基幹系WWWキャッシュサーバの運用実験について

5T－7

鍋島　公章
ＮＴＴソフトウェア研究所

1. はじめに

WWWにおいても，情報の複製によるスケーラビリティの向上が求められている．このための一つの機構として，キャッシュサーバが注目され，広く使われている．しかし，次のような項目が課題として残っている．本運用実験では，これらを確立することを目標にしている．

- 運用上のノウハウ．
- キャッシングに必要な計算機資源の予測方法．
- OS，サーバソフトウェアの最適化方法．
- キャッシュサーバの定量的な有効性評価方法．
- 有効なキャッシュサーバ間の協調運用法．

本稿では，今までの実験で得られた運用上のノウハウと，キャッシングに必要な計算機資源についての解析の中間報告を行う．

2. 実験環境

実験用のサーバ(cache.imnet.ad.jp)はSun Sparc Station-20 71(OSはSunOS 4.1.4)を使用し，512 MBのメモリとキャッシュ用ディスク20GB(4GB SCSI-Iディスク5台)を用意した．使用したキャッシュサーバ・ソフトウェアはSquid Internet Object Cache [2]である．これをIMnet (Inter-Ministry research information network) 大手町NOCのFDDIリングに接続した．

SunOSは，標準だとプロセスあたりの使用可能ファイルディスクリプタ(FD)数は256個である．これがボトルネックとなり，ユーザからのHTTPアクセスが頻繁にタイムアウトしていた．そのため，Sun DBEを導入し，使用可能最大FD数を2048個にした．これと同時に，OS内部変数の一部を変更した．MAXUSER(この値を増やすことにより，各種OS内領域が大きくなる)の値を64から128に増加させ，UDPパケット処理のオーバフローを抑制した．また，SOMAXCONN(ソケットコ・ネクション待ち最大数)は標準の5から64にまで大きくした．

An Experiment on Global WWW Cache Server Operation
Masaaki NABESHIMA (nabe@slab.ntt.co.jp)
NTT Software Laboratories
本研究は，科学技術庁の平成９年度科学技術振興調整費による「生活工学アプリケーション研究」の一環として行われている．

ただし，TCPチューニングとしてよく行われるTCP送受信バッファサイズの拡張は行わなかった．これは，OS内部メモリ(mbuf)管理テーブルがあふれ，通信ができなくなる可能性があるためである．標準の大きさのバッファサイズでは，確立されたTCPコネクション数が700強，標準の6倍の大きさだと，TCPコネクション数が100強でバッファテーブルあふれが発生した．

3. 統計

ここでは，97年7月17日の1日分のアクセスに関する統計情報を示す．Squidの使用していたプロセスサイズは276MBであった．このうち，128MBはアクセスの中継と，メモリ上でのオブジェクト(本稿におけるオブジェクトとはHTTPアクセスによって得られる情報をさす)のキャッシュに使われていた．この128MBの領域も，アクセスが多い日は使い切る場合があった．キャッシュディスク使用量は17GBで，iostatコマンドにより計測したキャッシュ用ディスクの使用率はピークで18%程度であった．1日のアクセス総数はICP[1]:84万アクセス，HTTP:40万アクセス(3.7GB)であり，ヒット率は，ICPベースで24%，HTTPベースで48%であった．また，オブジェクトのキャッシュ内での生存時間は約1週間であり，キャッシュ量とトラフィック量のバランスは良好であると思われる．

グラフ1のように，ピーク時では，30個程度のHTTPコネクション開始要求が待ち行列に入り，ファイルディスクリプタは300個程度使用していた．ICPリクエストの処理数は毎秒18リクエスト程度，HTTPリクエストは毎秒8リクエスト(80KB/秒)程度処理していた．これが，この実験サーバの限界であると思われる．

また，グラフ2のように，サーバ負荷のピーク時でも，キャッシュサーバ上で動いているHTTPサーバに，直接アクセスするのに必要な時間は0.1秒程度である．しかし，このHTTPサーバにキャッシュサーバ経由でアクセスすると，平均3秒以上必要としている．負荷の増加とともにキャッシュサーバでの中継レイテンシ(キャ

[1] Inter Cache Protocol: Cacheサーバ間の問合わせプロトコル

ッシュサーバでオブジェクトがヒットした場合は，厳密には，中継は行わないが，本稿ではこれも中継レイテンシと呼ぶ)の増加は非常に大きいといえる．

4. 実験によって得られたこと

ここでは，今まで得られた運用上の注意点についてのべる．

- Squid はログファイルを切替える際に，プロセスサイズの倍以上の仮想記憶領域を必要とする．
- 何らかの原因でサーバをリスタートした時に，キャッシュディスクからオブジェクトのメタ情報を再構築するのに約 12 時間かかる．この処理は重く，処理中はキャッシュサーバにおける HTTP 中継レイテンシが大きくなる．
- OS の限界付近で動いているため，不安定になりやすい．そのため，FDDI リングのような環境で使うには，マシンのダウンによるリング切断のリスクが大きい．
- Squidには，サーバ間で協調しあう機能があるが，現在，負荷を考慮したキャッシュサーバの選択機能がない．そのため，中継レイテンシが大きいが，アクセスは正常に受付けるキャッシュサーバは，それと協調して動いている他のキャッシュサーバにとって，大きなボトルネックになる．つまり，その中継レイテンシの大きいキャッシュサーバからオブジェクトを取り出すより，元のWWW サーバから直接オブジェクトを取り出す方が速い場合でも−ICP により，情報がそのキャッシュサーバにあることがわかると−squid は無条件にそのキャッシュサーバからオブジェクトを取得する．キャッシュサーバの負荷に注意を払い，中継レイテンシを小さく押さえ続ける必要がある．

5. おわりに

この実験により，チューニングの余地はまだ残っているが，ある程度キャッシュサーバの処理限界が把握できた．Sun Sparc Station-20 71 の場合，HTTP 処理は毎秒 8 アクセス(80KB/秒)程度が限界である．このため，例えば外部に対して T1 リンク(180KB/秒)を持つ組織に必要なキャッシュサーバは本実験で使用したサーバの3倍の処理能力が必要であるといえる(4 割のトラフィックがヒットすると仮定した場合，キャッシュサーバが処理しなければならない流量は約 250KB/秒であるため)．

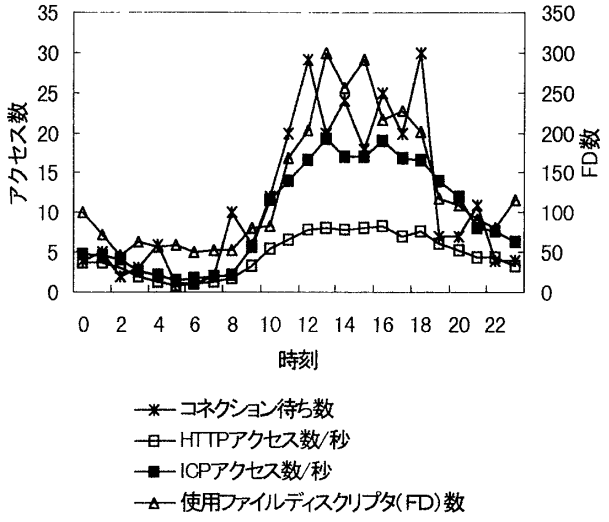
今後は，サーバのモニタリングの精度を高め，キャッシュサーバにおける中継レイテンシが何を要因にして大きくなるかを明らかにする予定である．

この実験の最新の情報は，cache.imnet.ad.jp のホームページ[1]を参照されたい．

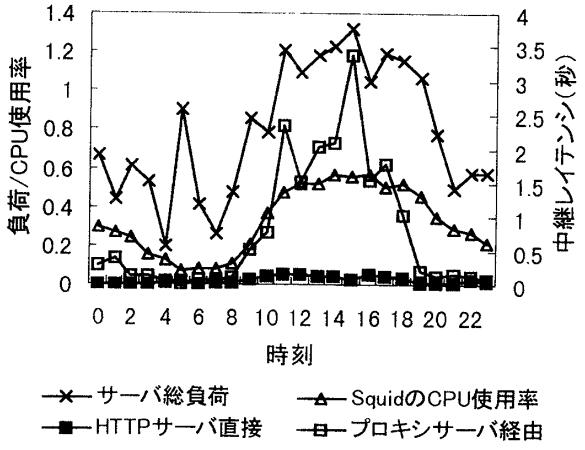
謝辞

不安定な実験サーバの面倒をみてくださる IMnet 東京 NOC チームの方々，三上リーダをはじめとするソ設グループ IMnet チームの方々に感謝します．

グラフ1: アクセス



グラフ2: 負荷と中継レイテンシ



参考文献

[1] 鍋島 公章，cache.imnet.ad.jp ホームページ，http://cache.imnet.ad.jp/

[2] Duane Wessels, Squid Internet Object Cache, http://squid.nlanr.net/Squid/

[3] Adrian Cockcroft (日本サンマイクロシステムズ監訳), Sparc & Solaris パフォーマンスチューニング