

複数のミラーサーバを用いたWebアクセス最適化

— Range機能及びリソース先読みによる効率化 —

03T227 川合信一（最所研究室）

本研究では、サーバから効率良くリソースを取得するために必要とされる基本的な機能の特性調査を行った。さらに、リソース先読み機能をプロキシに追加し、その評価を行った。

1. はじめに

現代、ネットワークが急速に普及し、いつでも、どこでも、だれとでも通信が可能なユビキタス社会が到来している。その一方で Web アクセスが急増し、アクセスが集中する Web サーバのサービスを著しく低下させている。そこで、負荷を分散させる手段として、ミラーサーバと呼ばれる同じサービスを提供するサーバを複数設置することがある。

我々の研究室では、ミラーサーバを効率よく利用するプロキシプログラムを研究しており、すでに複数のミラーサーバに同時にアクセスすることが有効であるとわかっている[1]。加えて、HTTP パイプラインと Range 機能を用いることが示唆されていたが、これらの特性はまだ調査されていない[2]。そこで本研究では、これらの特性を調査する。さらに HTTP パイプラインを効率よく利用するためのリソース先読み機能をプロキシに追加し、その評価を行う。

2. HTTP パイプラインの特性調査

プロキシプログラムでは、リソースの大きさに応じて複数のミラーサーバにリクエストを振り分ける。このため、リソースの大きさを知る必要がある。HTTP においてリソースを取得する際には、基本的に GET と呼ばれるメソッドが使われる。その他にも、リソースの情報のみを取得する HEAD も用意されている。HTTP1.1 では様々な機能が加わった。その代表に HTTP パイプラインと Range 機能がある。前者は複数のリクエストを、レスポンスを待つことなく連続してサーバへ送信する機能であり、後者は GET に付加することでリソースの取得範囲を指定できる機能である。

GET と HEAD の応答時間差及び、HTTP パイプラインの効果を調査する。対象とするサーバは hc4.seikyoku.ne.jp（以降、外部ホスト）、対象リソースは事前に準備した約 40KB のテキストファイルとする。調査結果を図 1 に示す。

GET と HEAD は、HTTP パイプラインを使用すると取得時間が短くなるのが読み取れる。これは HTTP パイプラインを用いることにより、TCP コネクションの回数が少なくなるためである。また、HTTP パイプライン使用時において、GET は HEAD と比較して取得時間が非常に大き

いことがわかる。このことから Web アクセスを最適にするためには、まず、HEAD を用いて HTTP パイプラインにより各リソースのサイズを取得し、その後、適切な割り当てを行うことが有効であるといえる。

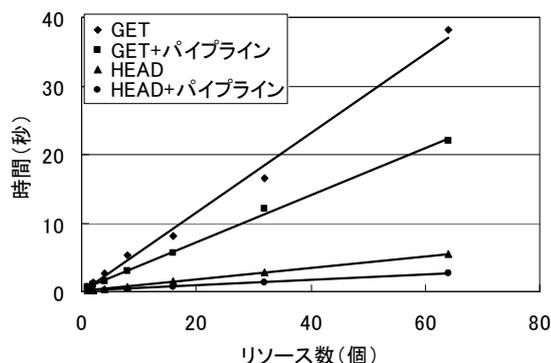


図 1 HTTP パイプラインのリソース取得特性

3. Range 機能の特性

本調査は、Range 機能を用いることによるオーバーヘッドの有無を確認する目的で行う。調査内容は以下の 3 つである。

1. 実際のリソースサイズより大きなサイズを指定して取得した場合と、Range 機能なしで取得した場合に応答時間差がでるのか
2. リソースの一部を取得した場合の応答時間はどのようになるのか
3. リソースの取得位置を変えることにより応答時間が変化するのか

これらの調査の対象とするホストとリソースは前の調査と同一である。調査結果を図 2、3 に示す。

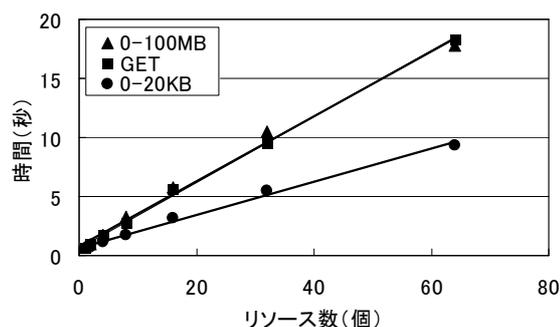


図 2 Range を用いたリソース取得特性(1)

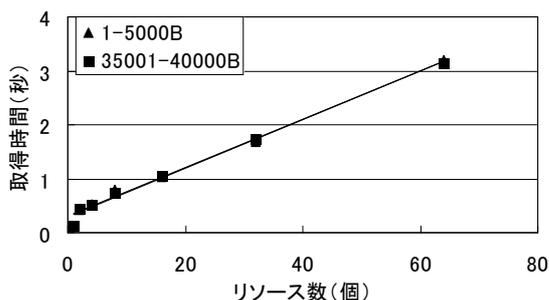


図3 Rangeを用いたリソース取得特性(2)

調査の結果、Range機能を用いることによる目だったオーバーヘッドはないことがわかった。

4. リソース先読み機能

HTTPパイプラインは、一度に多くのリクエストをサーバに送信する際に有効な機能である。しかし、クライアントが断続的に少しずつ送信する場合があります、この機能を有効利用できないことが考えられる。そこで、プロキシにリソース先読み機能を実装することにした。

リソース先読み機能とは、クライアントが要求するリソースを予測し、クライアントからリクエストを受け取る前にサーバからあらかじめリソースを取得する機能である。本プロキシには、その第一歩として、HTML形式のレスポンスからWebページに最も多く含まれていると思われる画像データのURLを抽出する機能を実装することにした。抽出は、画像データを表すimgタグを検索することにより行う。

5. リソース先読み機能の評価

クライアントからのリクエストを素通しするプロキシと、リソース先読み機能を実装したプロキシをそれぞれクライアントのブラウザに設定し、インターネット上のWebページに実際にアクセスする際の取得時間を測定する。その取得時間の差から本機能の性能を評価する。対象とするページを表1、測定結果を表2に示す。

表1 対象Webページ以降の呼び名 URL

以降の呼び名	URL
P1	www.yahoo.co.jp/
P2	auction.rakuten.co.jp/
P3	www.kentei.ne.jp/
P4	www.amazon.co.jp/

表2 測定結果

対象ページ	P1	P2	P3	P4
時間短縮率(%)	9.01	18.38	-4.21	32.26

表2に示すように、対象ページによって時間短縮率が大きく異なり、P3ではマイナスの効果となっている。対象とするページによって効果が大きく異なる原因は主なものとして次の3つが考えられる。

1. 必要のないリソースの取得
2. クライアントからの総リクエスト数に占める予測できたリソース数の割合の違い
3. サーバからクライアントが要求している順にレスポンスが返ってこないことによるオーバーヘッド

これらのうち、1と2について調査した。その結果が表3である。

表3 サーバへのリクエストの内訳

対象ページ	P1	P2	P3	P4
リクエスト総数	29	81	65	60
予測リソース数	25	83	34	56
予測不可リソース数	4	1	33	4
予測誤り数	0	3	2	0

表2と組み合わせて見ると、P3では、全リクエストの半数程度のリソースしか予測できておらず、必要のないリソースも取得している。このために、時間短縮につながらなかったといえる。

6. おわりに

HTTPパイプラインとRange機能の特性調査により、その有効性が確認できた。

HTTPパイプラインを有効利用するために、リソース先読み機能を実装し、その性能を確かめるため実験を行った。その結果、対象ページによって効果が大きく異なることを確認した。

今後の課題は、より多くのリソースを予測できるように改良することである。そのためにはimgタグだけでなく、iframeタグやbackgroundタグに対応させること、css形式やjs形式のリソースからも予測することが考えられる。

参考文献

- [1] 藤澤弘：複数のミラーサーバへの分散アクセスによる効率的なアクセス機構に関する研究、香川大学工学部、修士論文、2003
- [2] 岩佐年倫：複数のミラーサーバを用いたWebアクセスの最適化に向けて、香川大学工学部、卒業論文、2005