

科学研究費助成事業（科学研究費補助金）研究成果報告書

平成 25 年 5 月 31 日現在

機関番号：37102

研究種目：研究活動スタート支援

研究期間：2011～2012

課題番号：23800068

研究課題名（和文）

配信能力を拡張可能な自律分散型配信システムに関する研究

研究課題名（英文） Research on autonomous distributed contents delivery system extended contents delivery capacity

研究代表者

九州産業大学・情報科学部・助手

神屋 郁子 (KAMIYA YUKO)

研究者番号：00615354

研究成果の概要（和文）：

クラウドコンピューティングの普及により、計算機システムの処理能力を向上することが可能となった。しかし、クラウドコンピューティングを利用して向上可能なのは計算能力など局所的な処理能力のみであり、ネットワーク性能の向上は困難である。本研究では、複数のクラウドを組み合わせ利用可能な、サーバ広域分散配置システムを提案する。これによりネットワーク性能の向上を実現する。本研究では複数のクラウドにまたがるオートスケール機能を実現する。サーバの増設時期およびサーバの増設場所は、本研究で提案するクラウド選択ポリシーに基づき決定する。

研究成果の概要（英文）：

With the spread of cloud computing, it is possible to improve processing power of computer system. By using cloud computing systems, processing power can be increased in a local area, however, it is difficult to increase processing power in a wide area. In this paper we propose widely distributed server system by using multiple cloud computing systems. In this proposal, we realize auto scaling across multiple cloud computing systems. The timing and the place that servers will be increased are decided by cloud selection policies we proposed in this research. In this paper, we implemented prototype system that can switch cloud selection policies. To do so, it is possible to prepare environments to run virtual machine. Under the real data, we evaluated our system and showed the efficiency of our system.

交付決定額

(金額単位：円)

	直接経費	間接経費	合計
2011 年度	800,000	240,000	1,040,000
2012 年度	800,000	240,000	1,040,000
年度			
年度			
年度			
総計	1,600,000	480,000	2,080,000

研究分野：計算機システム・ネットワーク

科研費の分科・細目：情報学・計算機システム・ネットワーク

キーワード：インターネット高速化・高信頼性ネットワーク

1. 研究開始当初の背景

インターネットの普及により、様々なコンテンツ配信が行われている。配信時の急激なアクセス数の上昇（以下 Flash Crowd）により、サーバの処理能力やネットワーク帯域などの計算機資源が不足すると、正常な配信が行えなくなる。例として、3月11日に発生した東日本大震災により、東京電力が計画停電のスケジュールを Web で配信したところ、アクセスが増え、正常にアクセスできない状態が続いた。

Flash Crowd に対抗する手法の一つにコンテンツ配信ネットワーク（以下 CDN）がある。特に大規模な配信を行う際には CDN を利用して配信を行うことが多い。しかし、すべてのコンテンツ配信者が既存の CDN を利用しているわけではない。この原因の一つに、既存の CDN では、大規模配信を行う際には配信システム全体を CDN に移行させなければならないことがあると考えている。すなわち、大規模配信を想定しない通常配信時と、大規模配信を想定した CDN 利用配信時で全く異なるシステムを運用しなければならないことが問題である。

2. 研究の目的

コンテンツ配信時の Flash Crowd に対抗する手段として CDN の開発が進められている。しかし、すべてのコンテンツ配信者が CDN を利用して配信しているわけではない。この原因の一つは、大規模配信を想定しない通常配信時と、大規模配信を想定した CDN 利用時で全く異なるシステムを運用しなければならないためである。申請者は、この問題を解決するために、通常配信時の配信サーバに必要なに応じて大規模な配信能力を拡張可能な自律分散型配信システムを構築してきた。このシステムの実現に必要な要素技術は以下の 5 つである。1)新規配信サーバの動的な広域分散配置手法 2)新規配信サーバの増設時期の決定手法 3)新規配信サーバの配置位置決定手法 4)動的に増設される配信サーバ間でのリクエスト誘導手法 5)動的に増設される配信サーバ間でのコンテンツ同期手法。本申請研究ではこれまでの研究を発展させ、上述の 2), 3), 4)の三要素技術について研究開発を行い 5)については方針設計を行う。

3. 研究の方法

「目的」欄でも述べた下記の項目について、本申請研究は平成 23~24 年度の 2 年間で遂行する。

(1)新規配信サーバの増設時期の決定手法

(2)新規配信サーバの配置位置決定手法

(3)動的に増設される配信サーバ間でのリクエスト誘導手法

これらの各要素技術について、方式設計→プロトタイプ実装→シミュレーション実証実験の手順で平成 23~平成 24 年はじめに遂行する。そして、

(4)実験及び実地検証

(5)総括

について平成 24 年度に取り組む。

4. 研究成果

本研究では、複数のクラウドを利用した広域負荷分散手法を実現するため、複数のクラウドを利用したサーバ広域分散配置システムを提案した。複数のクラウドを利用したサーバ広域分散配置システムでは、サービス利用者が利用したいクラウドに関する情報、クラウドを選択するポリシー、各クラウドで起動する仮想計算機に関する情報を登録しておくことで、サービスの提供を可能とし、必要な時に計算機資源の増設を行うことが可能となる。プロトタイプシステムでは、仮想計算機イメージをあらかじめクラウドに登録しておき、サービス提供者が登録した情報をもとに仮想計算機インスタンスを起動させ、サービスの提供を可能とする。また、考案したクラウド選択ポリシーに基づき、クラウドを選択し、選択したクラウド上で仮想計算機インスタンスを起動する。これにより、複数のクラウドを利用したサーバの広域分散配置を可能としている。

本研究で提案するシステムは、複数のクラウドを組み合わせるために、制御システム、監視システム、リクエスト誘導システムの 3 つで構成される。本システムの構成を図 1 に示す。図中実線で囲まれた VM[a], VM[b], VM[c]は起動中の仮想計算機インスタンス、点線で囲まれた VM[d]は起動前の仮想計算機インスタンスを示す。仮想計算機インスタンスを囲むグレーの雲型は 2 つのクラウド、クラウド A とクラウド B を示す。以下で図 1 を用いて、本提案システムを説明する。

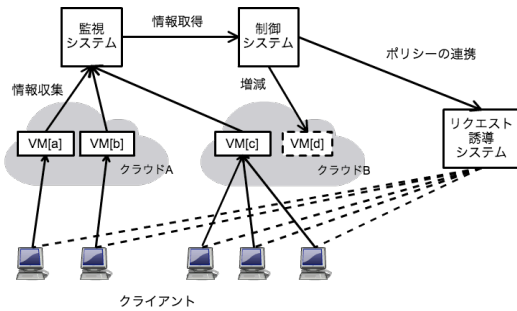


図 1 システム構成図

クラウドに関する情報と起動する仮想計算機に関する情報は制御システムに登録し、クラウド選択ポリシーは制御システムと監視システムの両方に登録する。

監視システムは、登録されたクラウド選択ポリシーに基づいて、情報を収集する。図 1 中、VM[a], VM[b], VM[c] 3 つの実行中の仮想計算機インスタンスから監視システムへの実線矢印は、監視システムがこれらから情報収集していることを示している。

制御システムは登録されたクラウド選択ポリシーと監視システムから取得した情報に基づき、次の仮想計算機インスタンスをいつ、どこで、何台起動あるいは停止するか、およびそれに先立ち仮想計算機イメージをいつ、どこに登録するかについて判断し、仮想計算機インスタンスを起動・停止、仮想計算機イメージの登録を行う。図 1 中、監視システムから制御システムへ伸びた実線矢印は、制御システムが情報を取得していることを示している。ここではクラウド B に仮想計算機インスタンスを増設することを決定し、VM[d] を起動している。制御システムから伸びた鎖線矢印がこの指示を示している。

リクエスト誘導システムは、こうして複数のクラウド上に展開した仮想計算機インスタンス群に対して、クライアントを適切な仮想計算機インスタンスへと誘導する。リクエスト誘導システムの基本的な機能は、いわゆるグローバルロードバランサーと同等である。すなわち、クライアントは、まずリクエスト誘導システムに対して、アクセスするサーバを問い合わせる。これは DNS による名前解決の中で行われる。各クライアントとリクエスト誘導システム間の点線がこの問い合わせを示している。ここで決定したサーバ（本システムにより起動された仮想計算機インスタンスの 1 つ）にアクセスする。

各ポリシーはプラグイン形式でシステムに組み込む。基本的なポリシーは予め実装済みのプログラムモジュールとして提供する。一方、それらでは実現不可能なポリシーが必要な場合、利用者が自らプラグインをプログラムしポリシーとして利用可能である。利用するポリシー名およびそのポリシーが利用するパラメータ（判断の閾値な

ど）を本システムの設定ファイルに記述する。これは、本研究においてリクエスト誘導システムとして採用した Tenbin で用いたのと同じ手法であり、有効性が確認できているため、本研究でも採用した。

監視用サーバが収集する情報の種類、保存形式および、それらの利用法は、上述の通りプラグインされているポリシーにより変化する。このように、プラグイン形式によりポリシーを追加できるため、拡張性も確保できていると考えている。

本システムでは、誘導先となる仮想計算機インスタンスが動的に増減し、更にはその際には上述の配置位置決定ポリシーによって増減する先が判断される。従って、このポリシーと連携したリクエスト誘導が必要である。例えば、サーバの過負荷を避けるために新たな仮想計算機インスタンスを増設したのであれば、負荷の高い仮想計算機インスタンスを避け、負荷の低い仮想計算機インスタンスに誘導するべきであるし、利用者との RTT を小さくするために新たな仮想計算機インスタンスを増設したのであれば、利用者との RTT が小さな仮想計算機インスタンスに誘導するべきである。このように、サーバ配置位置を決定したポリシーとリクエスト誘導のポリシーは共通の目標に基づくように連携することが必要である。

各クラウドにローカルロードバランサーが存在する場合、各クラウド内の仮想計算機インスタンス間の負荷分散をそのロードバランサーに任せるようなポリシーも実現可能である。

本研究では、複数のクラウド上で動的広域分散を実現するため、プロトタイプシステムの構築を行った。プロトタイプシステムの構成を図 2 に示す。

制御用システムの実装には、OS として Ubuntu12.04、言語として Ruby を使用した。クラウド選択ポリシーの記述言語も Ruby である。複数の異なる仮想化環境を制御す

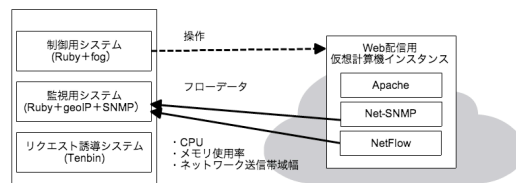


図 2 プロトタイプシステムの構成

るために、Ruby のライブラリである fog を使用した。fog は Amazon EC2 や Rackspace などの商用クラウドや、OpenStack, CloudStack などのクラウド基盤ソフトに対応している。今回の実装では、AmazonEC2 と OpenStack の 2 種類のクラウドでの動作を確認している。OpenStack の一部の機能の制御については

fog が未対応なため、OpenStack のコンポーネントである nova を使用して制御を行った。fog は様々なクラウド環境に対応しているため、本システムは今後様々なクラウド基盤システムに対応することが可能である。また設定ファイル中に、起動する仮想計算機インスタンスが必要とする CPU やメモリのサイズを記述することで、各クラウドが提供するインスタンスサイズの違いにも対応している。

監視用システムの実装には、OS として Ubuntu12.04、言語として Ruby を使用した。仮想計算機インスタンスからの情報収集には SNMP と NetFlow を利用した。現在の実装は、仮想計算機インスタンスの CPU 負荷、メモリ使用状況、ネットワーク送信帯域幅、およびフローデータを取得している。

リクエスト誘導システムには Tenbin を利用した。Tenbin は DNS を基盤とした様々なリクエスト誘導ポリシーを利用可能である。本システムに組み込むために、クラウド選択ポリシーと連携したリクエスト誘導ポリシーを開発する。これにより制御サーバからの指示によって動的にリクエスト誘導先を変更することが可能である。

プロトタイプシステムを用いて、提案手法の有効性を評価した。本評価実験においては、インターネット上の広域に分散配置された複数のクラウド上にサーバシステムを構築した。このサーバシステム上に、過去の実際のサービスで観測された負荷をシミュレートした負荷をかけ、透過的に複数のクラウドが連携しサービスを提供可能であることを示した。過去の実際のサービスとしては、LIVE!UNIVERSE が実施した日蝕中継 LIVE!ECLIPSE 2008 (以下、LE2008) における Web ページへのアクセスログを利用した。



図 3 評価システムの構成

本評価システムの構成を図 3 に示す。

本評価システムでは、制御サーバ・監視サーバ・リクエスト誘導サーバを、AWS の北バージニアリージョン上の単一の仮想計算機上に構築した。この仮想計算機インスタンスを管理サーバ用仮想計算機インスタンスと呼ぶ。管理サーバ用仮想計算機インスタンスのスペックは 7.5GB のメモリと 2ECU×2 仮想コアの CPU を持つ。

前述のように LE2008 の Web サービスをシミュレートするために広域分散型 Web サーバを構築した。Web サーバ用仮想計算機インスタンスは、北バージニア・オレゴン・北カリフォルニア・アイルランド・シンガポール・東京・シドニー・サンパウロの 8 つのリージョンで起動できるようにした。この Web サーバ用仮想計算機インスタンスの初期状態としては、LE2008 の時の実際のサーバの状況をシミュレートし日本にサーバを一台起動した。Web サーバ用仮想計算機インスタンスのスペックは、上述の管理サーバ用仮想計算機インスタンスと同じもので 7.5GB のメモリと 2ECU×仮想 2 コアである。Web サーバとして apache 2.2、NetFlow エクスポートとして softflowd、snmp マスターエージェントとして net-snmp を利用した。

負荷をかけるクライアントとなるクライアント用仮想計算機インスタンスは東京リージョン、シンガポールリージョン、アイルランドリージョン、オレゴンリージョン、サンパウロリージョンに配置した。クライアント用仮想計算機インスタンスサイズはメモリサイズが 613MB で 1ECU×1 仮想コアである。負荷は Jmeter を用いて発生させた。

本実験における仮想計算機インスタンス増設の状況を表 1 に示す。表中 JP1、JP2 は東京リージョンで起動している Web サーバ用仮想計算機インスタンス、IE1、IE2、IE3、IE4 はアイルランドリージョンで起動している Web サーバ用仮想計算機インスタンスである。実験開始後、2分4秒、6分30秒、8分58秒、11分33秒、13分54秒後に Web サーバ用仮想計算機インスタンスにおけるネットワーク送信帯域幅が閾値を超えたため Web サーバ用仮想計算機インスタンスが増設されている。2分4秒後の時点では、増設先として東京リージョンが選択されている。これは本評価実験環境ではフローデータを集めるのに遅延があり、この時点ではクライアントの IP アドレスに基づく判断ができず、デフォルトである東京が選択されたためである。フローデータ集約の遅延の原因は softflowd を用いていることによる過負荷ではないかと考えている。その次の増設判断のタイミングである 2分30秒後にはどの Web サーバ用仮想計算機インスタンスも閾値を超えていなかったために、増設は実行されていない。その後は約 2分30秒毎に増設されている。

表 1 実験の時系列

経過時間	発生したイベント
0:02:04	JP1 過負荷 増設開始
0:02:32	JP2 起動
0:02:56	JP2 への誘導開始
0:06:30	JP2 過負荷 増設開始
0:06:58	IE1 起動
0:07:22	IE1 への誘導開始
0:08:58	IE1 過負荷 増設開始
0:09:27	IE2 起動
0:09:57	IE2 への誘導開始
0:11:33	IE1 過負荷 増設開始
0:12:01	IE3 起動
0:12:33	IE3 への誘導開始
0:13:54	IE3 過負荷 増設開始
0:14:20	IE4 起動
0:14:55	IE4 誘導開始

増設場所としては、クライアントのアクセス元から分かるようにヨーロッパ・中東・中央アジア地域 (RIPE) からのアクセスが多いことをネットワークフロー情報から判断し、アイルランドリージョンが選択されている。Web サーバ用仮想計算機インスタンス増設のタイミングが2分おきではなく約2分30秒おきになっている。この理由は、Web サーバ用仮想計算機インスタンス増設途中に、さらにWeb サーバ用仮想計算機インスタンスを増設してしまうことを避けるため、Web サーバ用仮想計算機インスタンス増設途中には増設の判断を行わないためである。Web サーバ用仮想計算機インスタンスの起動に約30秒かかっているため、結果として約2分30秒おきに増設されているように見える。ただし上述のように実験開始後約4分30秒の時点でも増設の判断が実行されている。

以上の結果より、アクセス数に基づき適切なリクエスト誘導が行われていることが分かる。また、配置位置決定ポリシーに沿って選択されたクラウド上で仮想計算機インスタンスが起動し、正しくリクエスト誘導を行うことができた。

今後の課題として、他のクラウド選択ポリシーによる評価実験、様々なサービスに適したクラウド選択ポリシーの開発、仮想計算機インスタンスの縮退への対応、制御用サーバ、監視サーバの分散化、広域データベースへの対応などが挙げられる。

5. 主な発表論文等

(研究代表者、研究分担者及び連携研究者には下線)

[雑誌論文] (計1件)

① Yuta Miyauchi, Noriko Matsumoto, Norihiko Yoshida, Yuko Kamiya, and Toshihiko Shimokawa, "Adaptive Content Distribution Network for Live and On-Demand Streaming", Proceedings of Workshop on Architectures for Self-Organizing Private IT-Spheres (in 25th International Conference on Architecture of Computing Systems), pp.27-37, Munchen, Germany (査読有)

[学会発表] (計4件)

- ① 川津祐基, 神屋郁子, 下川俊彦, "動的広域分散配置型サーバシステムにおけるサーバ配置位置選択システムの開発," 電子情報通信学会 技術研究報告 (インターネットアーキテクチャ研究会), 2013年2月23日~24日, 機械振興会館 (東京都)
- ② 神屋郁子, 川津祐基, 下川俊彦, "複数のクラウドを利用したサーバ広域分散配置システムの提案," 電子情報通信学会 技術研究報告 (インターネットアーキテクチャ研究会), 2012年12月15日~16日, 広島市立大学 (広島市)
- ③ 川津祐基, 下川俊彦, 神屋郁子, "メタクラウドコンピューティングシステムにおけるクラウド選択技術の開発," 2012年電子情報通信学会総合大会, 2013年3月22日, 岡山大学 (岡山市)
- ④ 神屋郁子, 下川俊彦, "動的な広域分散型サーバシステムの問題点に関する考察," 電子情報通信学会 技術研究報告 (インターネットアーキテクチャ研究会), 2012年2月23日, 機械振興会館 (東京都)

6. 研究組織

(1) 研究代表者

神屋 郁子 (KAMIYA YUKO)
九州産業大学・情報科学部・助手
研究者番号: 00615354

(2) 研究分担者

()

研究者番号:

(3) 連携研究者

()

研究者番号：