

Web サーバの負荷状況に応じた動的ミラーリング機構

02T286 入江正行（最所研究室）

近年のネットワークの普及や技術の革新に伴い、Web サーバの負荷は増大している。そこで、Web サーバの負荷状況に応じて動的にミラーリングを行うシステム的设计、実装、評価を行った。

1. はじめに

近年のブロードバンドの普及に伴い、ユーザの求める要求も多様性に富むようになった。Web ページでは、ファイルサービスだけでなく、ブログサービスなどのクライアントからの要求によって、Web サーバ上のデータを更新するサービスや今後大きな需要が見込まれているストリーム型の映像配信サービスなど、様々なサービス体制がとられるようになった。しかし、ネットワークサービスを提供する Web サーバのリクエスト数は増加の一途を辿っている。その対策として、検索エンジンや大手企業の人気 Web ページ等では、ミラーサーバを用いることが多い。ミラーサーバとは、同じ情報を提供するためにデータをコピーした Web サーバのことで、これらを複数用意することで、アクセスを分散できるだけでなく、信頼性を向上できる。しかし、それらの多くは静的にミラーリングされたものであり、被ミラーサイトに対して固定的である。固定的に設置されたミラーサーバでは、その数が増えるに従い、管理が難しくなり各種コストも増大する。その上、Web サーバへのアクセス数が比較的少ない時間帯では、ミラーサーバは遊んでしまい非効率的である。

そこで本研究では、単一のミラーサーバに対し、複数の Web サイトをその負荷状況によって動的にミラーリングすることで、各サーバの負荷を軽減し、時間帯によるミラーサーバの CPU 資源を有効利用できるアクセス制御機構を提案し実装する。

2. 設計

2.1 システム構成

Web サーバの負荷状況によって動的にミラーリングを行い、アクセスを分散させることで、Web サーバの負荷軽減を図る。この場合の問題点とその対策方法を以下に示す。

(1) ミラーサーバ設置のタイミング

Web サーバの CPU 負荷値を基に決定する。

(2) アクセス制御

HTTP におけるリダイレクト機能を用いて Web サーバに送信されたリクエストをミラーサーバへ受け渡す。

(3) 動的ミラーリング間のデータ更新方法

一定時間毎に Web サーバ側、ミラーサーバ側のデータを参照し、一方が更新されればもう一方も更新する。

(4) ミラーサーバ自身の負荷が増大した場合 幾つかの選択肢がある。2.2 節で論じる。

動的ミラーリングを行うに当たって、Web サーバ側では随時、負荷状況を調べ、その値を基にミラーリングのタイミングを計る。値が閾値を超えればミラーリングを行い、そうでない場合は Web サーバ、クライアント間で通常の通信を行う（図 1）。

ミラーサーバ側では各サーバの負荷状況に対して優先付けを行い、負荷の高い Web サーバから優先的に処理を行う。優先付けの因子としては、Web サーバ側の負荷状況、回線速度等が挙げられる。

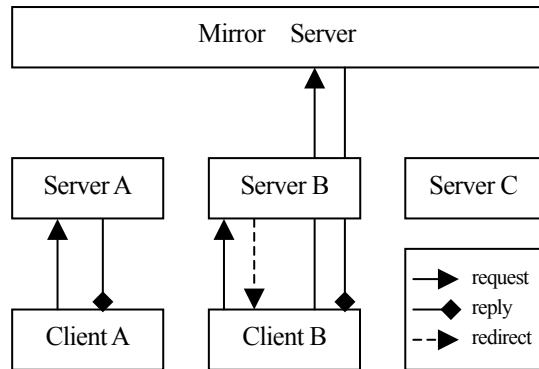


図 1：負荷状況に応じた動的ミラーリング

2.2 動的ミラーリング機構

各状況に対する動的ミラーリングの処理の流れを説明する（図 2）。

① 接続先の Web サーバ負荷が閾値を超えた場合

Server A にアクセスが集中した場合は、Server A は Client からのリクエストを Mirror Server へリダイレクトする。リダイレクトを受け取った Client は Mirror Server へリクエストを送信し応答を受け取る。

② 複数のサーバのミラーリングを行う場合

Mirror Server が Server A のミラーサーバとして通信している状態において、Server B にアクセスが集中した場合は Mirror Server は同時に複数の Web サーバのミラーリングを行う。

③ ミラーサーバ自身の負荷が増大した場合

負荷の高い Web サーバを複数ミラーリングすると、Mirror Server 自身の負荷が増大してしまい、通信速度が低下してしまう。そこで、Mirror Server は自身の負荷状況を監視しておき、その値が閾値を超えると、優先度の低いサーバのミラーリングを止める。この場合、ミラーサーバへのアクセスは Web サーバが受け取ることになる。そして負荷が軽減するのを待ち再度ミラーリングを行う。

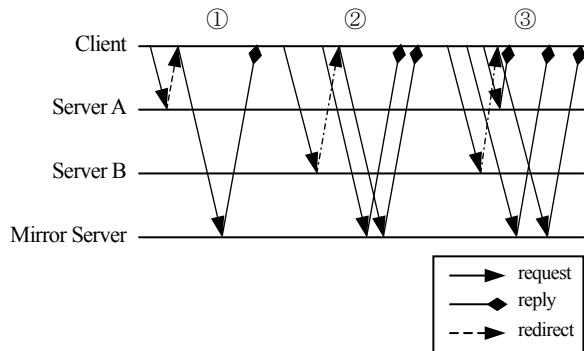


図 2 : 動的ミラーリングの流れ

3. 実装

2節の設計に基づいてLinux上でC言語を用いて動的ミラーリング機構の実装を行った。複数のWebサーバに対するアクセス制御部の実装までは行っていない。実装したのは単体Webサーバに対する動的ミラーリング機構である。

4. システム評価実験

3節で実装したシステムに対して評価実験を行った。評価を行う際に用いたプログラムを以下に示す。

- ・ 負荷増加システム
 - CGIによる負荷増加プログラム
 - 多量リクエストによる負荷増加プログラム
- ・ 計測システム
 - 表示時間計測プログラム(クライアント側)
 - リクエスト処理時間計測プログラム(サーバ側)

WebサーバはCPUパワー不足、メモリ不足、ディスクI/O不足等のハードウェアリソースの不足が複合的に絡み過負荷状態になる。このため、負荷増加システムでは、以上に上げたプログラムを同時に使用することで、実際のWebサーバ負荷に近い負荷状況が発生させる。また、計測システムではクライアント側でファイルのダウンロード開始から完了までの時間を計測するブラウザをVisual Basicを用いて作成した。またサーバ側では単一リクエストに対する処理時間をログに出力するように設定した。これらにより、クライアント、サーバの両方向からの計測を行った。

図3は動的ミラーリング機構を動作させていない場合である。リクエスト処理時間及び表示時間は共にWebサーバ負荷が高いほど増大する傾向が見られた。図4は動的ミラーリング機構を動作させている場合であり、Webサーバの負荷値が閾値を超えた時に動的にミラーリングを行う。評価実験ではWebサーバの負荷値が2.3の時に動的にミラーリングを行うように設定した。これにより、ミラーリング時はリクエスト処理時間及び表示時間が増加せず、揺らぎも抑えられることが確認できた。

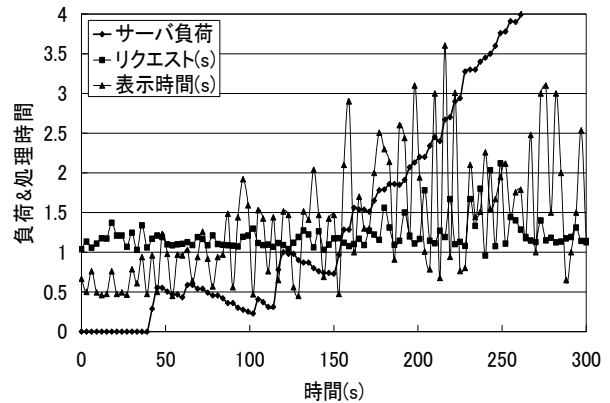


図 3 : 動的ミラーリング未実行時

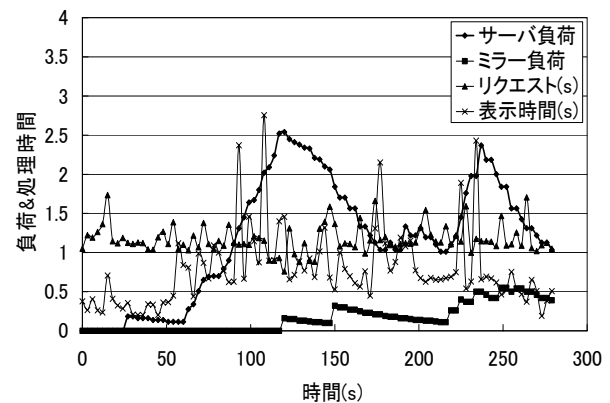


図 4 : 動的ミラーリング実行時

5. まとめ

本研究ではWebサーバの負荷を制御し、過負荷時における表示時間及びリクエスト処理時間の短縮を目的とするシステムの開発を行った。始めに負荷がかかった場合のリクエスト処理時間を調査した。次に本システムの通信機構及びアクセス制御機構の設計を行い、その実装を行った。そして、実装したシステムの評価実験を行い、その有効性を確認した。今回実装したシステムの問題点は次の通りである。

- (1) 急激なWebサーバの負荷増加によるリクエストの取りこぼしが発生
- (2) 負荷増加システムにおける負荷制御が不十分
今後の課題としては、以下の点が挙げられる。
 - (a) 動的ミラーリングのタイミングの最適化と負荷制御
 - (b) 複数のWebサーバに対する動的ミラーリング機構の実装及び各サーバに対する優先度の決定
 - (c) 以上のシステムを実装した場合の有効性の評価

参考文献

- [1] Apache
<http://www.apache.jp/>
- [2] LFTP-sophisticated file transfer program
<http://lftp.yar.ru/>
- [3] 加地智彦, 「Webサーバの負荷状況に応じたアクセス制御」, 香川大学工学部, 卒業論文 2004