The NMIA WebTimer: A Traceable Time Service For Checking Stopwatches and Timers

Michael J. Wouters, E. Louis Marais and Robert J. Williams
Standards for Time and Frequency
National Measurement Institute Australia
Lindfield, NSW 2070, Australia
Michael.Wouters@measurement.gov.au

***Abstract*—We have developed the WebTimer, an Internet service, accessible through a browser, for checking the operation of stopwatches and timers. We explain its design, operation and performance. A detailed uncertainty budget has been developed to assist the user. The WebTimer has been available as a paid subscription service since July 2020.**

Keywords—stopwatch; timer; calibration

I. INTRODUCTION

In Australia, the National Association of Testing Authorities (NATA) provides accreditation for test and calibration laboratories to ISO/IEC17025. NATA recommends that stopwatches and timers used in laboratories be checked against a credible time source every six months. Until recently, the recommended reference was the national speaking clock service, which was traceable to Co-ordinated Universal Time Australia (UTC(AUS)) maintained at the National Measurement Institute Australia (NMIA). The speaking clock service was shut down on 30 September 2019 due to the dismantling of the legacy copper telephone network, leaving calibration laboratories without a suitable source of time. Although we offered our traceable Network Time Protocol (NTP) service as a replacement, we thought that a more convenient and easily-accessible method would be helpful to users. The Internet-based WebTimer was our solution.

We considered two ways of implementing the WebTimer: as a conventional application running on the user's computer, communicating with a dedicated, custom time server; or a browser-based application communicating with a web server. At least one other national metrology institute (SP Sweden, now RISE) has run an Internet service for calibrating stopwatches in the past. Implementation of the service as a conventional application has the advantage of ease of securing the time server but has the usual problems of supporting software installed and running on a user's computer and across multiple platforms. Conversely, a browser application is simpler to support but securing the server is more difficult. In particular, for best performance and transparent traceability we wanted the server to be synchronized directly to UTC(AUS), precluding its operation as a managed service with a hosting provider. We ultimately decided on a browser-based solution, which was also the best match to our software development expertise.

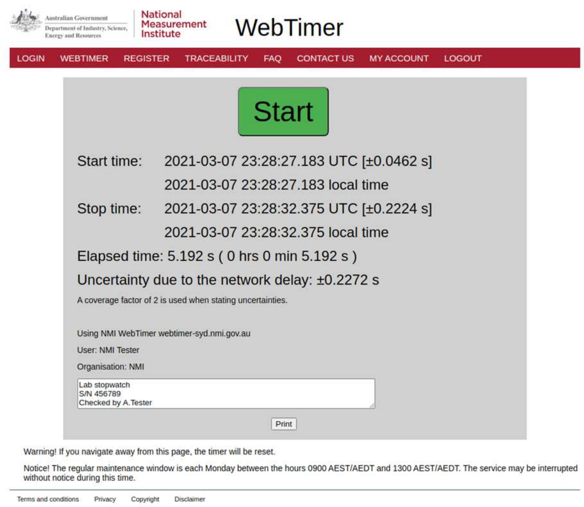

Fig.1 Interface of the WebTimer, showing the results of a completed measurement.

One issue we considered prior to making the service publicly available was whether measurements performed with it could be deemed a calibration by **us** and a NMIA calibration certificate could be issued. However, since the **user** is performing the measurement, it is clear that it is only they who can issue any calibration certificate. The WebTimer is simply the traceable reference standard for the measurement. Where the user operates under a formal accreditation system, they must be accredited for the measurement. In this case, there is also the complication that the reference standard is not maintained within their quality system. We therefore describe the WebTimer as only being suitable for checking instruments, to discourage the interpretation of its output as a calibration. We further require that users agree not to use the WebTimer for providing calibration services to external clients.

Here we explain how the WebTimer works, present data that demonstrate the accuracy achievable with it and develop a complete uncertainty budget. We also describe some of our experiences in running the service.

978-1-6654-3935-0/21/$31.00 ©2021 Crown

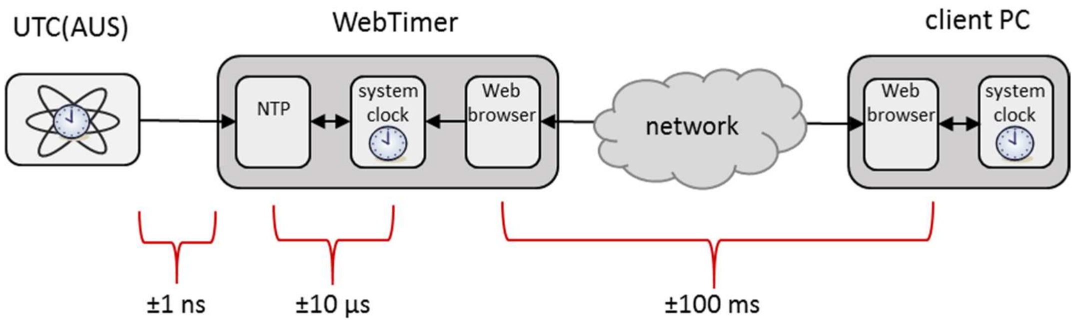

Fig. 2 Measurement chain for the WebTimer and some typical uncertainties.

II. Design and Operation

The WebTimer application is designed to be used like a stopwatch (Fig.1). When a user presses the 'Start' button, the user's computer requests the time from the remote WebTimer server. The button then changes to read 'Stop'. When the 'Stop' button is pressed, a second time request is made. The difference between these two times is then calculated and reported to the user. The uncertainty in each measurement is mainly determined by the time taken for each time request to travel to the server and back (round trip time, RTT) and this is also reported to the user. A full report can be generated by the user to document the measurement.

In the current implementation of the WebTimer the round trip time is measured by the user's computer. There are guards in the software to protect against bad measurements such as those that would be caused by the PC time stepping backwards during a measurement.

Traceability of the WebTimer server time is established as follows. The WebTimer server takes its timing reference directly from the Australian primary standard UTC(AUS) as a one pulse-per-second (1 pps) electrical signal (Fig. 2). This keeps the server synchronized with respect to UTC(AUS) to better than 10 µs.

Time-of-day is obtained from NMI's network of NTP servers which are themselves fully traceable to UTC(AUS). One of the servers is on the same LAN as the WebTimer.

Continuous records of the synchronization of the WebTimer and its offset from UTC(AUS) are kept and are available to users of the service.

III. Uncertainty Analysis

A. *WebTimer Server Time*

The principal uncertainty in the accuracy of the 1 pps signal provided to the WebTimer server is the uncertainty in the delay of the cable providing this 1 pps. Other factors such as the effect of temperature variations are negligible in comparison.

The server's operating system must time stamp each incoming 1 pps to measure the difference between the server time and the 1 pps, and the server does not respond instantly to each incoming 1 pps. The typical delay is measured to be between 10 µs and 20 µs and is accounted for as a systematic correction of 15 ± 5 µs. It also takes a measurable time to make a call to the server's operating system to retrieve the current time. An upper bound on this systematic delay is measured to be between 17 and 25 ns and is accounted for as a correction of 21 ± 4 ns.

Finally, a contribution for the instability of the server time is added. The log files of time offsets applied to the system time by ntpd, the NTP software maintaining system time on the WebTimer server, are used to estimate the time deviation (TDEV). TDEV at an averaging time of 16 s, the interval between updates to the server time by ntpd, is used to characterize the instability and is approximately 0.6 µs.

These uncertainties are summarized in Table I below. The uncertainty is dominated by the contribution from the 1 pps time stamping delay.

TABLE I. Uncertainty Budget for the Server Time

Component	Raw uncertainty (µs)	Reducing factor	Standard uncertainty (µs)
Cable delay	0.001	1	0.001
1 pps time stamping delay	5	1	5
Server time stamp latency	0.004	1	0.004
Server time stability	0.6	1	0.6
	Combined uncertainty		5
	Expanded uncertainty		10

B. *Uncertainty of the time stamp received by the client*

The WebTimer client requests a time stamp via HTTP from the WebTimer server over the Internet. The time taken for the request to be transmitted to the server and the reply to be returned to the client (network delay) is unpredictable and the major source of measurement uncertainty.

To estimate the delay and its uncertainty, the WebTimer uses the same method as is used in the Network Time Protocol to synchronize computers over a network. Four time stamps are recorded and utilised by the client:

T_1 = *client* time at which the client sends a request,
T_2 = *server* time when the server receives the request,
T_3 = *server* time when the server replies to the request, and
T_4 = *client* time when the client receives the server's reply.

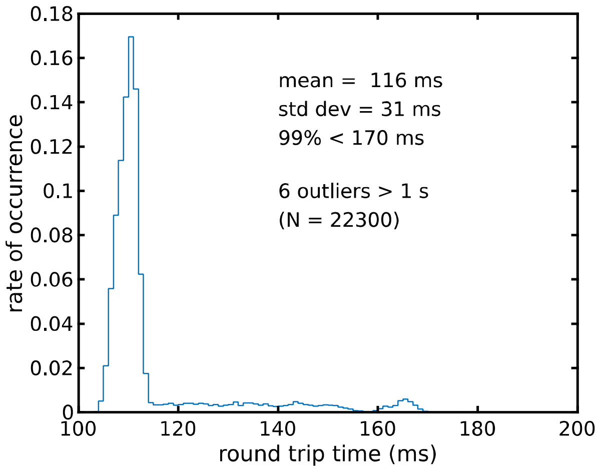

Fig. 5. Round trip time (network delay) of time stamp queries.

The time stamp T_1 is the time stamp associated with pressing the 'Start/Stop' button. It is made using the client computer as the time reference and is not traceable. However, it can be corrected to the (traceable) server time as follows:

T_1(corrected) = T_2 – forward network delay.

The round trip network delay Δ is estimated as:

$\Delta = (T_4 - T_1) - (T_3 - T_2)$.

The term $T_3 - T_2$ represents the processing delay on the server and is usually much less than 1 ms.

To estimate the one-way network delay, δ, it is necessary to assume that the forward and return network delays are equal, so that $\delta=\Delta/2$. Under normal conditions this is a good approximation but it cannot be assumed to be always true, so δ is assigned an uncertainty of $\Delta/2$.

The use of a mouse to trigger recording of time stamps introduces a time delay and uncertainty. USB human interface devices like mice and keyboards are polled devices so there is a delay between, for example, the mouse being clicked and the event being handled by the operating system. The polling interval is typically 8 ms, so a systematic delay of 4 ms with an uncertainty of ±4 ms is assigned.

The time stamps have associated resolution errors. T_1 and T_4 are obtained using a Javascript function which has a resolution of 1 ms. T_2 and T_3 are obtained from the server and have a resolution of 1 µs.

The delay δ is not strictly traceable because it is estimated using the client PC time. The uncertainty due to errors in the PC crystal clock frequency can be readily bounded though. At worst, this crystal will have a fractional frequency error of a few parts in 10^4. Assuming that δ = 1 s and a fractional frequency error of 1 part in 10^3, the additional error in δ is just 1 ms. δ is typically much less than 1 s, so this estimate is conservative.

The use of NTP to synchronize the client PC time introduces some unpredictability in δ because of the varying behaviour of NTP clients. For example, a Simple Network Time Protocol (SNTP) client may just step the system time to update it, and this step could be of the order of a few seconds. A large positive step results in a large RTT and this can be rejected by the user according to their uncertainty requirements. Negative RTTs are rejected by the software but a negative step, comparable with the RTT, cannot be reliably detected in software. An experienced user may however detect that the RTT is unusually low. For this reason, it is recommended that the client PC not be NTP synchronized.

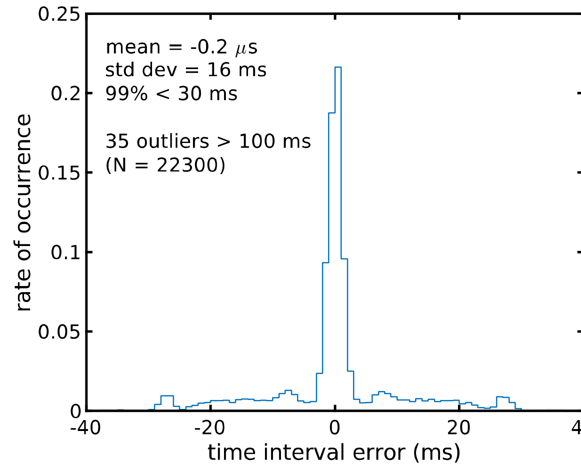

Fig. 3. Difference between the nominal (10 s) and measured time interval.

The fixed components of the uncertainty budget for a single time stamp, including that of the server time as estimated in Table I, are summarized in Table II. Some components appear twice, and have been added in quadrature. The main contribution to the uncertainty is the mouse click latency.

The uncertainty presented to the user in the WebTimer does not include the components described here because some of them may have to be adjusted according to the user's circumstances. Detailed guidance is given in a document available on the website [1]. In practice, the network delay will be the largest source of uncertainty apart from the user's estimate of reaction times.

TABLE II. UNCERTAINTY BUDGET FOR A RECEIVED TIME STAMP

Component	Raw uncertainty (ms)	Reducing factor	Standard uncertainty (ms)
WebTimer server time	0.005	1	0.01
Client PC time stamp resolution	1.4	1.73	0.8
Server time stamp resolution	0.0014	1.73	0.0008
Client PC frequency error	1	1	1
Mouse click latency	4	1.73	2.3
	Combined uncertainty		2.6
	Expanded uncertainty		5.2

IV. PERFORMANCE

To characterize the WebTimer, we used an automated test wherein mouse clicks were generated directly from a 0.1 Hz digital signal driving a modified mouse. Various combinations of operating system (Linux, Windows XP/7/10) and browser (Chrome, Chromium, Edge, Internet Explorer) were tested.

Fig. 3 and Fig. 4 show some typical data collected over three days using the Chrome browser on Windows 10. The network connection was a consumer grade ADSL line. For the data presented here, the error due to the network delay is less than 60 ms, much less than the uncertainty a user might assign to their reaction time. Data obtained for other combinations of operating system and browser were very similar.

Whilst some measurements may have to be rejected because of a long network delay, this should be a rare event. Before registering and paying, users have the option of trying the WebTimer to see whether their network connection is satisfactory. Anecdotal evidence indicates that the WebTimer remains useable over many network hops. A user in New Zealand reports a network delay between 100 and 200 ms.

The speaking clock service did not provide a measurement of the network delay, however the one way delay on a voice call has to be less than 200 ms for acceptable quality [2]. The WebTimer network delay matches or betters this, and thus is a satisfactory replacement for the speaking clock in this respect.

One effect noted during testing was that the first click of a mouse had extra latency associated with its processing for some combinations of browser and operating system. Clicking immediately after the first click resulted in a measurably shorter apparent RTT (for example, 20 ms vs. 70 ms). The 'Stop' click, being typically made some minutes after the 'Start' click, thus also showed this extra latency.

V. OPERATING THE SERVICE

The WebTimer is operated as a paid service, following Australian government guidelines for recovering the costs of providing services. A yearly subscription costs 55 Australian dollars.

The WebTimer has a very simple interface and there have been very few questions from users about its operation. The most common of these has been how to use it to check multiple stopwatches simultaneously, which can be achieved by opening multiple instances of the WebTimer using browser tabs or windows. Most interaction with users has occurred with respect to organizing payment for the service, which is primarily through an online shopfront with credit card payment facility and automatic subscription activation. Alternative payment methods are available but require manual processing steps which delay activation and add substantial labor cost relative to the subscription fee.

VI. CONCLUSIONS

The WebTimer was conceived as a tool for checking the operation of stopwatches and timers, motivated by the shutdown of the national speaking clock service. The aim was to achieve an accuracy at least as good as the speaking clock for the majority of users, and this target was met. The availability of an uncertainty estimate for the measurement and the utility of a report on the measurement for its documentation make it a superior replacement for the speaking clock.

We believe there are also intangible benefits to us in operating the service. First, it directs traffic to the NMI website since the payment system is hosted there. This showcases NMI to laboratories that wouldn't normally deal with us and makes them aware of other services that we offer such as training, and chemical and biological reference materials. Users have commented on this. Second, it increases the visibility and impact of NMI's time and frequency standards and services: we have 30 or so customers for our traditional calibration services whereas the WebTimer currently has about 150 (and growing). The WebTimer is available to try out at:

https://webtimer-syd.nmi.gov.au/trytimer.php

REFERENCES

[1] https://webtimer-syd.nmi.gov.au/docs/Traceability of the NMI WebTimer.pdf

[2] http://itu.int/rec/T-REC-G.114