

12月か6月の末日（日本ではUTCより9時間進んでいるから、1月1日か7月1日の8時59分59秒）に1秒を挿入もしくは、引き抜くことで行われる。今まで挿入される“正のうるう秒”ばかりで、引き抜かれたことはない¹⁾。よく「次のうるう秒は、いつ入りますか？」という質問を受けるが、地球の自転によるわけで、実施数ヶ月前に国際度量衡局から連絡があるものの、それが無い時は「地球に聞いてください」というのが答えになる（表1）。

一方天文観測は、天文保時室の前身である東京天文台天文時部において、写真天頂筒（PZT: Photo Zenith Tube）を使い、地球の自転運動を±10 msの精度で観測していた。しかし、1988年国際的に地球回転の観測は、このような従来の光学による方位・位置観測から、電波、レーザーを使った高精度の距離観測（VLBI, 月・人工衛星レーザー等）に移行することになり、精度が原子時計（ 10^{-13} ）とかけ離れていることもあって、その年の5月でやめることになった。数年前から、光による精度の高い地球回転の観測にも対応できる、光赤外干渉計の計画も始まっている。

また天文時部はその昔、時刻を決定し維持する保時と、その時刻を通報する報時、無線報時による外国との時刻比較をする経度の3部門であった。三鷹キャンパスの敷地内に今でも国際報時所

の門柱が残っている。かつて三鷹から報時を行っていたこともあったのだが（1924～1951年）、標準電波による報時が小金井の電波研究所（現・通信総合研究所）に移ってから、天文保時室は報時の手段を持たなかった²⁾。NTPを始めるまでは、

2. WIDE (Widely Integrated Distributed Environments) プロジェクトとの出会い

今日の技術革新はとどまる所を知らない。その中でも最も動きが著しいのは計算機分野であることは論をまたないだろう。現在世界的に張り巡らされている計算機ネットワーク、インターネットも、元とも言えるイーサネット（Ethernet）が発表されてから、まだ20年がたったばかりである。日本でも大学、研究所のボランティア的な活動を中心に環境が整備され、昨今のブームののって広く一般にも知られるようになった。

このインターネット上で何ができるか、それを実現するためにはどんな問題点があり、解決するにはどんな技術が必要か、といった大規模広域ネットワークの総合的な研究と運用を行っているのが、WIDEプロジェクト³⁾である。

はじめはある雑誌のネットワークの解説記事⁴⁾が、天文台と計算機ネットワークというまったく畑の違う研究者、技術者を結び付けるきっかけになったのであるが、すぐにWIDEのNTPワーキング・グループというメーリング・リストが開設され、ネットワークの上で実際の問い合わせや議論が行われた。

さて、どうしてネットワークでつながれた計算機の時刻があっていると都合が良いのだろうか。まず、広域ネットワークのような分散環境で、ファイルやデータベースを共有して協調作業を行っている場合、ファイルの新旧や、データベースの更新時期を明確にできる事。次に正確な時計を使って、ネットワークの伝送経路の分析を行う事ができ、メールの配送などのサービスの効率向上が

年	月	日	ΔT	年	月	日	ΔT
1972	1	1	10	1982	7	1	21
1972	7	1	11	1983	7	1	22
1973	1	1	12	1985	7	1	23
1974	1	1	13	1988	1	1	24
1975	1	1	14	1990	1	1	25
1976	1	1	15	1991	1	1	26
1977	1	1	16	1992	7	1	27
1978	1	1	17	1993	7	1	28
1979	1	1	18	1994	7	1	29
1980	1	1	19	1996	1	1	30
1981	7	1	20				

ΔT は TAI - UTC 単位は秒

表1 TAIとUTCの関係

はかれる事。また、時刻の不確かさ、同期性のな
 きのために出来なかった、例えば交通機関、金融
 機関の仕事への応用が可能になる事。そのため、
 計算機の研究者は正確な時刻を欲しがっていた。

一方、天文保時室は保時している時刻をネット
 ワーク上の時刻同期に活用できるという事で、
 1992年春から WIDE プロジェクトとの共同研究
 が始まった。

そのころは、天文台でもようやくネットワーク
 環境が整い、先に述べた国内外の関係機関との時
 計比較データ交換を電子メールで始めた時期であ
 った。筆者も UNIX 環境に触れ始めたばかりだっ
 たので、慣れるまで苦労したが、電子メールを使
 ったの会議や、文書や図、データをネットワーク
 上で持ち寄ったの出版等、驚くことばかりであっ
 た。

3. ネットワーク上の時刻同期

実際に測ってみるとわかるが、それぞれの計算
 機の内蔵時計の進み具合は様でなく、複数の計
 算機の時計は、そのままではすべて異なった時刻
 を指していると考えて良い。もちろんこのような
 問題を解決するために UNIX に基づく計算機ネ
 ットワークでは、いくつかの方法がある。

ただ単に、2台の計算機同士の時刻同期であ
 れば rdate というプログラムで、一方の計算機に他
 方の時刻を合わせることができる。しかし、この
 やり方には、秒単位で一挙に時刻を合わせるため
 時刻に連続性がなくなる事、合わせようとする計
 算機の時刻が本当に合っているかわからない事等
 の問題があって、ネットワークの時刻同期には不
 向きである。

もう一つは、timed を使う方法である。これは、
 地域的に限られたネットワークにつながれた計算
 機の持つ時刻の平均を取り、これに合わせるやり
 方である。当然時刻を合わせようとするそれぞれの
 計算機でこのプログラムを働かせなければなら
 ないが、“デーモン”と呼ばれる、計算機立ち上げ

時に起動されバックグラウンドに常駐するシステ
 ム処理用プログラムであるため、一度動かしてし
 まえば特に意識することない。また計算機の時計
 の進み具合を早くしたり遅くしたりして時刻を合
 わせられるため、連続性を保った時計合わせがで
 きる。ただし、この方法でもそのネットワーク全
 体の時刻がずれていた場合には間違った時刻に合
 せてしまう事がありうる。

最後の一つが NTP による時刻同期システム
 (以下 NTP システム) で、インターネットのよう
 な広域ネットワークにおいて、時刻を同期させる
 ために開発されたプロトコル (通信方法の取り決
 め) である。NTP では UTC に同期した時計を持
 つ計算機が、ネットワークにつながっていること
 を前提としていて、NTP デーモンを働かせてお
 けば自動的に計算機の時刻を UTC に同期させて
 くれる。また、timed と同様時刻だけでなく、時計
 の進み具合も調整してくれる。

保時室では、NTP システムで前提となる UTC
 に同期したセシウム原子時計がすぐに利用できる
 ので、広域ネットワークにも対応できるこの方式
 を採用することにした。

4. NTP のしくみ

NTP はネットワーク上でクライアント-サー
 バーモデルで働くため、この時刻情報を提供する
 計算機を NTP サーバーと呼ぶ (図 2)。

NTP サーバーは、UTC に同期して最も精度の
 高いサーバーから stratum (階層) 1, 2, 3, ……

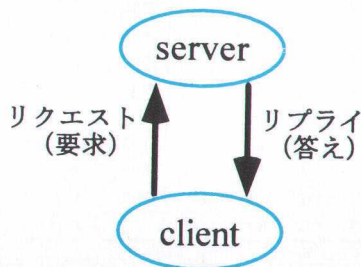


図 2—サーバーとクライアント

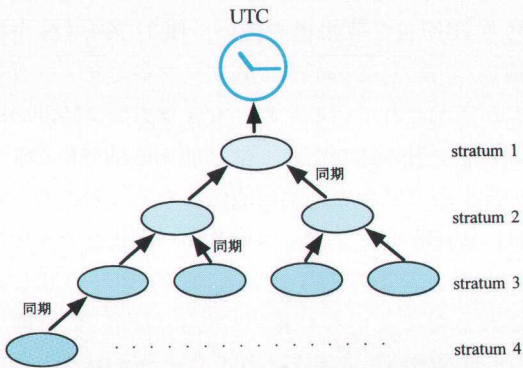


図3 NTPサーバーのツリー構造

とランク付けされたツリー構造になっている。stratum 1のサーバーにとって、stratum 2のサーバーはクライアントになり、同様にstratum 2のサーバーにとって、stratum 3のサーバーはクライアントになる。こうして、階層の高い方から低い方へ時刻の受け渡しをして、ネットワーク全体の時刻同期を取るしくみであるから、その精度・確度もそれに応じていると考えて良い(図3)。

stratum 1サーバーをUTCに同期させるには、一般に短波報時、(最近になって)GPSが用いられ、これらの受信機から時刻情報を受け取っている。

アメリカでもWWV, WWVH等の短波報時(日本ではJJYにあたる)や、GPS受信機を用いた物がほとんどで、UTCに同期したセシウム原子時計を使った物は少ない。1994年春の天文台のNTPサーバー1号機立ち上げ前は、日本国内では太平洋を越えてアメリカの精度の良い短波報時を元にしたstratum 1サーバーを参照して、stratum 2サーバーを維持している状態であった。

実際にNTPサーバーによって時刻を同期する手順を考えてみると次のようになる。

クライアントが時計を合わせるのに必要となる、ある時刻におけるサーバーの時計との差

$$\Delta T = t'c - t's \text{ は、次の様に計算される (図4).}$$

時刻 $t1$ にクライアントからサーバーにむけて時刻情報の要求をする。これがサーバーに届いた時刻 $t2$ とクライアントに向け情報を送り出す時刻 $t3$ を一緒にして送り返す。

こうして、クライアントに時刻情報が届いた時刻を $t4$ とすると、ネットワーク上の伝送遅延が行き帰り等でしいと仮定すれば、 ΔT は次の式で求められる。

$$\Delta T = t'c - t's = \frac{t1 + t4}{2} - \frac{t2 + t3}{2}$$

また、ネットワーク通信による遅延 d は

$$d = (t4 - t1) - (t3 - t2)$$

で求められる。

これを何度か行って ΔT や d の分散を計算し、この値の変化を見て通信経路の安定度、時刻精度を判断してデータの取捨選択を行う。また複数のNTPサーバーが参照できる場合、得られた時刻情報のばらつきをstratum番号による重みづけを行って評価し、精度に問題がありそうなサーバーを不適格として参照しないようにする事もある。

この様な一連の動作は前述のNTPデーモンと呼ばれる専用処理プログラムによって、自動的にかつ定期的に行われるので、ユーザーが意識するこ

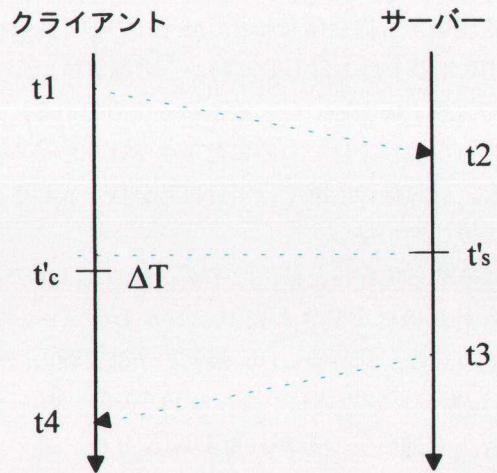


図4 NTPによる時刻同期の手順

となく、計算機内部の時計を適切なサーバーに同期させることができる。その際 UNIX のシステムコールを使い、時計の進み具合を遅くしたり早めたりして、時刻を緩やかに同期させることができる。

NTP は UNIX 上で動くフリーソフトで、主要な FTP サイトに大抵置いてある。stratum 1 サーバーでなければ、UNIX が動いてインターネットにつながっていれば、このソフトウェアをインストールしていくつかの設定をするだけで、NTP による時刻同期が実現できる。現在のバージョンは xntpd 3.4 である。興味を持たれた方はぜひ NTP を立ちあげてみていただきたい。

5. stratum 1 NTP サーバーの実現

セシウム原子時計を使って stratum 1 NTP サーバーを実現するために用意した装置は次のようなものである。まずセシウム原子時計から出力される 1 秒パルスと 1 MHz 信号を計算機に取り込まれる時刻情報にするため、2 組のバイナリカウンタで積算して、秒、マイクロ秒の 32 bit データに変換するカウント回路を製作した。これはセシウム原子時計自体には時刻の情報を持っていないため(最新のものは出力出来るようになったが)、それぞれを計数して、UNIX で標準的に使われる時刻を格納する構造体 struct timeval の秒 : tv_sec (1970 年の 1 月 1 日 UT 0 時からの積算秒)と、マイクロ秒 : tv_usec に一致させるためである。人間には積算秒ではいつの事だかさっぱりわからないが、計算機の内部では年月日時分秒で表わされるより処理し易い。

こうやって作った秒、マイクロ秒の 32 bit データを計算機に実装されたパラレル I/O ボードから読み込む。計算機には、最初の予備実験の段階では、時刻を読み込む手持ちの PC-9801 とイーサネットを通じてつながれた UNIX ワークステーション NEWS で 1 つのサーバーとして機能させていた。これを 1 台にまとめるために、計算機に

はボードの拡張が容易で、UNIX 互換 OS が動かせる IBM-PC 互換機を使い、BSD 系の OS を採用した。また、この OS は基本ソフトのソースコードが開示されているため、セシウム原子時計の時刻情報を読み込むために付け加える部分を OS の内部に取り込むことが可能になる。もちろんこれは、WIDE のメンバーがいるから出来たことで、プログラムと言えば BASIC と FORTRAN しかさわったことのない筆者にとっては、C 言語で OS の内部にまで手を入れてしまうというのは信じられない事であった。

予備実験用のシステムを 2 組使って、ネットワーク遅延の変化、計算機の内蔵時計の分解能調査等をした結果、同一構内のネットワーク (例えば国立天文台三鷹キャンパス内)であれば、NTP を使って数十マイクロ秒の精度で時刻同期が可能である見込みがたった。

しかし、天文台三鷹と東京工業大学 (目黒区)の間等、いくつかの回線を経由するもっと広域のネットワークになると、NTP 単独ではネットワーク遅延等で時刻同期の精度はサブミリ秒程度しか得られない。また、運用していくうちに NTP システムの問題点も見えてきた。

その一つは、まだ国内の stratum 1 サーバーは少数であり、これに多数の問い合わせが集中した場合、ネットワークの通信量が増えて時刻精度が落ちるばかりか、他の通信にも影響を与えることになりかねない。

これを防ぐためには、安価で簡便な NTP サーバーを開発し、少しでも多くの stratum 1 サーバーを稼動させることにより、問い合わせを分散する必要がある。また、時刻精度がそれほど必要ない計算機が stratum 1 サーバーを直接参照することがないように、stratum の適正な運用も必要となって来るであろう。

6. GPS を使った NTP サーバー

現在、精度の高い時刻情報をもっとも手軽に得

られるのはGPSである。もともとGPSは、精密な位置を求めるためのシステムであるが、GPS衛星から送られてくる位置決めのために必要な情報のなかには、搭載されたセシウム原子時計の時刻情報も含まれている。また、GPS受信機はセシウム原子時計の様な設備や手間がいらず、アンテナを接続し電源を入れるだけでUTCに同期した時刻が得られる。そこでGPS受信機を使ってNTPサーバーを立ちあげれば、外部ネットワークが簡単に引けない孤立した環境（例えば観測所等）でも、衛星が受信可能な場所であれば、NTPシステムでの時刻同期が可能になる。

最新のNTPサーバー用プログラム・パッケージには、市販されている何種類かのGPS受信機が標準でサポートされていて、海外だけでなく国内でもGPSを使ったNTPサーバーが立ちあげられている。

しかし、我々はセシウム原子時計の代わりに、1秒信号の出力できるGPS受信機(古野電気GN-72T)と、1MHz信号は安価な水晶発振器で代用して、これを先に製作したNTPサーバーのカウンタ回路に同じように入力して、GPSを使ったNTPサーバーを構成した。

前述したカウンタ回路には、1秒信号と1MHz信号の同期がとれていない場合、秒信号でマイクロ秒カウンタをリセットできる回路を組み込んである。もし、水晶発振器の1MHz信号が温度変化のため、 10^6 マイクロ秒≠1秒になってもそれ以上の誤差は出ない。

実験ではGPS受信機と水晶発振器の精度にもよるが、数百～数十マイクロ秒の時刻精度を実現できそうである。これ以上の精度が必要な場合には、例えば天文台のセシウム原子時計とGPS受信機の時刻差の測定値をネットワークで公表すれば、リアルタイムとはいかないが、短時間でマイクロ秒程度の時刻精度が得られるはずである。

7. おわりに

1994年春から1号機は、「世界で初めて、UTCに同期したセシウム原子時計直結のStratum 1サーバー」として本格運用を始め(IPアドレスは133.40.40.133と回文になっている。ちなみに133はセシウムの原子量である)、プロジェクトは今も少しづつではあるが前進している。今年1月には、1号機のOSと共に時刻取り込みソフトのバージョンアップを行い、2号機もまだ非公開ではあるがStratum 1として動きはじめた。

また、すばる望遠鏡でもほとんどの計算機の時刻同期に、先に述べたGPSベースのNTPサーバーで、水晶発振器のフィードバックに工夫した改良版が使われる事になっている。

最後になったが、NTPをはじめのきっかけを作ってくれた大橋正健さん、また天文保時室とWIDEのメンバー、特にNTPワーキング・グループで中心となって我々を引っ張ってくれている大野浩之さんに感謝したい。それから、これまでにネットワークを育て、支えてきた多くの人々に感謝したい。

参考文献

- 1) 国立天文台編, 1995, 理科年表平成8年度版, P 168
- 2) 青木信仰, 1982, 時と暦
- 3) WIDE Project 編, bit 別冊インターネット参加の手引き 1994, 「時刻基準の設置」, P 263
- 4) 山口 英, 1992, 「NTP」, UNIX MAGAZINE, 2月号, P 29

Time Keeping Office and NTP Servers

Koh MATSUDA

National Astronomical Observatory

The NTP (=Network Time Protocol) is a method to synchronize clocks of UNIX-based computers via computer networks. We introduce the history of our research on the NTP server and their present status. Also we discuss on the fundamental relations between the deployment of computer networks and the clock synchronization.