

1. 制御計算機からの遠隔操作スクリプトの実行オーバヘッド

2. 移動の手順が通常の移動と異なるため

NEMO 移動ルータ機能の停止および開始が ssh による遠隔スクリプト実行によって実現されているため、TCP 接続のための時間、ホスト計算機でのプロセス生成時間、遠隔コマンドによる NEMO 機能デーモンプログラムの制御などの時間が通常の運用と比較して余分に必要となる。また、今回移動処理が単一の移動ノードで実行されるのではなく、二つの異なる移動ノードで実行されている。すなわち、一方の移動ルータからホームエージェントへの登録解除処理を完了した後に、もう一方の移動ルータの登録処理を実行している。同一移動ルータで移動処理を実行する場合は、登録解除処理は不要で、連続的に登録処理が可能であるため、その分余分な時間がかかっている。

7 考察

本実験は、二つの技術 (NEMO BS と Xen 仮想化) の組み合わせ検証といえる。本節では検証の過程で得た知見を述べる。

7.1 ネットワークストレージの問題

Xen のような仮想計算機環境を提供する仕組みでは、ゲスト計算機のストレージを提供する方法として、ホスト計算機のストレージの一部を提供する方式と、ネットワークストレージを提供する方式の二種類が考えられる。ライブマイグレーションを利用する場合、ゲスト計算機が稼働するホスト計算機が変わるので、ストレージはネットワーク上に配置する必要がある。現在想定されている利用方法では、ゲスト計算機は同一リンクを越えて移動することがないため、ネットワークストレージを利用しても移動前後に大きな環境変化は発生しない。しかし、今回の実験のように、リンクを越えて移動する場合、ネットワークストレージへの到達性、応答性などが問題になる可能性がある。ひとつの解決策としては、外部ストレージを用いずに、すべてのメモリ上で処理するゲスト計算機環境を構築する方法が考えられるが、この方式ではメモリ上に保持するデータ量が増えるに従って、マイグレーション完了までの時間が増加する。ライブマイグレーションを使うことで、ほぼ無停止でのマイグレーションができるものの、移動を指示してから実際に移動が完了するまでの時間は長くなる。別の方法は、インターネット上複数地点から効率的かつ透過的に扱えるストレージを提供する方法である。簡単なものでは、ストレージのミラーリング技術を発展させ、複数拠点から同一のストレージデータにアクセスさせることで、ゲスト計算機に対する変化を抑えるとともに、ストレージの耐障害性の向上も計ることができると思われる。

7.2 利便性の問題

今回ルータ移動技術として NEMO BS を用いたが、この方式には二つほど問題点がある。まず一つ目は、NEMO BS を含む Mobile IPv6 系の移動通信技術一般に言える事であるが、ホームエージェントを介したトンネル通信が前提となってしまう点が挙げられる。ホームエージェントが一点障害になるため、ホームエージェントの多重化技術を導入しなければならないなど、別途考慮する点が発生する。二つ目は、NEMO BS を利用したことにより、すべてのゲスト計算機が同一の移動ネットワークのノードとして管理されるため、ホスト計算機が移動する場合は、関連するすべてのゲスト計算機群も同時に移動しなければならない点である。これは、ゲスト計算機がホスト単位の移動通信技術、例えば Mobile IPv6 を採用することでも解決できるが、この場合すべてのゲスト計算機で対応が必要となり、導入の敷居が高い。NEMO BS を利用すると、ゲスト計算機変更が不要になり、Xen で利用する標準のゲスト計算機をそのまま利用できるという利点もある。

一つ目の問題はトンネルを用いない移動通信技術を採用することで解決できる可能性がある。例えば、

HIP[11] や LIN6[12]、MAT などを基礎とした移動ルータを用いる事が考えられる。仮想計算機のオフリンクライブマイグレーションを実現するための条件は、仮想計算機が接続する仮想ブリッジのネットワーク環境の維持であるため、それを実現する方法が NEMO BS である必然性はない。

二つ目の問題は、以下のいずれかの方法で対応可能と考えられる。まず、すでに述べたようにゲスト計算機自体が Mobile IPv6 などの移動通信プロトコルを実装すること、もうひとつは、ゲスト計算機毎に NEMO BS 環境を提供することである。後者の案は、一種の Mobile IPv6 プロキシのような運用になる。現在の実装では後者を実現することはできないが、独立したモバイルネットワークプレフィックスをひとつの計算機で独立して運用することは理論上可能である。ゲスト計算機の変更を避けつつ、ゲスト計算機単位でのマイグレーションができる利点がある。

7.3 実装上の問題

今回の実験では実装上の制限に起因すると思われる問題にも遭遇した。

まず、仮想化環境のために利用した Xen 3.2 では、仮想計算機識別子 (VMID) の上限問題が発生した。VMID は、仮想計算機が生成されたときに割り当てられる、ホスト計算機上でのユニークな数値である。新しく生成された仮想計算機には、過去に利用された最大の VMID+1 の値が割り当てられるので、VMID は単調に増加する。ライブマイグレーションが実行されると、移動先のホスト計算機では、移動してきた仮想計算機に対して新しい VMID を割り当てる。しかし、Xen では VMID の上限が 128 となっており、これを越えるとライブマイグレーションに失敗する。

ライブマイグレーションを利用するためには、ゲスト計算機のストレージ (起動パーティションを含む) をすべてネットワークストレージとして運用しなければならない。今回 Linux オペレーティングシステムが提供している、NFS ルートパーティションからの OS 起動の仕組みを利用したが、Linux 用 NFS はまだ IPv6 をサポートしていない。逆に、NEMO BS の実装として利用した NEPL は IPv6 通信しかサポートしていない。NFS の IPv6 化、NEMO BS の DSMIPv6[13] 対応などで、IP バージョン非依存の環境へ移行していく必要がある。

8 まとめ

今後到来するクラウド環境において、計算機資源の流動性確保が重要な技術となる。仮想計算機のライブマイグレーション技術は、そのような技術の候補である。現在のライブマイグレーション技術では、移動先が同一ネットワーク内の別のホスト計算機に限定されている。これを別リンク、遠隔のネットワークに移動できれば、計算機資源をより流動的に管理できる。本論文では、仮想化技術と IP 移動通信技術を用いて、リンクを共有していない複数のホスト計算機間でゲスト計算機を移動させる手法を提案した。ゲスト計算機の移動が同一ネットワークに限定されるのは、ホスト計算機が提供するネットワーク環境が、ホスト計算機が接続している物理ネットワークに依存するためである。IP 移動通信技術を使い、ホスト計算機の接続ネットワークに関わらず、常に固定のネットワーク環境をゲスト計算機に提供することで、リンクを越えたマイグレーションを可能にした。提案技術は実際のネットワーク環境を用いて運用され、手法が実現可能であることを示した。

謝辞

本研究を進めるにあたり、中村雅英氏、Jean Lorchat 氏、Martin André 氏から Linux および MIPL/NEPL の構成について多くの助言をいただきました。また、実験環境の準備に尽力していただいた三宅喬氏、織学氏および Interop Tokyo 2009 NOC チームのみなさまに感謝いたします。

参考文献

[1] Aaron Weiss. Computing in the clouds. *netWorker*, 11(4):16–25, December 2007.

[2] Brian Hayes. Cloud computing. *Communications of the ACM*, 51(7):9–11, July 2008.

[3] Paul Barham, Boris Dragovic, Keir Fraser, Steven Hand, et al. Xen and the art of virtualization. In *SOSP '03: Proceedings of the nineteenth ACM symposium on Operating systems principles*, pages 164–177. ACM, 2003.

[4] Google. Google App Engine, August 2009. http://appengine.google.com/.

[5] Amazon. Amazon Elastic Compute Cloud (Amazon EC2), August 2009. http://aws.amazon.com/ec2/.

[6] Vijay Devarapalli, Ryuji Wakikawa, Alexandru Petrescu, and Pascal Thubert. *Network Mobility (NEMO) Basic Support Protocol*. IETF, January 2005. RFC3963.

[7] Basavaraj Patil, Phil Roberts, and Charles E. Perkins. *IP Mobility Support for IPv4*. IETF, August 2002. RFC3344.

[8] David B. Johnson, Charles E. Perkins, and Jari Arkko. *Mobility Support in IPv6*. IETF, June 2004. RFC3775.

[9] Citrix, August 2009. http://www.xen.org/.

[10] Interop Tokyo 2009, June 2009. http://www.interop.jp/.

[11] Robert Moskowitz, Pekka Nikander, Petri Jokela, and Thomas R. Henderson. *Host Identity Protocol*. IETF, April 2008. RFC5201.

[12] Mitsunobu Kunishi, Masahiro Ishiyama, Keisuke Uehara, Hiroshi Esaki, and Fumio Teraoka. LIN6: A New Approach to Mobility Support in IPv6. In *Wireless Personal Multimedia Communication (WPMC)*, November 2000.

[13] Hesham Soliman. *Mobile IPv6 Support for Dual Stack Hosts and Routers*. IETF, June 2009. RFC5555.