Network Working Group
Request for Comments: 957

D.L. Mills
M/A-COM Linkabit

September 1985

Experiments in Network Clock Synchronization

Status of this Memo

This RFC discusses some experiments in clock synchronization in the ARPA-Internet community, and requests discussion and suggestions for improvements. Distribution of this memo is unlimited.

Table of Contents

List of Figures

List of Tables

1. Introduction

One of the services frequently neglected in computer network design is a high-quality, time-of-day clock capable of generating accurate timestamps with small residual errors compared to intrinsic one-way network delays. Such a service would be useful for tracing the progress of complex transactions, synchronizing cached data bases, monitoring network performance and isolating problems.

Several mechanisms have been specified in the Internet protocol suite to record and transmit the time at which an event takes place, including the ICMP Timestamp message [6], Time Protocol [7], Daytime protocol [8] and IP Timestamp option [9]. A new Network Time Protocol [12] has been proposed as well. Additional information on network time synchronization can be found in the References at the end of this document. Synchronization protocols are described in [3] and [12] and synchronization algorithms in [2], [5], [10] and [11]. Experimental results on measured roundtrip delays in the Internet are discussed in [4]. A comprehensive mathematical treatment of clock synchronization can be found in [1].

Several mechanisms have been specified in the Internet protocol suite to record and transmit the time at which an event takes place, including the ICMP Timestamp message [6], Time protocol [7], Daytime protocol [8] and IP Timestamp option [9]. Issues on time synchronization are discussed in [4] and synchronization algorithms in [2] and [5]. Experimental results on measured roundtrip delays in the Internet are discussed in [2]. A comprehensive mathematical treatment of the subject can be found in [1], while an interesting discussion on mutual-synchonization techniques can be found in [10].

There are several ways accurate timestamps can be generated. One is to provide at every service point an accurate, machine-readable clock synchronized to a central reference, such as the National Bureau of Standards (NBS). Such clocks are readily available in several models ranging in accuracies of a few hundred milliseconds to less than a millisecond and are typically synchronized to special ground-based or satellite-based radio broadcasts. While the expense of the clocks themselves, currently in the range $300 to $3000, can often be justified, all require carefully sited antennas well away from computer-generated electromagnetic noise, as well as shielded connections to the clocks. In addition, these clocks can require a lengthy synchonization period upon power-up, so that a battery-backup power supply is required for reliable service in the event of power interruptions.

If the propagation delays in the network are stable or can be predicted accurately, timestamps can be generated by a central server, equipped with a clock such as described above, in response to requests from remote service points. However, there are many instances where the trans-network delay to obtain a timestamp would be intolerable, such as when timestamping a message before transmission. In addition, propagation delays are usually not predictable with precisions in the order required, due to probabilistic queuing and channel-contention delays.

In principle, a clock of sufficient accuracy can be provided at each service point using a stable, crystal-controlled clock which is corrected from time to time by messages from a central server. Suitable inexpensive, crystal-controlled clock interfaces are available for virtually any computer. The interesting problem remaining is the design of the synchronization algorithm and protocol used to transmit the corrections. In this document one such design will be described and its performance assessed. This design has been incorprated as an integral part of the network routing and control protocols of the Distributed Computer Network (DCnet) architecture [5], clones of which have been established at several sites in the US and Europe. These protocols have been in use since 1979 and been continuously tested and refined since then.

2. Design of the Synchronization Algorithm

The synchronization algorithm is distributed in nature, with protocol peers maintained in every host on the network. Peers communicate with each other on a pairwise basis using special control messages, called Hello messages, exchanged periodically over the ordinary data links between them. The Hello messages contain information necessary for each host to calculate the delay and offset between the local clock of the host and the clock of every other host on the network and are also used to drive the routing algorithm.

The synchronization algorithm includes several features to improve the accuracy and stability of the local clock in the case of host or link failures. In following sections the design of the algorithm is summarized. Full design details are given in [5] along with a formal description of the Hello protocol.

2.1. The Logical Clock

In the DCnet model each service point, or host, is equipped with a hardware clock, usually in the form of an off-the-shelf interface. Using this and software registers, a logical clock is constructed including a 48-bit Clock Register, which increments at a 1000 Hz rate, a 32-bit Clock-Adjust Register, which is used to slew the Clock Register in response to raw corrections received over the net, and a Counter Register, which is used in some interface designs as an auxilliary counter. The configuration and decimal point of these registers are shown in Figure 1.

```
        Clock Register

        0               16               32
        +---------------+---------------+---------------+
        |               |               |               |
        +---------------+---------------+---------------+
                                        A
                                  decimal point

        Clock-Adjust Register

                        0               16
                        +---------------+---------------+
                        |               |               |
                        +---------------+---------------+
                                        A
                                  decimal point

        Counter Register

                        0              16
                        +---------------+
                        |               |
                        +---------------+
                                        A
                                  decimal point
```

Figure 1. Clock Registers

The Clock Register and Clock-Adjust Register are implemented in memory. In typical clock interface designs such as the DEC KMV11-A the Counter Register is implemented in the interface as a buffered counter driven by a crystal oscillator. A counter overflow is signalled by an interrupt, which results in an increment of the Clock Register at bit 15 and the propagation of carries as required. The time of day is determined by reading the Counter Register, which does not disturb its counting process, and adding its value to that of the Clock Register with decimal points aligned.

In other interface designs such as the simple LSI-11 event-line mechanism, each tick of the clock is signalled by an interrupt at intervals of 10, 16-2/3 or 20 ms, depending on interface and clock source. When this occurs the appropriate number of milliseconds, expressed to 32 bits in precision, is added to the Clock Register with decimal points aligned.

It should be noted at this point that great care in operating system design is necessary in order to preserve the full accuracy of timestamps with respect to the application program, which must be protected from pre-emption, excessive device latencies and so forth. In addition, the execution times of all sequences operating with the interrupt system disabled must be strictly limited. Since the PDP11 operating system most often used in the DCnet (the "Fuzzball" operating system) has been constructed with these considerations foremost in mind, it has been especially useful for detailed network performance testing and evaluation. Other systems, in particular the various Unix systems, have not been found sufficiently accurate for this purpose.

Left uncorrected, the host logical clock runs at the rate of its intrinsic oscillator, whether derived from a crystal or the power frequency. The correction mechanism uses the Clock-Adjust Register, which is updated from time to time as raw corrections are received. The corrections are computed using roundtrip delays and offsets derived from the routing algorithm, described later in this document, which are relatively noisy compared to the precision of the logical clock. A carefully designed smoothing mechansim insures stability, as well as isolation from large transients that occur due to link retransmissions, host reboots and similar disruptions.

2.2. Linear Phase Adjustments

The correction is introduced as a signed 32-bit integer in milliseconds. If the magnitude of the correction is less than 128 ms, the low-order 16 bits replaces bits 0-15 in the Clock-Adjust register. At suitable intervals, depending on the jitter of the intrinsic oscillator, the value of this register is divided by a fixed value, forming a quotient which is first added to the Clock Register, then subtracted from the Clock-Adjust Register. This technique has several advantages:

1. The clock never runs backwards; that is, successive timestamps always increase monotonically.

2. In the event of loss of correction information, the clock slews to the last correction received.

3. The rate of slew is proportional to the magnitude of the last correction. This allows rapid settling in case of large corrections, but provides high stability in case of small corrections.

4. The sequence of computations preserves the highest precision and minimizes the propagation of round-off errors.

Experience has indicated the choice of 256 as appropriate for the dividend above, which yields a maximum slew-rate magnitude less than 0.5 ms per adjustment interval and a granularity of about 2.0 microseconds, which is of the same order as the intrinsic tolerance of the crystal oscillators used in typical clock interfaces. In the case of crystal-derived clocks, an adjustment interval of four seconds has worked well, which yields a maximum slew-rate magnitude of 125 microseconds per second. In the case of power-frequency clocks or especially noisy links, the greatly increased jitter requires shorter adjustment intervals in the range of 0.5 second, which yields a maximum slew-rate magnitude of 1.0 ms per second.

In most cases, independent corrections are generated over each link at intervals of 30 seconds or less. Using the above choices a single sample error of 128 ms causes an error at the next sample interval no greater than about 7.5 ms with the longer adjustment interval and 30 ms with the shorter. The number of adjustment intervals to reduce the residual error by half is about 177, or about 12 minutes with the longer interval and about 1.5 minutes with the shorter. This completely characterizes the linear dynamics of the mechanism.

2.3. Nonlinear Phase Adjustments

When the magnitude of the correction exceeds 128 ms, the possiblity exists that the clock is so far out of synchronization with the reference host that the best action is an immediate and wholesale replacement of Clock Register contents, rather than a graduated slewing as described above. In practice the necessity to do this is rare and occurs when the local host or reference host is rebooted, for example. This is fortunate, since step changes in the clock can result in the clock apparently running backward, as well as incorrect delay and offset measurements of the synchronization mechanism itself.

However, it sometimes happens that, due to link retransmissions or occasional host glitches, a single correction sample will be computed with magnitude exceeding 128 ms. In practice this happens often enough that a special procedure has been incorporated into the design. If a sample exceeding the limit is received, its value is saved temporarily and does not affect the Clock-Adjust Register. In addition, a timer is initialized, if not already running, to count down to zero in a specified time, currently 30 seconds.

If the timer is already running when a new correction sample with magnitude exceeeding 128 ms arrives, its value and the saved sample value are averaged with equal weights to form a new saved sample value. If a new correction sample arrives with magnitude less than 128 ms, the timer is stopped and the saved sample value discarded. If the timer counts down to zero, the saved sample value replaces the contents of the Clock Register and the Clock-Adjust Register is set to zero. This procedure has the effect that occasional spikes in correction values are discarded, but legitimate step changes are prefiltered and then used to reset the clock after no more than a 30-second delay.

3. Synchronizing Network Clocks

The algorithms described in the previous section are designed to achieve a high degree of accuracy and stability of the logical clocks in each participating host. In this section algorithms will be described which synchronize these logical clocks to each other and to standard time derived from NBS broadcasts. These algorithms are designed to minimize the cumulative errors using linear synchronizing techniques. See [10] for a discussion of algorithms using nonlinear techniques.

3.1. Reference Clocks and Reference Hosts

The accuracy of the entire network of logical clocks depends on the accuracy of the device used as the reference clock. In the DCnet clones the reference clock takes the form of a precision crystal oscillator which is synchronized via radio or satellite with the NBS standard clocks in Boulder, CO. The date and time derived from the oscillator can be sent continuously or read upon command via a serial asynchronous line. Stand-alone units containing the oscillator, synchronizing receiver and controlling microprocessor are available from a number of manufacturers.

The device driver responsible for the reference clock uses its serial-line protocol to derive both an "on-time" timestamp relative to the logical clock of the reference host and an absolute time encoded in messages sent by the clock. About once every 30 seconds the difference between these two quantities is calculated and used to correct the logical clock according to the mechanisms described previously. The corrected logical clock is then used to correct all other logical clocks in the network. Note the different nomenclature: The term "reference clock" applies to the physical clock itself, while the term "reference host" applies to the logical clock of the host to which it is connected. Each has an individual address, delay and offset in synchronizing messages.

There are three different commercial units used as reference clocks in DCnet clones. One of these uses the LF radio broadcasts on 60 KHz from NBS radio WWVB, another the HF radio broadcasts on 5, 10 and 15 MHz from NBS radio WWV or WWVH and the third the UHF broadcasts from a GOES satellite. The WWVB and GOES clocks claim accuracies in the one-millisecond range. The WWV clock claims accuracies in the 100-ms range, although actual accuracies have been measured somewhat better than that.

All three clocks include some kind of quality indication in their messages, although of widely varying detail. At present this indication is used only to establish whether the clock is synchronized to the NBS clocks and convey this information to the network routing algorithm as described later. All clocks include some provision to set the local-time offset and propagation delay, although for DCnet use all clocks are set at zero offset relative to Universal Time (UT). Due to various uncertaincies in propagation delay, serial-line speed and interrupt latencies, the absolute accuracy of timestamps derived from a reference host equipped with a WWVB or GOES reference clock is probably no better than a couple of milliseconds.

3.2. Distribution of Timing Information

The timekeeping accuracy at the various hosts in the network depends critically on the precision whith which corrections can be distributed throughout the network. In the DCnet design a distributed routing algorithm is used to determine minimum-delay routes between any two hosts in the net. The algorithms used, which are described in detail in [5] and only in outline form here, provide reachability and delay information, as well as clock offsets, between neighboring hosts by means of periodic exchanges of routing packets called Hello messages. This information is then incorporated into a set of routing tables maintained by each host and spread throughout the network by means of the Hello messages.

The detailed mechanisms to accomplish these functions have been carefully designed to minimize timing uncertaincies. For instance, all timestamping is done in the drivers themselves, which are given the highest priority for resource access. The mechanism to measure roundtrip delays on the individual links is insensitive to the delays inherent in the processing of the Hello message itself, as well as the intervals between transmissions. Finally, care is taken to isolate and discard transient timing errors that occur when a host is rebooted or a link is restarted.

The routing algorithm uses a table called the Host Table, which contains for each host in the network the computed roundtrip delay and clock offset, in milliseconds. In order to separately identify each reference clock, if there is more than one in the network, a separate entry is used for each clock, as well as each host. The delay and offset fields of the host itself are set to zero, as is the delay field of each attached reference clock. The offset field of each attached reference clock is recomputed periodically as described above.

Hello messages containing a copy of the Host Table are sent periodically to each neighbor host via the individual links connecting them. In the case of broadcast networks the Hello message is broadcast to all hosts sharing the same cable. The Hello message also contains a timestamp inserted at the time of transmission, as well as information used to accurately compute the roundtrip delay on point-to-point links.

A host receiving a Hello message processes the message for each host in turn, including those corresponding to the reference clocks. It adds the delay field in the message to the previously determined roundtrip link delay and compares this with the entry already in its Host Table. If the sum is greater than the delay field in the Host

Table, nothing further is done. If the sum is less, an update procedure is executed. The update procedure, described in detail in [5], causes the new delay to replace the old and the routing to be amended accordingly.

The update procedure also causes a new correction sample to be computed as the difference between the timestamp in the Hello message and the local clock, which is used to correct the local clock as described above. In addition, the sum of this correction sample plus the offset field in the Hello message replaces the offset field in the Hello Table. The effect of these procedures is that the local clock is corrected relative to the neighbor clock only if the neighbor is on the path of least delay relative to the selected reference clock. If there is no route to the reference clock, as determined by the routing algorithm, no corrections are computable and the local clock free-runs relative to the last received correction.

As the result of the operation of the routing algorithm, all local clocks in the network will eventually stabilize at a constant offset relative to the reference clock, depending upon the drift rates of the individual oscillators. For most applications the offset is small and can be neglected. For the most precise measurements the computed offset in the Host Table entry associated with any host, including the reference clock, can be used to adjust the apparent time relative to the local clock of that host. However, since the computed offsets are relatively noisy, it is necessary to smooth them using some algorithm depending upon application. For this reason, the computed offsets are provided separately from the local time.

4. Experimental Validation of the Design

The original DCnet was established as a "port expander" net connected to an ARPAnet IMP in 1978. It was and is intended as an experimental testbed for the development of protocols and measurement of network performance. Since then the DCnet network-layer protocols have evolved to include multi-level routing, logical addressing, multicasting and time synchronization. Several DCnet clones have been established in the US and Europe and connected to the DARPA Internet system. The experiments described below were performed using the DCnet clone at Linkabit East, near Washington, DC, and another at Ford Motor Division, near Detroit, MI. Other clones at Ford Aerospace and the Universities of Maryland and Michigan were used for calibration and test, while clones at various sites in Norway and Germany were used for occasional tests. Additional

ARPANET gateways of the WIDEBAND/EISN satellite system were also included in the experiments in order to determine the feasability of synchronizing clocks across the ARPANET.

There were four principal issues of interest in the experiments:

1. What are the factors affecting accuracy of a network of clocks using the power grid as the basic timing source, together with corrections broadcast from a central point?

2. What are the factors affecting accuracy of a network of clocks synchronized via links used also to carry ordinary data.

3. How does the accuracy of the various radio clocks - WWVB, GOES and WWV compare?

4. What is the best way to handle disruptions, such as a leap second?

These issues will be discussed in turn after presentation of the experiment design and execution.

4.1. Experiment Design

Figure 2 shows the configuration used in a series of tests conducted during late June and early July of 1985. The tests involved six hosts, three reference clocks and several types of communication links. The tests were designed to coincide with the insertion of a leap second in the standard time broadcast by NBS, providing an interesting test of system stability in the face of such disruptions. The test was also designed to test the feasability of using the power grid as a reference clock, with corrections broadcast as described above, but not used to adjust the local clock.

```
      ARPAnet                              |
      - - - - - - - - - - - - - - - - - -  |  - - - - - - - - - -
                                       56K |
      +---------+     +---------+     +----+----+ 1.2 +---------+
      |   WWV   | 1.2 |         | 4.8 |         +-----+  WWVB   |
      |  radio  +-----+  DCN6   +-----+  DCN1   |async|  radio  |
      |  clock  |async|         |DDCMP|         +--+  |  clock  |
      +---------+     +---------+     +----+----+  |  +---------+
                       Ethernet            |       |
      DCnet     ===o===============o=======o===    | 1822/DH
                   |               |               |
              +----+----+     +----+----+     +----+----+
      power   |         |     |         |     |         |    power
      freq <--+  DCN3   |     |  DCN5   |     |  DCN7   +--> freq
      60 Hz   |         |     |         |     |         |    60 Hz
              +---------+     +----+----+     +---------+
                               9.6 |
      - - - - - - - - - - - - - -  |  - - - - - - - - - - - - - -
                                   | DDCMP
                              +----+----+     +---------+
                              |         | 1.2 |  GOES   |
      FORDnet                 |  FORD1  +-----+satellite|
                              |         |async|  clock  |
                              +---------+     +---------+
```

Figure 2. Network Configuration

Only those hosts and links directly participating in the tests are shown in Figure 2. All hosts shown operate using the DCnet protocols and timekeeping algorithms summarized in this document and detailed in [5]. The DCnet hosts operate as one self-contained net of the Internet systems, while the FORDnet hosts operate as another with distinct net numbers. The gateway functions connecting the two nets are distributed in the DCN5 and FORD1 hosts and the link connecting them. This means that, although the clock offsets of individual DCnet hosts are visible to other DCnet hosts and the clock offsets of individual FORDnet hosts are visible to other FORDnet hosts, only the clock offset of the gateway host on one net is visible to hosts on the other net.

In Figure 2 the links are labelled with both the intrinsic speed, in kilobits per second, as well as the link protocol type. The DDCMP links use microprocessor-based DMA interfaces that retransmit in case of message failure. The 1822/DH link connecting DCN1 and DCN7 operates at DMA speeds over a short cable. The Ethernet link uses

DMA interfaces that retransmit only in case of collisions. The asynchronous links are used only to connect the reference clocks to the hosts over a short cable.

While all hosts and links were carrying normal traffic throughout the test period, the incidence of retransmissions was very low, perhaps no more than a few times per day on any link. However, the DDCMP link protocol includes the use of short control messages exhanged between the microprocessors about once per second in the absence of link traffic. These messages, together with retransmissions when they occur, cause small uncertaincies in Hello message delays that contribute to the total measurement error. An additional uncertaincy (less than 0.5 per-cent on average) in Hello message length can be introduced when the link protocol makes use of character-stuffing or bit-stuffing techniques to achieve code transparency, such as with the LAPB link-level protocol of X.25. However, the particular links used in the tests use a count field in the header, so that no stuffing is required.

Although the timekeeping algorithms have been carefully designed to be insensitive to traffic levels, it sometimes happens that an intense burst of traffic results in a shortage of memory buffers in the various hosts. In the case of the Ethernet interfaces, which have internal buffers, this can result in additional delays while the message is held in the interface pending delivery to the host. Conditions where these delays become significant occur perhaps once or twice a day in the present system and were observed occasionally during the tests. As described above, the correction-sample processing incorporates a filtering procedure that discards the vast majority of glitches due to this and other causes.

4.2. Experiment Execution

The series of experiments conducted in late June and early July of 1985 involved collecting data on the delays and offsets of the six hosts and three reference clocks shown in Figure 2. In order to accomplish this, a special program was installed in a Unix 4.2bsd system connected to the Ethernet link but not shown in Figure 2. The program collected each 128-octet Hello message broadcast from DCN1 every 16 seconds and appended it bit-for-bit to the data base. The total volume of raw data collected amounted to almost 0.7 megabyte per day.

The raw Hello-message data were processed to extract only the timestamp and measured clock offsets for the hosts shown in Table 1 and then reformatted as an ASCII file, one line per Hello message.

Host Name	Clock ID	Drift (ppm)	Experiment Use
DCN1	WWVB	-2.5	WWVB reference host
DCN3	-	60-Hz	power-grid (unlocked)
DCN5	DCN1	6.8	Ethernet host
DCN6	DCN1	-1.7	DDCMP host, WWV reference host
DCN7	DCN1	60-Hz	power-grid (locked)
FORD1	GOES	17.9	GOES reference host
WWV	-	-	WWV reference clock
WWVB	-	-	WWVB reference clock

Table 1. Experiment Hosts

In Table 1 the Clock ID column shows the reference host selected as the master clock for each host shown. In this particular configuration host DCN1 was locked to host WWVB, while hosts DCN5, DCN6 and DCN7 were locked to DCN1. Although the offset of GOES can not be directly determined from the Hello messages exchanged between DCnet and FORDnet hosts, the offset of FORD1 relative to GOES was determined by observation to be in the order of a millisecond, so for all practical purposes the offset of FORD1 represents the offset of GOES. In addition, since the WWVB clock was considered by experience the most accurate and reliable and the offset of DCN1 relative to WWVB was negligible, DCN1 was considered the reference clock with offset zero relative to the NBS clocks.

During the setup phase of the experiments the intrinsic drift rates of the crystal oscillators in the four hosts DCN1, DCN5, DCN6 and FORD1 equipped with them was measured as shown in the "Drift" column in Table 1. The two hosts DCN3 and DCN7 are equipped with line-frequency clocks. For experimental purposes DCN3 was unlocked and allowed to free-run at the line-frequency rate, while DCN7 remained locked.

An ASCII file consisting of about 0.2 megabyte of reformatted data, was collected for each Universal-Time (UT) day of observation beginning on 28 June and continuing through 8 July. Each file was processed by a program that produces an eight-color display of measured offsets as a function of time of observation. Since the display technique uses a bit-map display and each observation overwrites the bit-map in an inclusive-OR fashion, the sample dispersion is immediately apparent. Over eight samples per pixel on the time axis are available in a 24-hour collection period. On the other hand, the fine granularity of almost four samples per minute allows zooming the display to focus on interesting short-term fluctuations, such as in the case of the WWV clock.

4.3. Discussion of Results

Each of the four previously mentioned issues of interest will be discussed in following subsections.

4.3.1. On Power-Grid Clocks

Telephone interviews with operators and supervisors of the Potomac Electric Power Company (PEPCO), the electric utility serving the Washington, DC, area, indicate that there are three major operating regions or grids, one east of the Rockies, a second west of the Rockies and a third in parts of Texas. The member electric utilities in each grid operate on a synchronous basis, so that clocks anywhere within the grid should keep identical time. However, in the rare case when a utility drops off the grid, no attempt is made to re-establish correct time upon rejoining the grrd. In the much more common case when areas within the grid are isolated due to local thunderstorms, for example, clock synchronization is also disrupted.

The experiments provided an opportunity to measure with exquisite precision the offset between a clock connected to the eastern grid (DCN3) and the NBS clocks. The results, confirmed by the telephone interviews, show a gradual gain in time of between four and six seconds during the interval from about 1700 local time to 0800 the next morning, followed by a more rapid loss in time between 0800 and 1700. If the time was slewed uniformly throughout these extremes, the rate would be about 100 ppm.

The actual slewing rates depend on the demand, which itself is a function of weather, day of the week and season of the year. Similar effects occur in the western and Texas grids, with more extreme variations in the Texas grid due to the smaller inertia of the system, and less extreme variations in the western grid, due to smaller extremes in temperature, less total industrial demand and a larger fraction of hydro-electric generation.

The uilities consider timekeeping a non-tariffed service provided as a convenience to the customer. In the eastern grid a control station in Ohio manually establishes the baseline system output, which indirectly affects the clock offset and slewing rate. The local time is determined at the control station with respect to a WWVB radio clock. The maximum slewing rate is specified as .025 Hz (about 400 ppm), which is consistent with the maximum rates observed. In the western grid the baseline system output is adjusted automatically using a servomechanism driven by measured offsets from the NBS clocks.

Given the considerations above, it would seem feasable for hosts to synchronize logical clocks to a particular power grid, but only if corrections were transmitted often enough to maintain the required accuracy and these corrections were delivered to the hosts essentially at the same time. Assuming a worst-case 400-ppm slewing rate and one minute between correction broadcasts, for example, it would in principle be possible to achieve accuracies in the 20-ms range. There are a number of prediction and smoothing techniques that could be used to inhance accuracy and reduce the overhead of the broadcasts.

Host DCN3, which uses a line-frequency clock interface, was unlocked during the experiment period so that the offset between the PEPCO clock, which is locked to the eastern power grid, could be measured with respect to the reference host DCN1. Host DCN7, which uses the same interface, remained locked to DCN1. In spite of the previously noted instability of the power grid, DCN7 remained typically within 30 ms of DCN1 and only infrequently exceeded 100 ms in the vicinity of large changes in system load that occured near 0800 and 1700 local time. Over the seven-day period from 2 July through 8 July the mean offset was less than a millisecond with standard deviation about 24 ms, while the maximum was 79 ms and minimum -116 ms.

Experiments were also carried out using ICMP Timestamp messages with hosts known to use line-frequency clock interfaces in California, Norway and Germany. The results indicated that the western power grid is rather more stable than the eastern grid and that the overseas grids are rather less stable. In the Oslo, Munich and Stuttgart areas, for example, the diurnal variation was observed to exceed ten seconds.

4.3.2. On Clocks Synchronized via Network Links

As mentioned previously, all network links used to synchronize the clocks were carrying normal data traffic throughout the experiment period. It would therefore be of interest to investigate how this affects the accuracy of the individual clocks.

Table 2 summarizes the mean and standard deviation of the measured offsets between the WWVB radio clock and various hosts as shown in Figure 2. Measurements were made over the 24-hour period for each of several days during the experimental period. Each entry shown in Table 2 includes the mean of the statistic over these days, together with the maximum variation.

Host	Mean	Deviation	Link Type and Speed
DCN1	.08/.02	0.53/.02	WWVB radio clock (1200 bps)
DCN5	-13.61/.04	1.1/0.4	Ethernet (10 Mbps)
DCN6	0.27/.18	5.8/1.0	DDCMP (4800 bps)
FORD1	38.5/1.6	2.5/0.5	DDCMP (9600 bps)

Table 2. Link Measurements

The departure of the mean shown in Table 2 from zero is related to the drift of the crystal oscillator used in the hardware interface (see Table 1). As described previously, FORD1 was synchonized to the GOES radio clock with neglible offset, so that the mean and standard deviation shown can be accurately interpreted to apply to the GOES radio clock as well.

The results show that the uncertaincies inherent in the synchronization algorithm and protocols is in the same order as that of the reference clocks and is related to the speed of the links connected the reference hosts to the other hosts in the network. Further discussion on the FORD1/GOES statistics can be found in the next section.

Further insight into the error process can be seen in Table 3, which shows the first derivative of delay.

Host	Dev	Max	Min	Error
DCN3	2.3	12	-17	10
DCN5	1.5	45	-45	5
DCN6	9	94	-54	40
DCN7	1.4	6	-7	5
FORD1	3.4	68	-51	15

Table 3. First Derivative of Delay

The mean and standard deviation of delay were computed for all hosts on a typical day during the experimental period. In all cases the magnitude of the mean was less than one. The standard deviation, maximum and minimum for each link is summarized by host in Table 3. A common characteristic of the distribution in most cases was that only a handful of samples approached the maximum or minimum extrema, while the vast majority of samples were much less than this. The "Error" colum in Table 3 indicates the magnitude of the estimated error when these extrema are discarded.

A very interesting feature of the observations was the unexpectedly low standard deviation of DCN3, which was locked to the power grid and thus would be expected to show wide variations. Upon analysis, this turned out to be a natural consequence of the fact that the Hello messages are generated as the result of interrupts based on the line frequency when the local clock had just been incremented by 1/60th of a second.

The synchronizing protocol and implementation were carefully constructed to minimize the loss of accuracy due to sharing of the network links between data and control traffic, as long as sufficient resources (in particular, packet buffers) are available. Since the various network links shown in Figure 2 operate over a wide range of rates, it is possible that undisciplined bursts of traffic can swamp a host or gateway and precipitate a condition of buffer starvation.

While most hosts using paths through the experimental configuration were relatively well-disciplined in their packetization and retransmission policies, some Unix 4.2bsd systems were notorious exceptions. On occasion these hosts were observed sending floods of packets, with only a small amount of data per packet, together with excessive retransmissions. As expected, this caused massive congestion, unpredictable link delays and occasional clock synchronizing errors.

The synchronizing algorithms described above successfully cope with almost all instances of congestion as described, since delay-induced errors tend to be isolated, while inherent anti-spike and smoothing properties of the synchronizing algorithm help to preserve accuracies in any case. Only one case was found during the ten-day experiment period where a host was mistakenly synchronized outside the linear-tracking window due to congestion. Even in this case the host was quickly resynchronized to the correct time when the congestion was cleared.

4.3.3. On the Accuracy of Radio Clocks

One of the more potent motivations for the experiments was to assess the accuracy of the various radio clocks and to determine whether the WWV radio clock was an appropriate replacement for the expensive WWVB or GOES clocks. A secondary consideration, discussed further in the next section, was how the various clocks handled disruptions due to power interruptions, leap seconds and so forth.

4.3.3.1. The Spectracom 8170 WWVB Radio Clock

As the result of several years of experience with the WWVB radio clock, which is manufactured by Spectracom Corporation as Model 8170, it was chosen as the reference for comparison for the GOES and WWV radio clocks. Washington, DC, is near the 100-microvolt/meter countour of the WWVB transmitter at Boulder, CO, well in excess of the 25-microvolt/meter sensitivity of the receiver. The antenna is located in a favorable location on the roof of a four-storey building in an urban area.

Using the data from the instruction manual, the propagation delay for the path from Boulder to Washington is about 8 ms, while the intrinsic receiver delay is about 17 ms. The clock is read via a 1200-bps asynchronous line, which introduces an additional delay of about 7 ms between the on-time transition of the first character and the interrupt at the middle of the first stop bit. Thus, the WWVB radio clock indications should be late by 8 + 17 + 7 = 32 ms relative to NBS standard time. While it is possible to include this delay directly in the clock indication, this was not done in the experiments. In order to account for this, 32 ms should be subtracted from all indications derived from this clock. The uncertaincy in the indication due to all causes is estimated to be a couple of milliseconds.

4.3.3.2. The True Time 468-DC GOES Radio Clock

The GOES radio clock is manufactured by True Time Division of Kinemetrics, Incorporated, as Model 468-DC. It uses the Geosynchronous Orbiting Environmental Satellite (GOES), which includes an NBS-derived clock channel. Early in the experiment period there was some ambiguity as to the exact longitude of the satellite and also whether the antenna was correctly positioned. This was reflected in the rather low quality-of-signal indications and occasional signal loss reported by the clock and also its apparent offset compared with the other radio clocks.

Table 4 shows a summary of offset statistics for the GOES radio clock by day (all day numbers refer to July, 1985).

Day	Mean	Dev	Max	Min
2	31.6	9.4	53	-76
3	19.8	22.1	53	-64
4	42.8	17.1	>150	19
5	39.3	2.2	54	-45
6	37.8	2.7	53	19
7	62.2	13.0	89	22
8	38.2	2.8	90	-7

Table 4. GOES Radio Clock Offsets

On all days except days 5, 6 and 8 long periods of poor-quality signal reception were evident. Since the antenna and satellite configuration are known to be marginal, these conditions are not considered representative of the capabilities of the clock. When the data from these days are discarded, the mean offset is 38.4 ms with standard deviation in the range 2.2 to 2.8. The maximum offset is 90 ms and the minimum is -45 ms; however, only a very small number of samples are this large - most excursions are limited to 10 ms of the mean.

In order to compute the discrepancy between the GOES and WWVB clocks, it is necessary to subtract the WWVB clock delay from the mean offsets computed above. Thus, the GOES clock indications are 38.4 - 32 = 6.4 ms late with respect to the WWVB clock indications. which is probably within the bounds of experiment error.

4.3.3.3. The Heath GC-1000 WWV Radio Clock

The WWV radio clock is manufactured by Heath Company as Model GC-1000. It uses a three-channel scanning WWV/WWVH receiver on 5, 10 and 15 MHz together with a microprocessor-based controller. The receiver is connected to an 80-meter dipole up about 15 meters and located in a quiet suburban location. Signal reception from the Fort Collins transmitters was average to poor during the experiment period due to low sunspot activity together with a moderate level of geomagnetic disturbances, but was best during periods of darkness over the path. The clock locked at one of the frequencies for varying periods up to an hour from two to several times a day.

The propagation delay on the path between Fort Collins and Washington is estimated at about 10 ms and can vary up to a couple of milliseconds over the day and night. While it is possible to include this delay in the clock indications, which are already corrected for the intrinsic receiver delay, this was not done in the experiments. During periods of lock, the clock indications are claimed to be accurate to within 100 ms.

Table 5 shows a summary of offset statistics for the WWV radio clock by day (all day numbers refer to July, 1985).

Day	Mean	Dev	Max	Min
2	-31	36	110	-119
3	-42	38	184	-141
4	-21	38	61	-133
5	-31	37	114	-136
6	-48	42	53	-160
7	-100	80	86	-315
8	-71	70	115	-339

Table 5. WWV Radio Clock Offsets

On inspection of the detailed plots of offsets versus time the data reveal an interesting sawtooth variation with period about 25 cycles per hour and amplitude about 90 ms. Once the clock has locked for some time the variation decreases in frequency and sometimes disappears. This behavior is precisely what would be expected of a phase-locked oscillator and accounts for the rather large standard deviations in Table 5.

On inspection of the plots of offsets versus time, it is apparent that by far the best accuracies are obtained at or in the periods of lock, which is most frequent during periods of darkness over the propagation path, which occured roughly between 0800 UT and 1100 UT during the experiment period. Excluding all data except that collected during this period, the mean offset is -21.3 ms with standard deviation in the range 29-31. The maximum offset is 59 ms and the minimum is -118 ms.

In order to compute the discrepancy between the WWV and WWVB clocks, it is necessary to subtract the total of the propagation delay plus WWVB clock delay from the mean offsets computed above. Thus, the WWV clock indications are -21.3 - 10 - 32 = -72.3 ms late (72.3 ms early) with respect to the WWVB clock indications. Considering the large standard deviations noted above, it is probably not worthwhile to include this correction in the WWV clock indications.

On exceptional occasions excursions in offset over 300 ms relative to the WWVB clock were observed. Close inspection of the data showed that this was due to an extended period (a day or more) in which lock was not achieved on any frequency. The master oscillator uses a 3.6-MHz crystal oscillator trimmed by a digital/analog converter and register which is loaded by the microprocessor. The occasional excursions in offset were apparently due to incorrect register values as the result of noisy reception conditions and excessive intervals between lock. On occasion the oscillator frequency was observed in error over 4 ppm due to this cause, which could result in a cumulative error of almost 400 ms per day if uncorrected.

4.3.4. On Handling Disruptions

The experiment period was intentionally selected to coincide with the insertion of a leap second in the worldwide time broadcasts. The intent was to examine the resulting behavior of the various radio clocks and the synchronization algorithm when an additional second was introduced at 2400 UT on 30 June.

As it turned out, radio reception conditions at the time of insertion were quite poor on all WWV frequencies, the WWVB frequency and the GOES frequency. Thus, all three clocks took varying periods up to several hours to resynchonize and correct the indicated time. In fact, the only time signals heard around the time of interest were those from Canadian radio CHU, but the time code of the Canadian broadcasts is incompatible with the of the US broadcasts.

As mentioned above, the WWVB clock was used as the master during the experiment period. About two hours after insertion of the leap second the clock resynchronized and all hosts in the experimental network were corrected shortly afterwards. Since the magnitude of the correction exceeded 128 ms, the correction was of a step nature, but was not performed simultaneously in all hosts due to the individual timing of the Hello messages. Thus, if timing-critical network operations happened to take place during the correction process, inconsistent timestamps could result.

The lesson drawn from this experience is quite clear. Accurate time synchronization requires by its very nature long integration times, so that epochal events which disrupt the process must be predicted in advance and applied in all hosts independently. In principle, this would not be hard to do and could even be integrated into the operation of the step-correction procedure described earlier, perhaps in the form of bits included in Hello messages which trigger a one-second correction at the next rollover from 2400 to 0000 hours.

In order for such an out-of-band correction to be effective, advance notice of the leap second must be available. At present, this information is not available in the broadcast format and must be obtained via the news media. In fact, there are spare bits in the broadcast format that could be adapted for this purpose, but this would require reprogramming both the transmitting and receiving equipment. Nevertheless, this feature should be considered for future systems.

4.4. Additional Experiments

A set of experiments was performed using two WIDEBAND/EISN gateways equipped with WWVB radio clocks and connected to the ARPANET. These experiments were designed to determine the limits of accuracy when comparing these clocks via ARPANET paths. One of the gateways (ISI-MCON-GW) is located at the Information Sciences Institute near Los Angeles, while the other (LL-GW) is located at Lincoln Laboratories near Boston. Both gateways consist of PDP11/44 computers running the EPOS operating system and clock-interface boards with oscillators phase-locked to the WWVB clock.

The clock indications of the WIDEBAND/EISN gateways were compared with the DCNet WWVB reference clock using ICMP Timestamp messages [6], which record the individual timestamps with a precision of a millisecond. This technique is not as accurate as the one described in Section 3, since the protocol implementation involves the user-process level, which can be subject to minor delays due to process scheduling and interprocess-message queueing. However, calibration measurements made over several of the links shown in Figure 2 indicate that the measurement errors are dominated by the individual link variations and not by the characteristics of the measurement technique itself.

Measurements were made separately with each gateway by sending an ICMP Timestamp Request message from the ARPANET address of DCN1 to the ARPANET address of the gateway and computing the round-trip delay and clock offset from the ICMP Timestamp Reply message. This process was continued for 1000 message exchanges, which took about seven minutes. Table 6 shows the statistics obtained with ISI-MCON-GW and Table 7 those with LL-GW (all numbers are milliseconds).

ISI-MCON-GW	Mean	Dev	Max	Min
Offset	-16	40	126	-908
Delay	347	59	902	264

Table 6. ISI-MCON-GW Clock Statistics

LL-GW (a)	Mean	Dev	Max	Min
Offset	-23	15	32	-143
Delay	310	25	536	252

Table 7. LL-GW Clock Statistics

The smaller values of standard deviation and extreme for LL-GW are probably due to the shorter ARPANET path involved. The confidence in the mean offset can be estimated by dividing the standard deviation by the square root of the number of samples (1000), which suggests that the mean offsets are accurate to within a couple of miliseconds. The mean offsets of the WIDEBAND/EISN clocks as a group relative to the DCN1 clock may thus indicate a minor discrepancy in the setting of the delay-compensation switches.

It is well known that ARPANET paths exhibit wide variations in delays, with occasional delays reaching surprising values up to many seconds. In order to improve the estimates a few samples were removed from both the offset and delay data, including all those with magnitude greater than one second.

The above experiments involve a burst of activity over a relatively short time during which the ratio of the measurement traffic to other network traffic may be nontrivial. Another experiment with LL-GW was designed with intervals of ten seconds between ICMP messages and operated over a period of about three hours. The results are shown in Table 8.

LL-GW (b)	Mean	Dev	Max	Min
Offset	-16	93	990	-874
Delay	371	108	977	240

Table 8. LL-GW Clock Statistics

Note that the standard deviations and extrema are higher than in the previous experiments, but the mean offset is about the same.

The results of these experiments suggest that time synchronization via ARPANET paths can yield accuracies to the order of a few milliseconds, but only if relatively large numbers of samples are available. The number of samples can be reduced and the accuracy improved by using the techniques of Section 3 modified for ICMP Timestamp messages and the longer, more noisy paths involved.

5. Summary and Conclusions

The experiments described above were designed to verify the correct operation of the DCnet time-synchronization algorithms and protocols under a variety of scenarios, including the use of line-frequency clocks, three types of radio clocks and various types of interprocessor links. They involved the collection and processing of many megabytes of data collected over a ten-day period that included the insertion of a leap second in the standard NBS time scale. Among the lessons learned were the following:

1. The algorithms and protocols operate as designed, yielding accuracies throughout the experimental net in the order of a few milliseconds to a few tens of milliseconds, depending on the topology and link type.

2. Glitches due to congestion, rebooted hosts and link failures are acceptably low, even in the face of massive congestion resulting from inappropriate host implementations elsewhere in the Internet.

3. A synchronization scenario where the clocks in all hosts are locked to the line frequency and corrections are broadcast from a central time standard will work only if all hosts are on the same power grid, which is unlikely in the present Internet configuration, but may be appropriate for some applications.

4. In spite of the eastern power grid wandering over as much as six seconds in a day, it is possible to achieve accuracies in the 30-ms range using line-frequency interface clocks and corrections broadcast on the local net.

5. Radio clocks can vary widely in accuracy depending on signal reception conditions. Absolute time can be determined to within a couple of milliseconds using WWVB and GOES radio clocks, but only if they are calibrated using an independent

standard such as a portable clock. The inexpensive WWV clocks perform surprisingly well most of the time, but can be in error up to a significant fraction of a second under some conditions.

6. Adjustments in the time scale due to leap seconds must be anticipated before they occur. The synchronization protocol must include a mechanism to broadcast an adjustment in advance of its occurance, so that it can be incorporated in each host simultaneously. There is a need to incorporate advance notice of leap seconds in the broadcast time code.

7. Time synchronization via ARPANET paths can yield accuracies in the order of a few milliseconds, but only if relatively large numbers of samples are available. Further work is needed to develop efficient protocols capable of similar accuracies but using smaller numbers of samples.

6. References

1. Lindsay, W.C., and A.V. Kantak. Network Synchronization of Random Signals. IEEE Trans. Comm. COM-28, 8 (August 1980), 1260-1266.

2. Mills, D.L. Time Synchronization in DCNET Hosts. DARPA Internet Project Report IEN-173, COMSAT Laboratories, February 1981.

3. Mills, D.L. DCNET Internet Clock Service. DARPA Network Working Group Report RFC-778, COMSAT Laboratories, April 1981.

4. Mills, D.L. Internet Delay Experiments. DARPA Network Working Group Report RFC-889, M/A-COM Linkabit, December 1983.

5. Mills, D.L. DCN Local-Network Protocols. DARPA Network Working Group Report RFC-891, M/A-COM Linkabit, December 1983.

6. Postel, J. Internet Control Message Protocol. DARPA Network Working Group Report RFC-792, USC Information Sciences Institute, September 1981.

7. Postel, J. Time Protocol. DARPA Network Working Group Report RFC-868, USC Information Sciences Institute, May 1983.

8. Postel, J. Daytime Protocol. DARPA Network Working Group Report RFC-867, USC Information Sciences Institute, May 1983.

9. Su, Z. A Specification of the Internet Protocol (IP) Timestamp Option. DARPA Network Working Group Report RFC-781. SRI International, May 1981.

10. Marzullo, K., and S. Owicki. Maintaining the Time in a Distributed System. ACM Operating Systems Review 19, 3 (July 1985), 44-54.

11. Mills, D.L. Algorithms for Synchronizing Network Clocks. DARPA Network Working Group Report RFC-956, M/A-COM Linkabit, September 1985.

12. Mills, D.L. Network Time Protocol (NTP). DARPA Network Working Group Report RFC-958, M/A-COM Linkabit, September 1985.