

分散 Web システムにおけるキャッシュ更新方法を評価するためのシミュレータの開発

10T235 杉山 友啓 (最所研究室)

本項では、2種類のキャッシュ更新をシミュレートできるシミュレータの開発およびキャッシュ更新の評価について述べる。

1. はじめに

Web サーバの負荷を回避する手法として、キャッシュサーバを用いたキャッシングがある。我々の研究室では、Web サーバとキャッシュサーバで構成されている分散 Web システムの開発を行っている。このシステムではアクセス量に応じてキャッシュサーバの数を増減する。Web サーバでは、キャッシュサーバのコンテンツの更新のためのリソースを確保し、他のアクセスよりも優先してサービスを行う。

本稿では、Web サーバで確保されているリソース量に応じて、キャッシュサーバ上のキャッシュの品質が最大となるようなキャッシュ更新方法を評価するためにシミュレータの開発およびキャッシュ更新の評価について述べる。

2. 分散 Web システムについて

本研究で用いる分散 Web システムの構成を図 1 に示す。本研究ではキャッシュサーバ上のキャッシュ管理を対象としている。キャッシュサーバは Web サーバのコンテンツをキャッシュしており、クライアントからのアクセスに対しキャッシュしているデータを返す。Web サーバとキャッシュサーバの負荷状況を監視し、状況に応じてキャッシュサーバの増減および振り分け機構への振り分け先を指定する。このため、キャッシュサーバ数に応じてキャッシュの更新に利用できるリソース量に変化する。リソース量に応じたキャッシングを行う必要がある。

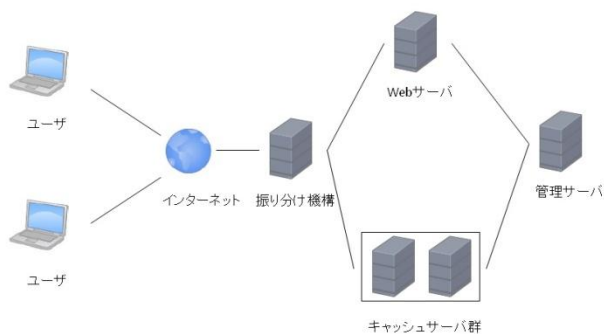


図 1: 分散 Web システム

3. キャッシュするコンテンツ

キャッシュは時間が経つにつれて提供する情報の品質が低下する。キャッシュするコンテンツには様々な種類があり、例えば広告などのコンテンツや、ブログや掲示板などのコンテンツがある。前者は、有効期限が来るまで変わらず、有効期限を超えてしまうとキャッシュの品質がなくなってしまうのに対し、後者は最新情報でなくとも十分な情報が得られるため、キャッシュの有効期限を超えても得られると考えられ、キャッシュの品質が急激には落ちず徐々に品質が低下していくと考える。さまざまなコ

ンテンツの種類があるが、本研究ではキャッシュの品質が 2 つのパターンで変化するコンテンツについて扱う。1 つはキャッシュの有効期限を超えると品質が 0 になるタイプ A (図 2) と、キャッシュの品質が徐々に低下していくタイプ B (図 3) である。

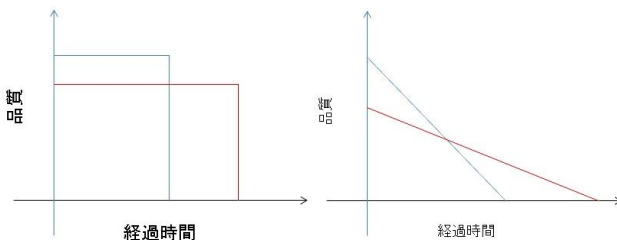


図 2: タイプ A

図 3: タイプ B

タイプ A のキャッシュには期限内の品質と時間を与えられ、有効期限内に更新を行う必要がある。タイプ B のキャッシュには初期の品質と低下の割合を与えられ、一定の割合まで低下すると更新を行うこととする。

4. キャッシュの更新方法

キャッシュを更新していく上でどのタイミングで更新を行うかを考える必要がある。キャッシュの更新のタイミングにはプリフェッチとオンデマンドの 2 種類が考えられる。

プリフェッチとは、キャッシュが決められた期限ごとに定期的に更新する方法である。有効期限が切れる前に自動的に更新することで、キャッシュの期限切れを防ぐことができる。ユーザに対してはキャッシュしているコンテンツをそのまま返す。このため、キャッシュの更新ミスによって品質が大きく低下する。

オンデマンドとは、アクセス要求に応じて更新する方法である。アクセスが発生したときキャッシュの品質がある値よりも下がっていたり、キャッシュの有効期限が切れていれば Web サーバに問い合わせ更新を行う。そのため、Web サーバの問い合わせが発生する場合、その分クライアントへの応答時間が遅くなる。

クライアントに与えるキャッシュの品質はオンデマンドの方が高くなると考えられるが、クライアントへの応答時間が遅くなる。今回は、ユーザへの応答時間の短縮を優先しプリフェッチを採用する。また、タイプ A のキャッシュの期限はできる限りすべて守るようにする。

5. キャッシュ更新のシミュレータ

今回の評価では、プリフェッチ方式でシミュレーションを行う。キャッシュサーバは、指定された

キャッシュの有効期限が切れるまえに Web サーバに更新のリクエストを送り、更新を行う。今回はキャッシュサーバにあらかじめ保持しているキャッシュの数をそれぞれのタイプで設定する。また、タイプ A は更新時間と更新間隔を設定し、タイプ B は更新時間とキャッシュの品質の減少量を設定する。

シミュレータの処理について述べる。まず、タイプ A のキャッシュについての処理を行う。タイプ A はできるだけ更新を行うため、タイプ A 全てのキャッシュが更新できるようにあらかじめ更新を開始する時間を決めておく。ここで更新が重なってしまった場合、図 4 に示すように更新時間の前倒しを行う。ここで、 n 番目のキャッシュの更新時間を t_{sn} 、更新終了時間を t_{cn} とする。 t_{sn} が t_{cn+1} より小さい場合は前倒しを行い、 t_{sn+1} から $t_{cn+1} - t_{sn}$ の値を引き、新しい t_{sn+1} とする。これを最後まで繰り返す。タイプ B のキャッシュの更新はタイプ A のキャッシュの更新を行わない時間に更新を行う。タイプ A、タイプ B ともに更新を行わなくてもよい場合は何もしない。最後に、キャッシュの品質はタイプ A については有効期限内は 1、有効期限外は 0 とする。タイプ B は品質を決められた減少量に従って減少する。ある時間のキャッシュ全体の品質は対象とするキャッシュの平均値とする。

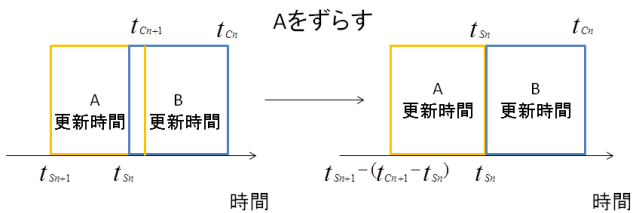


図 4：キャッシュの前倒し

6. シミュレーション

タイプ A とタイプ B が混在している場合のキャッシュの品質の変化について調べる。以下のパラメータで行った実験結果を図 5 に示す。

- ・タイプ A
 - キャッシュ数：50 個
 - 平均更新時間：1 秒 (0.8~1.2 秒の間の一様分散)
 - 更新間隔：100 秒
- ・タイプ B
 - キャッシュ数：50 個
 - 更新時間：1 秒
 - 品質の減少率：0.5%
 - 更新する割合：50%

タイプ B の品質の減少率は 1 秒間あたりの減少率であり、品質が 50%以下になると更新を行う。このため単独の場合、更新間隔は 100 秒である。タイ

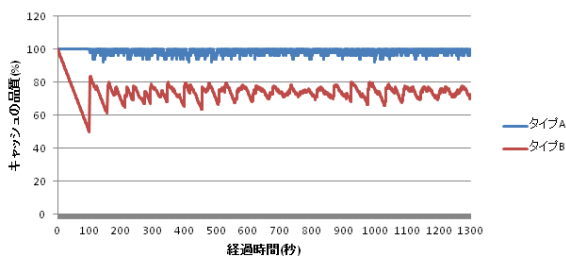


図 5：タイプ A の更新時間が 1 秒の時の品質の変化

ぎを持たせている。タイプ A のすべてのキャッシュ A のキャッシュの更新時間は乱数を用いて揺らの更新に要するリソースの量は、 $(1/100)*50 = 0.5$ (50%)、タイプ B も $(1/100)*50 = 0.5$ (50%) となり、全体で 100% になる。更新の要するリソース量が 100% 以下であるため両タイプともほとんど期限切れが起きず更新されている。

次にタイプ A の平均更新時間を 2 秒に変更したときの品質の変化を図 6 に示す。更新に要するリソース量は、 $(2/100)*50 = 100%$ 、タイプ B は $(1/100)*50 = 50%$ となる。この場合、タイプ A の更新の要するリソース量に余裕がないためタイプ A の更新ばかりが行われ、タイプ B の更新がほとんど行われていないことが分かる。本来ならタイプ B の更新は行われているが、タイプ A の更新時間に揺らぎを持たせているため、更新時間の空きが少しだけ出来ている時にタイプ B の更新が行われている。

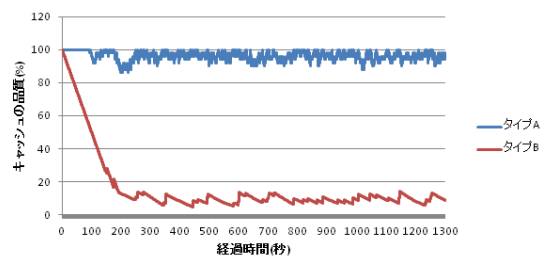


図 6：タイプ A の更新時間が 2 秒の時の品質の変化

7. おわりに

分散 Web システムにおけるキャッシュサーバ上でキャッシュの更新方法についてシミュレータを用いて検討を行った。ある有効期限を過ぎるとキャッシュの品質が 0 に落ちるタイプ A のキャッシュと、時間経過とともにキャッシュの品質が徐々に低下していくタイプ B の 2 つのキャッシュについて、ある一定の更新のリソースが与えられた上でシミュレータを用いてさまざま条件においてキャッシュの品質がどのように変化したか評価を行った。その結果、タイプ A の更新に要するリソースの割合によってタイプ B のキャッシュの品質が変化していくことが分かった。

参考文献

- [1] 小笹光来, 最所圭三, "クラウドに適したキャッシュサーバを用いる Web システムにおける管理機構の開発およびその評価", 平成 24 年度香川大学修士論文集, 2013.
- [2] 山田茂和, 最所圭三, "NAP-Web の時間予測に関する評価と優先アクセス機構の設計", 平成 24 年度香川大学修士論文集, 2013.
- [3] 曾川直也, 最所圭三, "NAP-Web における優先アクセスの設計と評価", 平成 25 年度香川大学卒業論文, 2014.
- [4] 堀内農彦, 最所圭三, "クラウドに適した Web システムの負荷監視機能の改善と評価", 平成 25 年度香川大学卒業論文, 2014.
- [5] 村上拓哉, 最所圭三, "分散 Web システムのためのキャッシュ管理機構の提案", 平成 24 年度香川大学卒業論文集, 2013.