

九州大学における WWW キャッシュサーバの運用と評価

笠原 義晃[†] 石田 慶樹[‡] 古川 善吾[‡]

[†]九州大学大型計算機センター [‡]九州大学情報処理教育センター

概要 近年、インターネットが世界的に注目されており、ユーザの爆発的増加に伴うトラフィックの増加が問題となっている。現在インターネットの幹線を流れるトラフィックの大部分は WWW に関するデータである。WWW のトラフィックには重複するデータが多いため、キャッシュサーバを利用することによって幹線にかかる負担を軽減することができる。本稿では、九州大学で約 1 年間運用してきた学内向けキャッシュサーバのログや応答時間をいくつかの側面から解析し、評価と問題点の考察を行った。結果として、ヒット率は 60% 程度と満足できるものの、アクセス集中時に通常の 50~100 倍程度の応答時間の劣化が観測された。

Management and Evaluation of a WWW Cache Server in Kyushu University

Yoshiaki Kasahara[†] Yoshiki Ishida[†] Zengo Furukawa[‡]

[†]Computer Center, Kyushu University

[‡]Educational Center for Information Processing, Kyushu University

Abstract Recently, an explosive increase of population and traffic of the Internet became a severe problem. The WWW (World Wide Web) is the major source of the explosion, and now the traffic on the backbone of the Internet is mostly dominated by the WWW traffic. There are a lot of duplication in the WWW traffic, and WWW cache servers may decrease traffic on the backbone. This paper analyzes the log files of a WWW cache server which has been running for WWW users in Kyushu University since August 1996. It is acceptable that the hit rate on the WWW server is about 60%, but the severe increase of a response time is observed during busy time.

1 はじめに

近年、インターネットが世界的に注目を集めており、ユーザ数の爆発的増加に伴ってトラフィックの増加が問題となっている。特に World Wide Web(WWW) はインターネット上での標準アプリケーションとして定着した感があり、現在インターネットの幹線を流れるトラフィックのかなりの部分がこの WWW に関するデータとなっている。現状ではトラフィックの爆発的な増加に回線容量の増強が追いついておらず、回線が常に圧迫されている状況が問題となっている。このような状況を緩和するために、さま

ざまな組織やプロバイダで WWW のキャッシュサービスが行われている。

一般に WWW のトラフィックには重複するデータが多いため、重複する部分をローカルにキャッシュして提供する事により、幹線にかかる負担を低減することが可能である。トラフィックの低減はキャッシュヒット率の増加によって達成される。また、ユーザの観点から見ると、ある URL を指定してから最初のデータが届くまでの時間(応答時間)、またデータが届き始めてから全部のデータが届くまでの時間(転送時間)が短縮されることが重要である。キャッ

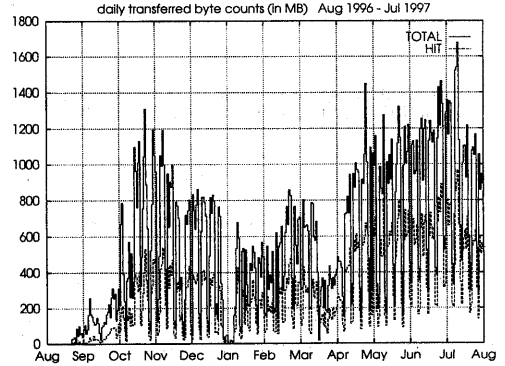
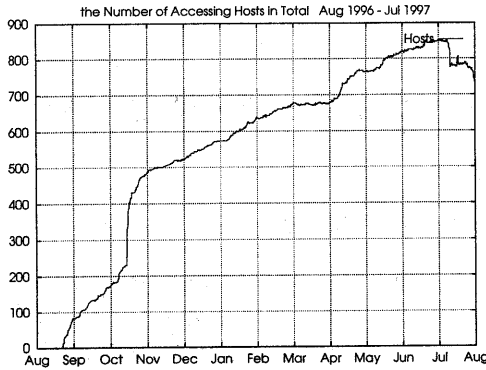


図 1 : アクセスホスト数とトラフィック量の推移

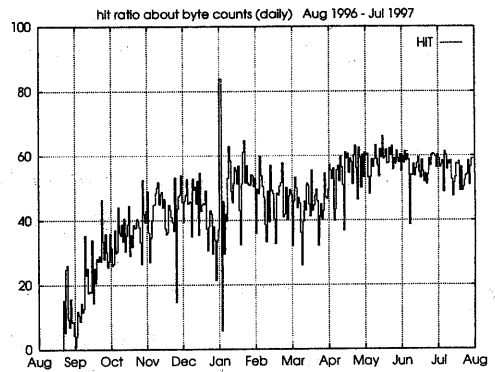
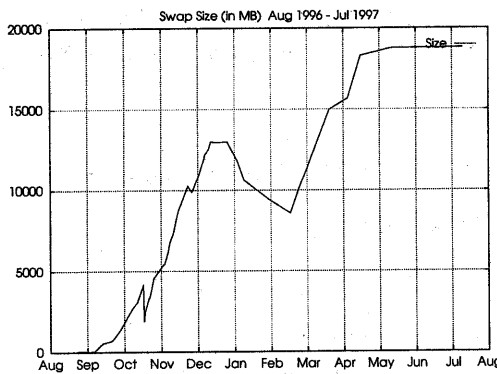


図 2 : キャッシュ容量とヒット率の推移

シユサーバを利用する場合、ネットワーク的に遠方にあるサーバの内容がキャッシュされると、次のアクセスではデータの転送時間が大幅に短縮される。また、内容がキャッシュされていて元サーバとの通信が発生しない場合、より近いキャッシュサーバとの通信になるため応答時間が短縮される。

九州大学でも、1996年の8月末から約1年に渡って試験的に学内全体向けWWWキャッシュサーバを稼働させている。本稿では、このサーバについて、稼働開始からのログと応答時間をいくつかの観点から解析し、キャッシュの性能評価と問題点の考察を行う。

2 運用環境

本キャッシュサーバでは、ソフトウェアにSquidを利用している[1]。ソフトウェアのバージョンは随時Squidのバージョンアップに追従

してきているため時期によって異なる。キャッシュサーバ用マシンとしてはSparc Station 20にメモリ192MBを搭載し、32GBの容量を持つハードディスクのうち20GBをキャッシュ用として割り当てている。メモリは当初64MB搭載していたが、20GBのキャッシュに対し64MBでは管理情報がメモリに乗らず、性能が劣化すると思われたため途中で増設した。このサーバにはネットワークインターフェイスとして学内の幹線FDDIループ直結のFDDIインターフェイスが装備されており、100Mbpsで学内ネットワークに接続されている。

3 アクセスホスト数とトラフィックの推移

図1にクライアントとしてキャッシュサーバを利用するホスト数とトラフィックの推移を示す。ホスト数はキャッシュサーバに初めてアクセスした時から最後にアクセスした時までを1

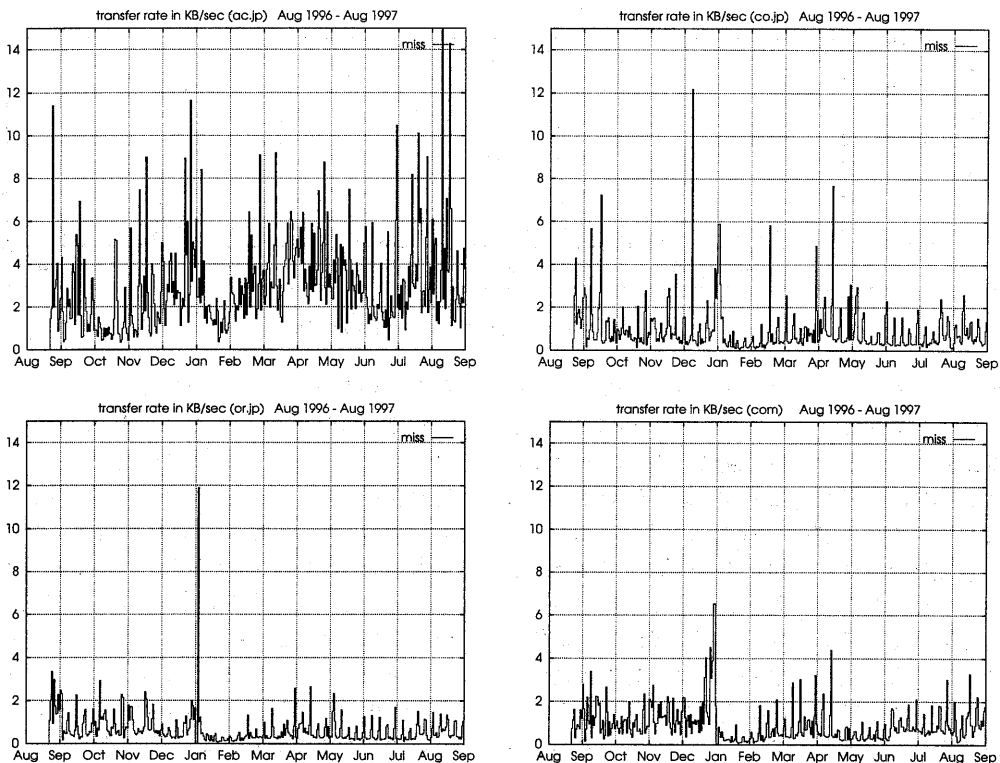


図 3：ドメイン別転送速度(一部)

と数え、間違っアクセスしたホストは数えていない。7月に急激な減少が見られるのは、解析したログが夏休み中で終わっており、夏休みに入ってアクセスをやめたホスト分が減っているためである。トラフィックはクライアントに対する一日の総転送量とそのうちのキャッシュヒット分をそれぞれMB単位で示してある。

ホストのグラフを見ると、現在 800 台程度のホストがキャッシュを利用している事がわかる。九大の学内ネットワークには 5,000~6,000 台のホストがあるが、特にポート 80 番を塞ぐなどしてキャッシュの利用を強制するような事はしていないため、この程度の利用に留まっていると思われる。

1996 年 10 月中旬に不連続な増加があるのは、情報処理教育センターの管理する学生向け端末(約 350 台)でキャッシュサーバの利用を行うような設定を一斉に行ったためである。ほぼこれと同時期からキャッシュサーバへのアクセ

スが急激に増加している事がわかる。

4 総キャッシュ容量とヒット率の推移

図2に総キャッシュ容量(MB)と、データ量の比率で計算したヒット率の推移を示す。キャッシュ容量は通常はログに記録されず、Squidを再起動して内部情報を再構成した際に記録されるだけであるため、かなり疎なグラフになっている。途中1996年12月から翌年3月にかけて減少傾向にあるが、これは主メモリの不足によるパフォーマンスの低下が危惧されたため、メモリ増設まで総キャッシュ容量の上限を10GBに制限した事による。

この図でのヒット率は、一日単位での全転送量とキャッシュヒット時転送量の比である。ここで言う「キャッシュヒット」には、キャッシュ元のデータに全くアクセスせずにキャッシュの内容をクライアントに返した場合と、キャッシュ元のデータが更新されているかどうかを調べた結

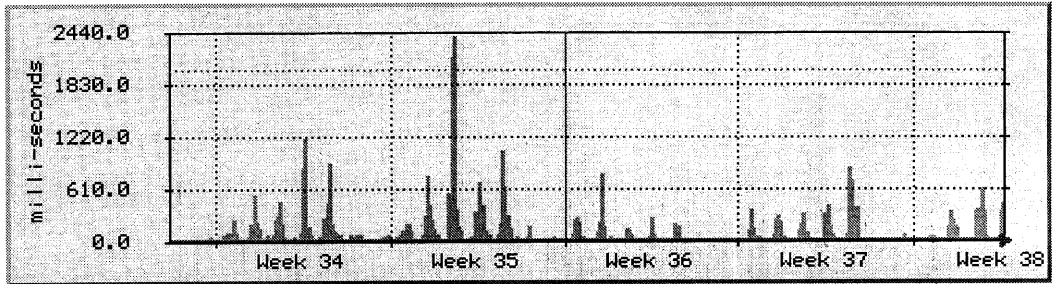


図 4：キャッシュサーバの応答時間

果変化がなかったためキャッシュ内のデータをクライアントに返した場合を含んでいる。キャッシュ元サーバとの通信が起る場合ユーザに対する応答時間は長くなるが、応答時間はログに残らないため、詳しい調査はできていない。

現在ヒット率は平均して60%程度で推移しており、ヒット率から見ると十分な性能が出ていると考えている。しかし、利用者が九大全体に比べて少ないと予想されるため、対外接続のトラフィックに対する貢献度はあまり高くない可能性がある。学外からのトラフィックの内容内訳は現在収集されておらず、全体のWWWトラフィックに対するキャッシュ経由トラフィックの割合は今の所明らかになっていない。

5 ドメインごとの転送速度の差異

図3にトップドメイン別のデータ転送速度のグラフを示す。日本のトップドメインから3種と、海外のドメインでアクセス数の多いcomドメインをグラフにした。キャッシュヒット時とキャッシュミス時の両方の転送速度を計算したが、両方描くとグラフが見づらくなるので、図ではキャッシュミスした時の転送速度だけを示している。図には示していないが、キャッシュヒット時の転送速度は概ねキャッシュミス時の転送速度より速く、転送速度についてキャッシュの効果が出ていることがわかる。アクセス数が少ない日には異常に大きな数値が出る場合があり、最大値までをグラフに描くと比較が難しいため、グラフの上限は15KB/sに制限してある。

ドメイン名ではデータの通る経路が考慮されないため厳密な議論はできないが、それでもドメインによってキャッシュミス時の転送速度に

ある程度傾向があることがわかる。九大は主にSINETを利用していて商用ISPとの接続性が悪いため、co.jpドメインやor.jpドメインは転送速度が遅い。逆にSINETで直接接続している組織の多いac.jpはかなり転送速度が速い。実際はco.jpやor.jpへのアクセスが他のjpドメインに比べて圧倒的に多いため、その影響でjp全体で見た転送速度は遅くなっている。

1996年12月31日まではInternet World Expo '96関連で臨時に東京から九大まで高速回線が引かれており、また海外線も広帯域の回線を利用することができた。しかしcomドメインの転送速度グラフを見る限りでは、言われてみれば去年の方が若干転送速度が速いようだという程度で、それほど数値に違いはないようである。

6 キャッシュサーバの応答時間の推移

図4は8月後半から9月前半までの約一箇月間における、5分おきに測定した応答時間の2時間ごとの平均を示したグラフである。Squidの場合、ログにはクライアントからの要求に回答するまでにかかった時間は記録されない。そのため、定期的にキャッシュサーバに対してあるファイルをHTTPで要求して、結果が帰ってくるまでにかかった時間を収集することによって応答時間を測定している。キャッシュサーバの応答時間だけを測定するために、対象ファイルは同一セグメントにあるWWWサーバ上の小さなファイルを使っている。このファイルはキャッシュに入っている確率が非常に高く、また転送時間はほとんど無視できる。グラフ化にはMRTGを利用し[2]、出力はリアルタイムに

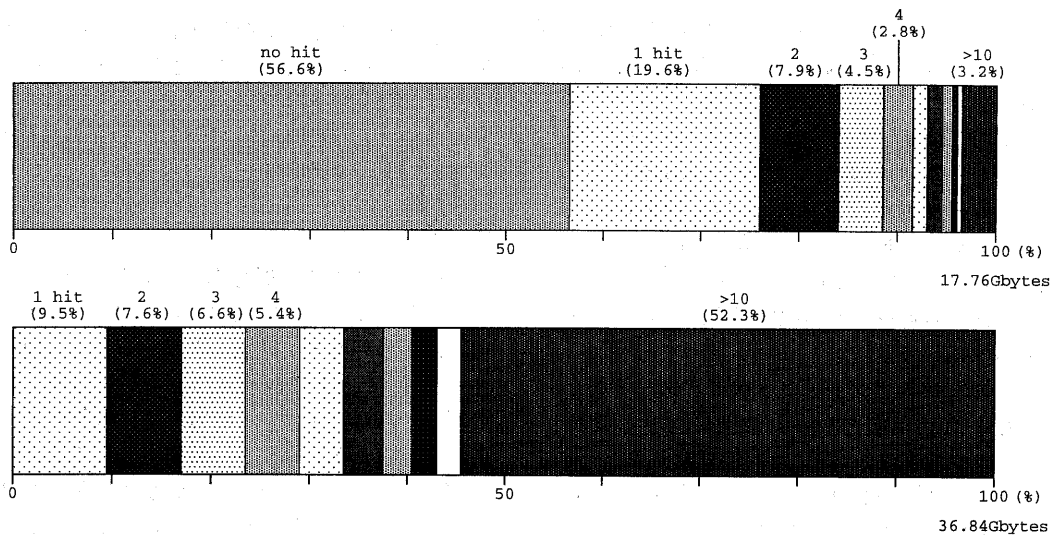


図 5: キャッシュ内データの利用率 (バイト換算)

WWWで公開している。このデータは1997年8月中旬から取り始めたため、それ以前の傾向は不明である。

この図を見ると、2時間平均のため最大値はならされてしまっているが、それでも平日のアクセスが集中する時期にはかなり応答時間が長くなっていることがわかる。キャッシュサーバへのアクセスが集中している時には、要求を出して最初のデータが届くまで、場合によっては2秒以上かかっている。5分おきの測定値を見ると、4秒を越えている場合もある。負荷が軽い時の応答時間が50ms程度であることを考えると、50~100倍となりかなり応答時間が増加していると言える。

原因の一つとしては、やはりメモリ不足が考えられる。現在 Squid のプロセスは150M~180Mのメモリを占有しており、その他のプロセスの大きさを考えると実メモリから溢れてしまっている。アクセスが集中すると、ネームサーバを引くためのプロセスも複数動き出すため、これによってスラッシングが発生していると考えられる。1997年9月から、メモリの使用量がより少ないバージョンに入れ換えてみたが、あまり効果は出ていない。

対策としては、実メモリを増やすことと、キ

ャッシュ容量の上限を下げることによって管理情報を減らすことが考えられる。Squidでは、キャッシュ内のオブジェクト(WWW上のページや画像等の総称)の管理情報として1オブジェクトにつき約100バイト必要であり[3]、実際当サーバで調べてみるとオブジェクト数約120万に対し約120MBが管理情報として常時メモリを占有していた。現状ではメモリの増設がすぐにはできないため、今後キャッシュの上限を減らすことによって管理情報を減らし、それによって応答時間に変化があるかどうかを観察していく予定である。

7 キャッシュ内データの利用率

約20GBあるキャッシュ容量のうち、どの程度の部分が実際にキャッシュヒットに貢献しているのかを調べたのが図5である。

これらの図は、1997年8月末日の時点でキャッシュに残っていた約18GB、120万個のオブジェクトについて、1997年7、8月の二箇月分のアクセスログから数えあげたヒット数と照合して算出した。厳密には、各オブジェクトについてそれがキャッシュに入った時刻を調べて、そこからのヒット数を数えるべきだが、解析処理時間の都合によりそこまでの調査はしていな

い。現在キャッシュ内データの生存期間は約50~60日で安定しているため、近似的に実際の傾向を示していると考えている。

上の図はヒット数別にオブジェクトサイズの合計を集計したグラフである。100%はキャッシュの総容量で、割合は(特定の回数ヒットしたオブジェクトのサイズの合計)/(キャッシュの総容量)で示されている。これはキャッシュのどの程度の容量がどの程度再利用されているかを示している。

下の図は各オブジェクトのサイズにそのヒット数を乗じた値をヒット数別に合計し集計したグラフである。100%は対象期間にキャッシュから取り出されたキャッシュヒット時の全トラフィックで、割合は(特定の回数ヒットしたオブジェクトサイズの合計) \times (ヒット回数)/(ヒット時の全トラフィック)で示されている。これはあるヒット回数のオブジェクトがどの程度ヒット時のトラフィックに貢献しているかを示している。

二つの図を併わせると、例えばキャッシュに入ってから1回だけヒットしたオブジェクトはキャッシュ全体の20%を占めており、この部分によって産み出されたトラフィックの、キャッシュヒットした全トラフィックに占める割合は約10%である、といったことがわかる。

上の図から、半分以上のデータは最初にキャッシュに入ってから一回もアクセスされておらず、無駄にキャッシュを占有しているということがわかる。もし仮に各データの未来のアクセス回数が予測できるとすれば、これらのアクセスされないデータはキャッシュしないで済むため、半分の10G程度のキャッシュ容量があれば足りることになる。また、全体の5%に満たない10回以上ヒットしたデータが、キャッシュヒット全体のトラフィックの半分以上を産み出していることがわかる。

文献[4]によれば、LRU(Least Recent Used)を用いたキャッシュアルゴリズムは、ある程度のキャッシュ容量があれば未来のアクセス頻度を予測できる場合に近いヒット率を達成できる。しかし Squid は厳密にLRUを実装していないため、キャッシュ容量が小さくなるとLRU方式

に比べてヒット率がかなり低下すると言われている。これを実際に検証するために、今後キャッシュが定常状態を保てる程度にゆっくりキャッシュ容量の上限を減らしてゆき、ヒット率や利用率の変化を観察する予定である。

8 おわりに

九州大学で約1年間稼働しているキャッシュサーバについて、ログを解析していくつかの側面からの評価と問題点の考察を行った。現在キャッシュヒット率は約60%と比較的高く、この観点からはキャッシュは効果的に動作していると考えられる。しかし、問題点としてアクセスが集中した時の応答時間の劣化が確認された。また、キャッシュ内データのかかなりの部分はキャッシュヒット率に貢献していないことがわかった。

今後は、キャッシュ容量の上限を減らすことによるヒット率、キャッシュ内データの利用率、およびメモリ占有量の低下に伴う応答時間の変化を観察する予定である。また、できれば対外トラフィックからWWWに関するトラフィック量を調べ、キャッシュサーバ経由のデータとそうでないデータの関係性を調査したい。キャッシュサーバ経由でないデータが多ければ、ユーザがキャッシュサーバをより積極的に使うような手段を考慮する必要がある。さらに、同程度のキャッシュ容量でLRUアルゴリズムを用いた場合のヒット率を、蓄積されたアクセスログを用いてシミュレーションするといったことも考えている。

参考文献

- [1] "Squid Internet Object Cache", <http://squid.nlanr.net/Squid/>, 1997.
- [2] "The Multi Router Traffic Grapher", <http://www.ee.ethz.ch/~oetiker/webtools/mrtg/mrtg.html>, 1997.
- [3] D. Wessels, "Release Notes for version 1.1 of the Squid cache", 1997.
- [4] 西川記史 他: "WWWトラフィック分析と分散キャッシュ", 情報処理学会研究報告 97-DSM-5, Vol. 97, No. 44, pp. 7-12, 1997.