

共有キャッシュサーバを用いた分散 Web システムにおける応答性の改善

15T226 河野 彰吾（最所研究室）

負荷量に応じてキャッシュサーバ数を動的に増減させる分散 Web システムにおいて、キャッシュサーバ起動時の応答性を改善するための共有キャッシュサーバを用いた応答性の改善について述べる。

1 はじめに

近年、クラウド環境の発展により、クラウド上にキャッシュサーバを仮想マシンとして構築することが容易になり、これらを用いることで Web サービスの応答性を向上させることが可能となった。当研究室では、負荷量に応じて動的にキャッシュサーバ数を増加（スケールアウト）・減少（スケールイン）させることで、応答性を確保しつつ運用コストを低減する Web システムを開発している [1][2]。開発中のシステムでは、リクエストを処理している Web システムのサーバが過負荷になると、キャッシュサーバを起動し、アクセスを振り分けるが、キャッシュサーバの起動が完了するまではアクセスを振り分けることができず、その間サーバの過負荷状態が続いてしまう。このため、本研究では、キャッシュサーバが起動するまでの時間のアクセスを肩代わりするキャッシュサーバを常時立ち上げることで応答性を改善する。しかし、アクセスを肩代わりするキャッシュサーバを常時立ち上げているため、その分コストがかかる。そこで、複数の Web システムでこのサーバを共有することで Web システム 1 つあたりにかかるコスト低減を図る。本稿では、この常時立ち上げておくサーバを共有キャッシュサーバと呼ぶ。

2 分散 Web システムの概要

先行研究では、図 1 に示すような、拡張ロードバランサと、キャッシュ元のコンテンツを持つオリジンサーバ、取得したキャッシュを提供する仮想キャッシュサーバ群を用いて分散 Web システムを実現している。拡張ロードバランサは、既存のロードバランサをベースにして以下の拡張プログラムが追加されている。

- A 負荷監視機能: サーバの負荷量を監視する。
- B キャッシュサーバ管理機能: 負荷量に応じて、仮想キャッシュサーバの起動・停止を行う。
- C 振分機能: 仮想キャッシュサーバ数に応じたアクセスの振り分け先を設定する。

負荷量の監視およびキャッシュサーバの増減は拡張プログラムを用いて行い、リクエストの制御はロードバランサの機能を用いて行う。

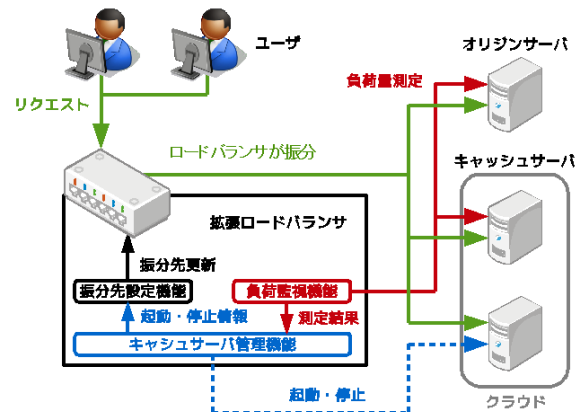


図 1: 分散 Web システムの概要

3 共有キャッシュサーバを用いたスケールアップアルゴリズム

共有キャッシュサーバのスケールアップアルゴリズムについて述べる。

- 稼働中のサーバ群が過負荷になる：
 - 共有キャッシュサーバへのリクエスト振分開始
 - 仮想キャッシュサーバの起動
- 仮想キャッシュサーバの起動完了：
 - 仮想キャッシュサーバへのリクエスト振分開始
 - 共有キャッシュサーバへのリクエスト振分停止

複数の Web システムで共有する場合は、2 つ以上の Web システムが同時に過負荷になったことで、同時に仮想キャッシュサーバの起動を行う場合もある。その場合は、それぞれの Web システムの負荷状況に応じて、共有キャッシュサーバへのリクエスト数を調整する必要があるが、本研究ではその前段階として、単一の Web システムにおいて共有キャッシュサーバを用いた分散 Web システムの開発を行う。

4 共有キャッシュサーバを用いた評価実験

共有キャッシュサーバの有効性を調べるため、先行研究と本研究のスケールアップアルゴリズムを用いた

分散 Web システムの負荷実験を行う。実験には、最大秒間リクエスト処理数が 200 の Web サーバを用いて、120 秒かけて秒間リクエスト数が 0 から 500 になるまで増加させた。その後、500 のまま 100 秒経過した後、120 秒かけてリクエスト数が 0 になるまで減少させた。仮想キャッシュサーバの起動時間は 30 秒である。

先行研究のシステムを用いた実験結果を図 2 に、共有キャッシュサーバを用いた実験結果を図 3 に示す。図 2 の稼働率 1 がオリジンサーバ、稼働率 2~5 が仮想キャッシュサーバを示し、図 3 では稼働率 1 がオリジンサーバ、稼働率 2 が共有キャッシュサーバ、稼働率 3~5 が仮想キャッシュサーバを示す。図 2 と図 3 の 40~70 秒付近を比較すると、図 3 では、共有キャッシュサーバへのリクエスト振分によって、オリジンサーバの過負荷を防ぐことに成功している。しかし、図 2 と図 3 のどちらも、130~200 秒付近では、過負荷が発生している。これは、合計稼働率を用いて仮想キャッシュサーバの起動・停止の判断を行っているため、停止を判断する閾値によって必要以上に仮想キャッシュサーバを停止していることが原因と考えられる。実験結果から、停止時の過負荷には対応できなかったが、起動時の過負荷は抑えることができたため、共有キャッシュサーバを用いた分散 Web システムは有効であることが確認できた。

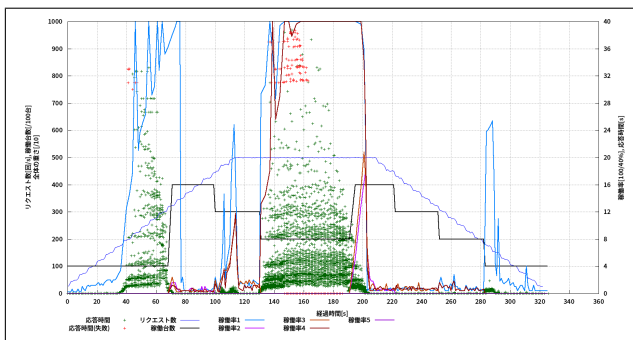


図 2: 先行研究のスケールアウトアルゴリズムを用いた実験結果

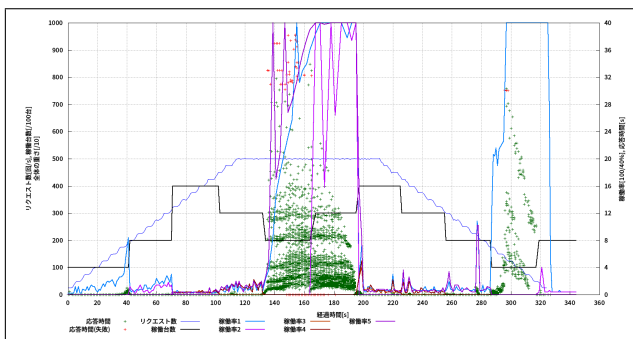


図 3: 共有キャッシュサーバを用いるスケールアウトアルゴリズムの実験結果

5 共有キャッシュサーバを用いたスケールアウトアルゴリズムの改良

起動直後の仮想キャッシュサーバでは、コンテンツがキャッシュされるまではオリジンサーバに常にアクセスし、その結果を返すため、処理量が定常状態より少なくなる。そこで、共有キャッシュサーバを用いたスケールアウトアルゴリズムを改良し、仮想キャッシュサーバの起動処理が完了しリクエスト振分を開始した後、共有キャッシュサーバへのリクエスト振分は 30 秒後に停止するように設定した。実験結果を図 4 に示す。図 3 と図 4 を比較すると、仮想キャッシュサーバの起動時における共有キャッシュサーバの働きは大差なかったが、仮想キャッシュサーバを停止することで過負荷が起こっていた 130~160 秒間では、過負荷が起こらなかったため、改良アルゴリズムは有効であると考えられる。しかし、160~220 秒間のように、過負荷状態を完全には防ぐことができていないため、さらにアルゴリズムを改良する必要があると考える。

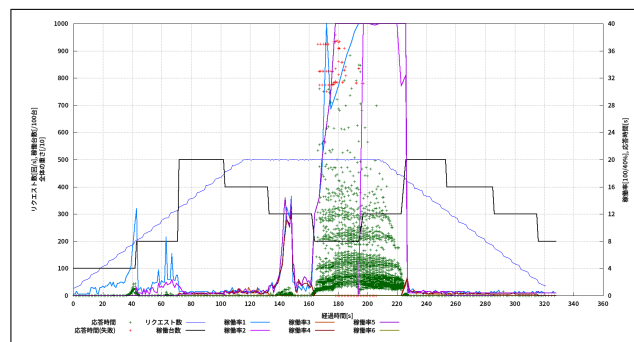


図 4: 共有キャッシュサーバを用いる改良スケールアウトアルゴリズムの実験結果

6 おわりに

本研究では、キャッシュサーバ起動時の応答性の改善を図るため、共有キャッシュサーバを用いた分散 Web システムの開発と評価を行う。また、仮想キャッシュサーバを必要以上に停止してしまう問題についても解決していく。

参考文献

- [1] 小笹光来, 最所圭三 “クラウドに適した Web システムについて”, 平成 24 年度 電気関係学会四国支部連合大会論文集”, 17-14, p.360, 2012.9
- [2] 堀内辰彦, 最所圭三 “分散 Web システムにおけるキャッシュサーバ管理機構の試作と評価”, 第 14 回情報科学技術フォーラム講演論文集, L-020, Vol.4, pp.227-230, 2015.9