

Tracking Users' Clicks and Submits: Tradeoffs between User Experience and Data Loss

Ron Kohavi, David Messner,
Seth Eliot, Juan Lavista Ferres, Randy Henne,
Vignesh Kannappan, and Justin Wang
Microsoft Experimentation Platform

Updated 9/28/2010

Abstract

Tracking users' online clicks and form submits (e.g., searches) is critical for web analytics, controlled experiments, and business intelligence. Most sites use web beacons to track user actions, but waiting for the beacon to return on clicks and submits slows the next action (e.g., showing search results or the destination page). One possibility is to use a short timeout and common wisdom is that the more time given to the tracking mechanism (suspending the user action), the lower the data loss. Research from Amazon, Google, and Microsoft showed that small delays of a few hundreds of milliseconds have dramatic negative impact on revenue and user experience (Kohavi, et al., 2009 p. 173), yet we found that many websites allow long delays in order to collect click. For example, until March 2010, multiple Microsoft sites waited for click beacons to return with a 2-second timeout, introducing a delay of about 400msec on user clicks. To the best of our knowledge, this is the first published empirical study of the subject under a controlled environment. While we confirm the common wisdom about the tradeoff in general, a surprising result is that the tradeoff does not exist for the most common browser family, Microsoft Internet Explorer (IE), where no delay suffices. This finding has significant implications for tracking users since no waits is required to prevent data loss for IE browsers and it could significantly improve revenue and user experience. The recommendations here have been implemented by the MSN US home page and Hotmail.

1. Introduction

Web sites track visitors' clicks on links and form submits (e.g., search) in order to assess user activities and improve the web site. Such tracking is necessary for web analytics, ranging from optimizing short-term content like news and sport headlines to assessing the impact of changes to the page, such as moving modules or redesigning the site when running controlled experiments (Box, et al., 2005; Mason, et al., 1989; Kohavi, et al., 2009). The typical tracking mechanism utilizes JavaScript to capture the click or form-submit. The event is suspended¹ as a request is made to a logging server to record the user's action before the action is taken. The crux of the problem is that the request to the logging server takes time and introduces a delay that negatively impacts the user experience, as the user's request is being suspended. There are several examples where delays of 100msec to 500msec had dramatic impact on revenues at Amazon and Google (Kohavi, et al., 2009). Waiting for the logging request to complete can take a long time and hurt user experience while proceeding before the logging server acknowledges the request can cause data about clicks to be lost (e.g., retries will not happen or the request may not even leave the client's browser if the duration is small).

The analysis done in this paper shows that the click-tracking and form-tracking mechanisms used at MSN, which waited for beacons like the Omniture tracking beacon, waited about 400msec. About 4% to 8% of the time, the beacons timed out at the full 2000msec limit. In an experiment run by the Bing team, a delay of 2000msec on Bing reduced queries by 2.5% and ad clicks by 4.4%; at Amazon, a delay of 100-250msec decreased revenues by 1% (Kohavi, et al., 2009 p. 173).

In this paper, we design a controlled experiment to study the tradeoffs between the delay and data loss See Kohavi et al. (2009) for Survey of controlled experiments. To the best of our knowledge, this is the first empirical online controlled experiment to understand the data loss for different tracking mechanisms with different threshold values. Microsoft

¹ For form submits the request is cancelled, the logging mechanism is fired, and the request is then resubmitted with a slight modification to avoid the infinite loop that arises otherwise.

uses a variety of tracking mechanisms and the set of experiments described in this paper is an important step at understanding the tradeoffs that exist between different tracking methods.

Based on the experiment results, we recommend that form tracking should not wait at all with the IE family of browsers since there was no change in loss when an IE browser waited longer. This is a very surprising fact, which runs counter to all our initial expectations. In our initial plans we did not even test this condition, but earlier runs showed little difference for several variants down to 50msec, therefore a 0msec treatment was added. As with many controlled experiments, the results are humbling and lead to changes that cannot be foreseen. We recommend a wait time of 150msec with all other browsers. For non-IE browsers, there is a tradeoff between waiting longer and data loss. The 150msec presents a reasonable tradeoff. The server could either generate appropriate JavaScript based on the browser (User Agent), or the JavaScript client code could branch on the browser family. Our recommendations were adopted by Hotmail and the MSN home page. One word of caution: our evaluation focused on form tracking, mostly search, but we believe the results should generalize.

This paper shares the insights from the controlled experiment where six tracking variants were tested with different timeout values and is organized as the follows. Following the experiment goals and details in Section 2, we review experiment results in Section 3. Section 4 details the beacon latencies and time outs and Section 5 concludes the paper.

2. Experiment goals and details:

2.1 Goals of the experiment

Two types of tracking mechanisms (Fixed-time and Out-of-band) with various threshold values and three tracking systems (MSN Tracking System, ExP, and Omniture) were implemented in this experiment. The principal goal to understand the tradeoff between data loss and user experience as described in the introduction. We are also interested in understanding if some of the three tracking systems faster than others, if browsers behavior differently and if users from different geographical regions make the tracking systems working differently.

2.2 Tracking mechanisms

There are three types of tracking mechanisms:

- a. **Fixed-time** (implemented as a spin-lock). Given a parameter t in msec, the beacon requests are initiated in parallel, and the browser spins for t msec before continuing to the destination (independent of whether the tracking requests came back).
- b. **Out-of-band** (OOB). Given a parameter t in msec, the beacon requests are initiated in parallel, and the browser waits for all of them to come back (maximum time for all beacons) or until time t elapsed (timeout).
- c. **Mousedown**. Tracking mousedown events. Firing a beacon early may have benefits. Mousedown mechanism introduces other complexities because it is incomparable to Fixed-time and Out-of-band, which only log clicks and form submits, whereas mouse down may log right-clicks and events that are later cancelled. Therefore, it was not tested in the experiment.

The six tracking mechanisms tested in this experiment are:

1. OOB-2000: OOB tracking with $t=2000$ msec timeout.
2. OOB-150: OOB tracking with $t=150$ msec timeout
3. Fixed-500: Fixed-time with $t=500$ msec
4. Fixed-150: Fixed-time with $t=150$ msec
5. Fixed-50: Fixed-time with $t=50$ msec
6. Fixed-0: Fixed-time with $t=0$ msec

In addition, a special treatment was setup that is equivalent to OOB-2000 (t=2000), but which reports the times the beacons took in order to answer some of the secondary goals. After the beacons returned (or timed out), a 2nd beacon was sent to ExP with the timings. The second beacon was OOB-500, i.e., it waited up to 500msec before continuing (see Section 4.2 for analysis).

2.3 Tracking systems

Three tracking systems exist today and all are used by the MSN home page:

1. **MSN Tracking System.** A Microsoft internal system. The data from this system provides near-real-time click-through statistics for reporting activities, such as editors tuning headlines.
2. **Omniure:** A tracking and web analytics system offered by Omniure/Adobe Systems.
3. **ExP.** Microsoft's Experimentation Platform (<http://exp-platform.com>) is another internal system used at Microsoft for conducting controlled experiments.

In the above tracking systems, parallel calls are made to the systems. In fixed-time treatments, the overhead is negligible. When a user clicks or submits a form, JavaScript handler calls the three tracking systems parallel, waits for time T (spinlock) then submits form or navigate to link despite the beacon calls from the three tracking systems return of not. In the case of Out-of-band tracking, the wait is for all three systems to return or until the timeout is reached to submit form or navigate to link.

2.4 design of the experiment

From the MSN US home page, 20% of total users were split into 5 equal groups of 19% each for the OOB and fixed treatments, except Fixed-0, which was assigned 3%². The special treatment was given 2% because it adds an additional delay of up to 500msec when reporting the timing results.

In an experiment like this, it is critical to run a *controlled experiment*, where all variants run in parallel. Any attempt to assess deltas in a non-controlled environment suffers from external factors, such as world events in the news, time of day, day of week, etc. As an example, Figure 1 is a graph of the click-through rate (the ratio of the number of clicks to the number of page views) on the MSN US home page over several days. With over many millions of points being aggregated (every user for every hour), each hour below is an average of many thousands of users, yet the variability is enormous and ranges from about 30% to over 50% during the day. In controlled experiments, one can see that the two treatments track each other very closely, allowing evaluation of the delta.

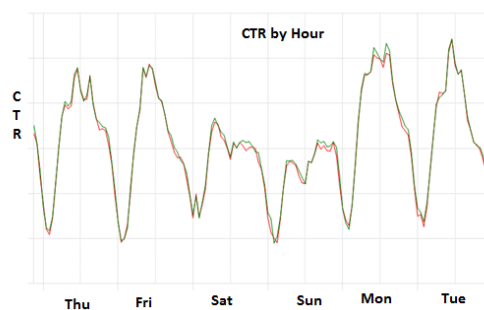
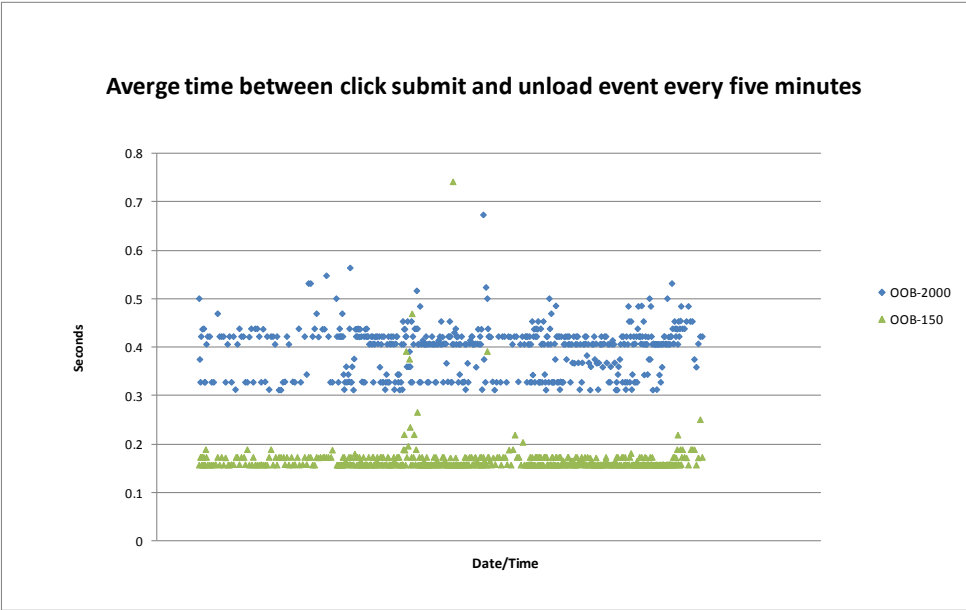


Figure 1: CTR by Hour for MSN HP

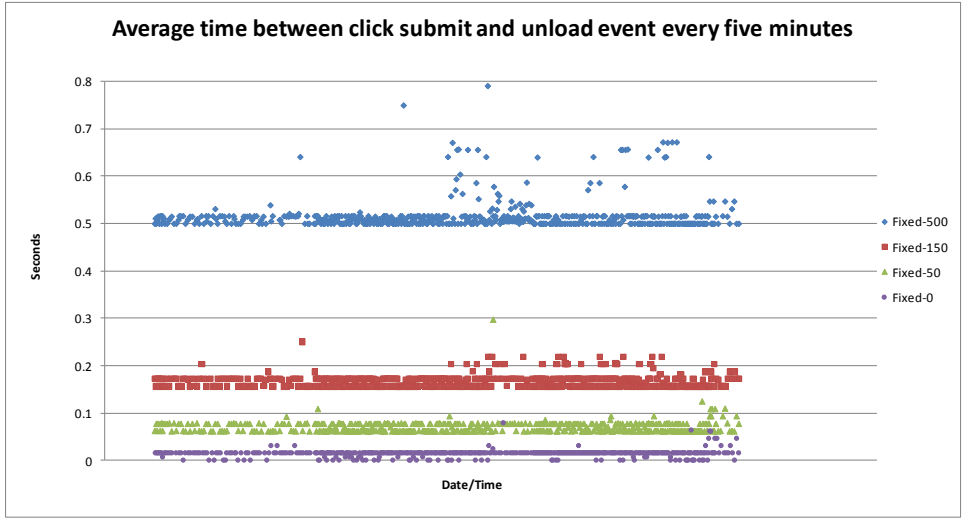
During the experiment period, Gomez, a 3rd party website performance monitoring tool, was asked to test if they can detect the click delay. They setup an agent that monitored the different treatments from Boston. The timing was

² As in many cases, results of experiments are humbling. The allocation to treatments shows that we didn't think fixed-0 would be useful in practice, but rather that it would give us an "upper bound." Initially it was not even in the plans, but we added it later and assigned it a lower percentage because we thought it would hurt click tracking significantly. It turns out that this is the best option for the IE family.

measured from the submit event to the “beforeUnload” event about every 5 minutes. For the first chart, showing OOB (out-of-band) treatments, we can see that OOB-150 times out practically in all cases when the browser agent is out of Boston, MA (both the MSN Tracking System and ExP were in Tukwila, WA). OOB-2000 shows that most requests finish around 350-450 msec. This is a significant delay to the user experience. It is interesting to note that there are no obvious hour-of-day effects



The second chart, below, shows the fixed times. The charts match the expected durations fairly precisely.



3 Experiment Results:

The best way to assess the data loss of different variants is to get data from the destination. For this experiment, we used Bing Search. When looking at the destination page requests, a common assumption is that any page request with a referrer of the MSN home page was a click from the MSN home page to that destination. This assumption is only approximately correct. There are cases where users actually didn't click on the MSN home page and the destination log implies they did. Some example include: page refreshes and users reopening tabs using features of IE7 and IE8. There are also known cases where clicks are not recorded on the MSN Home page. Examples include: refreshes, user with JavaScript off, robots that do not execute JavaScript, tracking beacon filtering by browsers, and right-clicks. We describe the problem and our data cleansing process below.

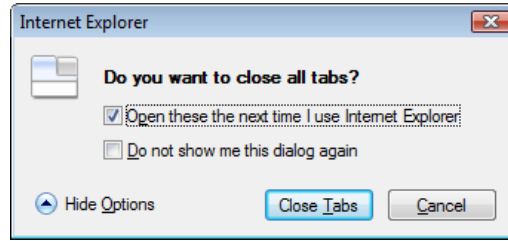
1. Refreshes. The browser will refresh the page by sending a page request with the original referrer, leading to a false click. While this isn't common for search, it is very common in some links like the MSN Money destination where users will refresh the stock quotes to get updated prices.
2. Users with JavaScript off. A small (but non-negligible) population of users don't have JavaScript on. Therefore, the tracking code on the MSN home page never fires. They do trigger a page view on the destination because Search logs server-side and MSN money logs with ExP, which uses an image beacon for page views. This is a non-recorded true click.
3. Many robots (bots, monitoring agents, automated systems, etc) don't run JavaScript. They behavior similar to JavaScript-less users mentioned above. A small number of robots may significantly impact clickthrough-rates if they accept cookies and fall into the same treatment (otherwise, they get randomized, diluting their impact).
4. Tracking beacons. Some browsers (or plugins that block ads) ignore small 1x1 images. The following table shows users who had at least two clicks in ExP, but no page views. Opera rarely records tracking beacons. For most browsers, this is a small percentage.

Browser	MajorVersion	Clicks	No PV's	PCT No PV's
Opera	9	20871	18840	90.3%
Opera	10	418	359	85.9%
MSIE	5	26944	91	0.3%
FireFox	1	22024	62	0.3%
MSIE	8	868531	2105	0.2%
FireFox	2	215230	464	0.2%
MSIE	7	18583681	33385	0.2%
FireFox	3	2085113	3742	0.2%
Chrome	1	73935	118	0.2%
MSIE	6	7498355	9597	0.1%
Safari	3	332386	180	0.1%
Safari	4	11954	4	0.0%
FireFox	0	435	0	0.0%
Chrome	0	347	0	0.0%
Safari Mobile	3	509	0	0.0%
Chrome	2	2155	0	0.0%

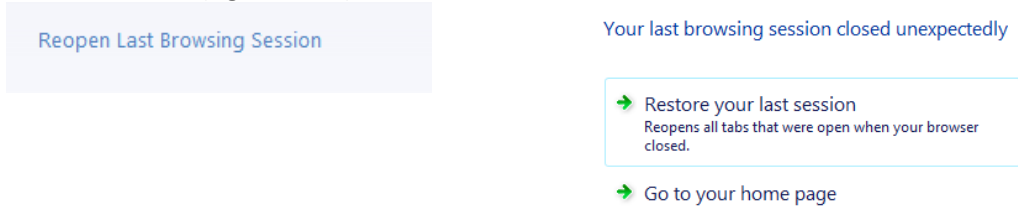
The Omniture tracking beacons on the MSN home page are now 2x2 images, probably for this reason.

5. Right-clicks. With modern browsers, it is becoming more common for users to right-click and open in new tab/window or control/shift-click. These actions are not currently tracked by the MSN home page. While these are less relevant for Search and Stock quotes form submits, they are important for link tracking.

6. Users electing to reopen the browser with the same tabs. In IE7, when closing tabs, you can show options and select “Open these the next time I use Internet Explorer” as shown below.

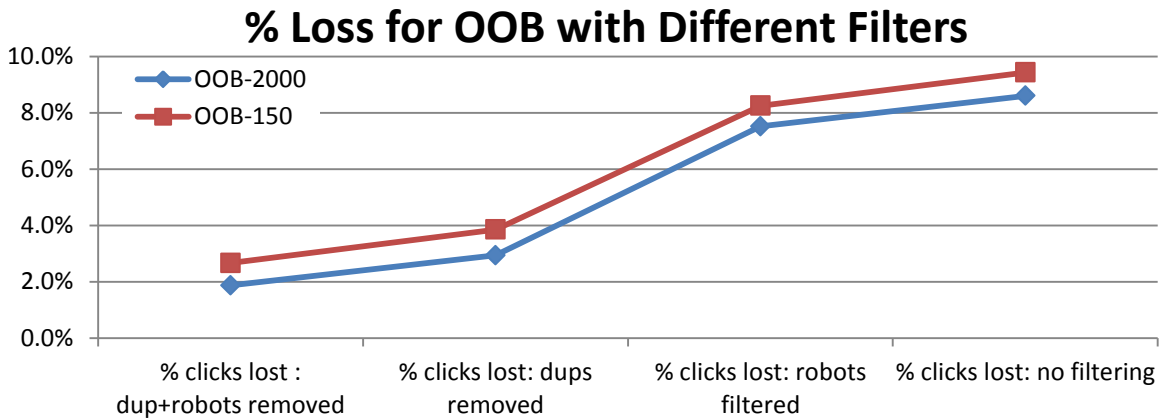


In IE8, when opening a tab, there’s an option to reopen last browsing session (left below) and after a crash you can restore sessions (right below)

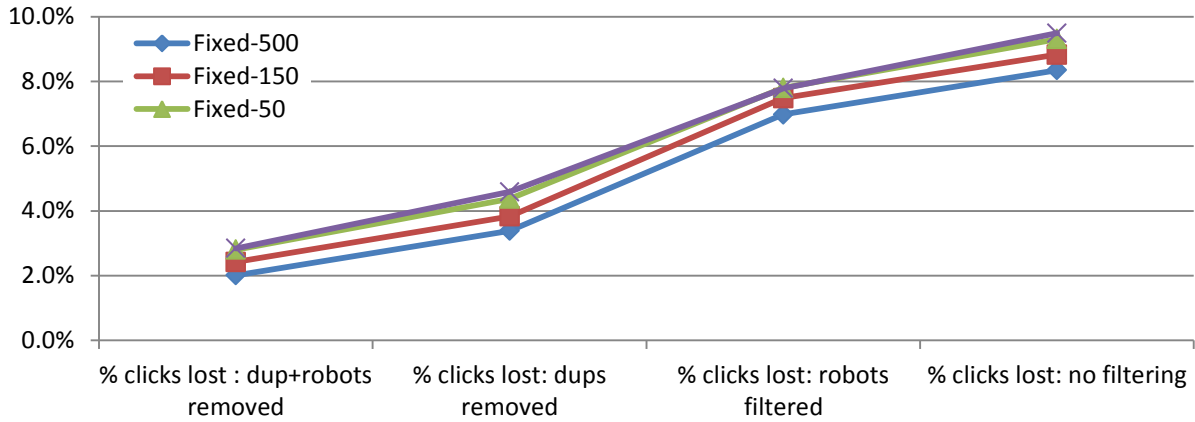


In both browsers, a shutdown (e.g., when installing patches) will automatically select this option so that after a reboot, the browser will open to the same state.

To remove effects of robots we employ several heuristics. The following graphs show the sensitivity of the results in this report relative to duplicate and robot removal. Basically, the ordering doesn’t change, which implies that the results are not sensitive, a desired property.



% Loss for Fixed Timers with Different Filters



After data cleansing, Figure 2 shows the loss for the different variants and makes it clear that that the common wisdom of: wait less, you lose more is real. When OOB-2000 is reduced to OOB-150, the loss grows 26% from 2.3% to 2.9%. When Fixed-500 is reduced to fixed-150 to fixed-50, the loss grows from 2.1% to 2.5% to 3.0%, a 43% increase.

Data Loss

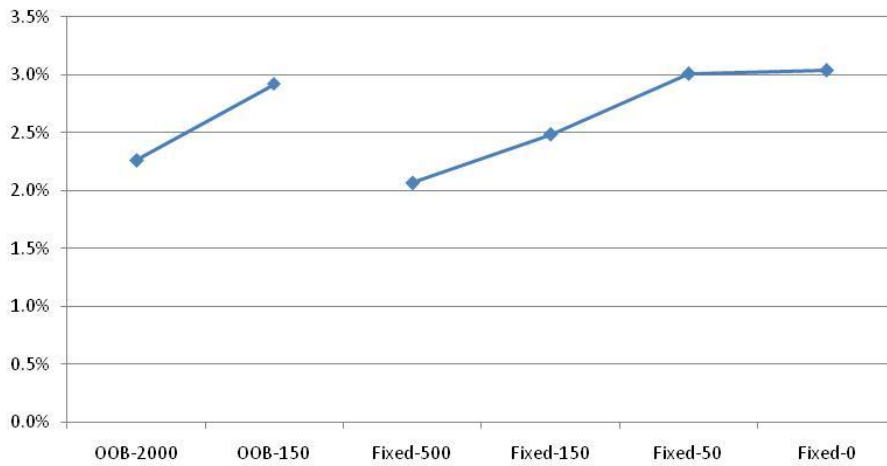


Figure 2: % clicks lost for the variants

Breaking down the percent of clicks loss by browser type, we found that all browsers, except the IE family, behave as expected: the smaller the wait times, the more data is lost (Figure 3). Except for Safari, the IE family, and Others, OOB-150 behaves fairly similarly to Fixed-time 150. Since the timeout is the same, the expectation is that the data loss will be similar although OOB-150 would provide a better user experience sometimes, if the beacon comes back faster. For the fixed timers, IE8 did significantly worse than IE6 and 7. This is due to a known issue in IE8 that was fixed before IE8’s final release. There was no change in loss when an IE browser waited longer (Figure 4), therefore search tracking should not wait at all with the IE family of browsers. For other browsers we recommend a wait time of 150msec. For non-IE

browsers, there is a tradeoff between waiting longer and data loss. The 150msec presents a reasonable tradeoff.

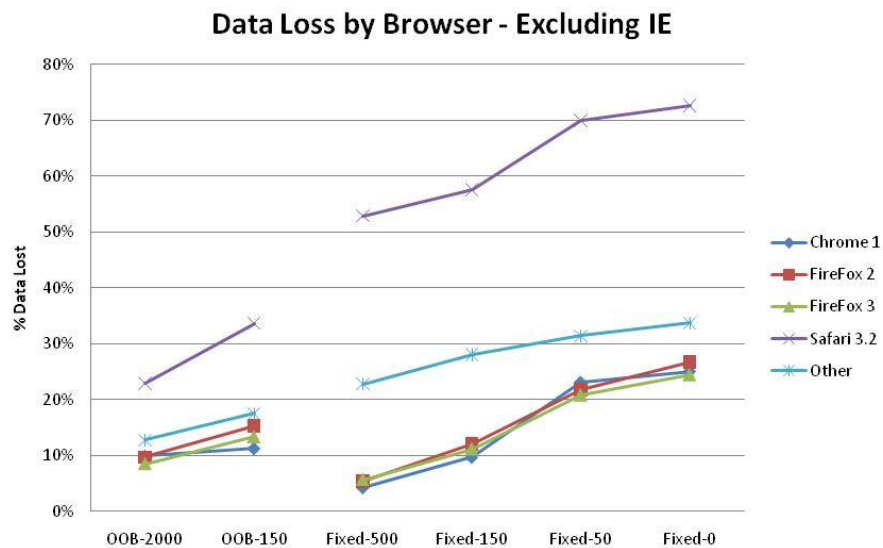


Figure 3: % clicks lost by browser excluding IE family

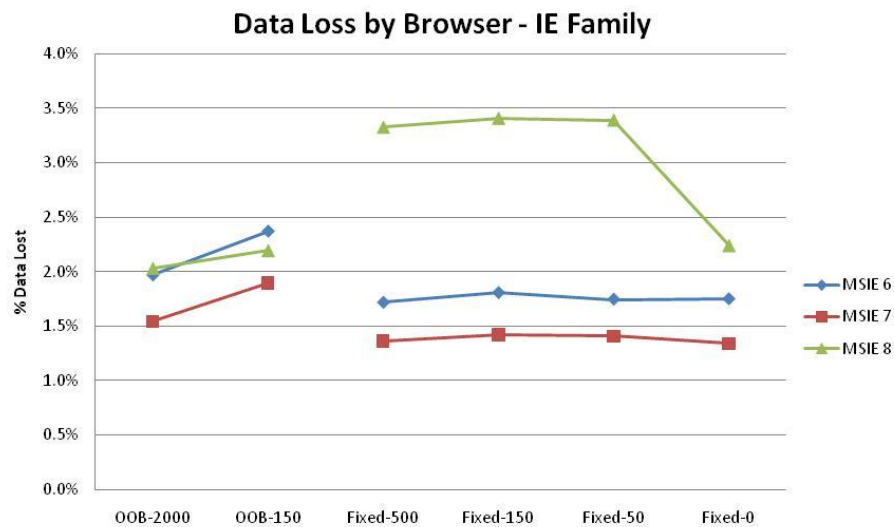
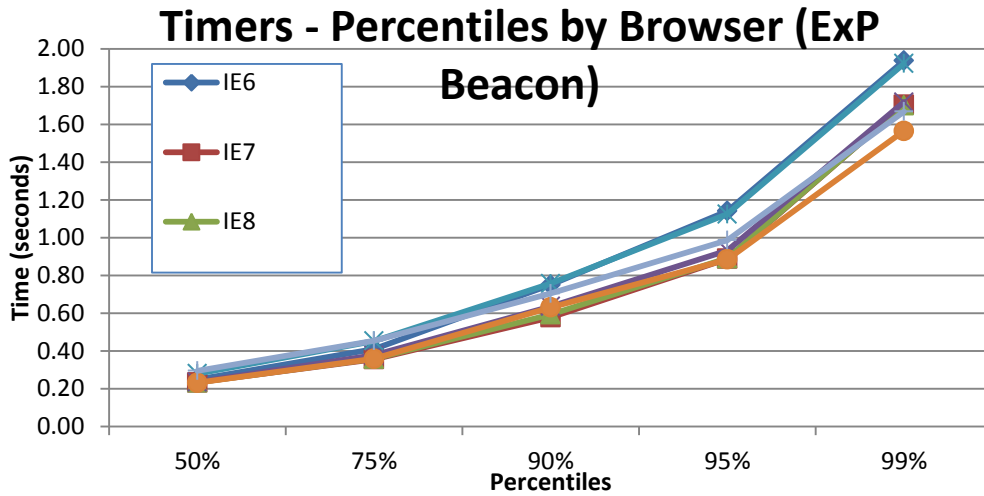


Figure 4: % clicks lost by IE browsers

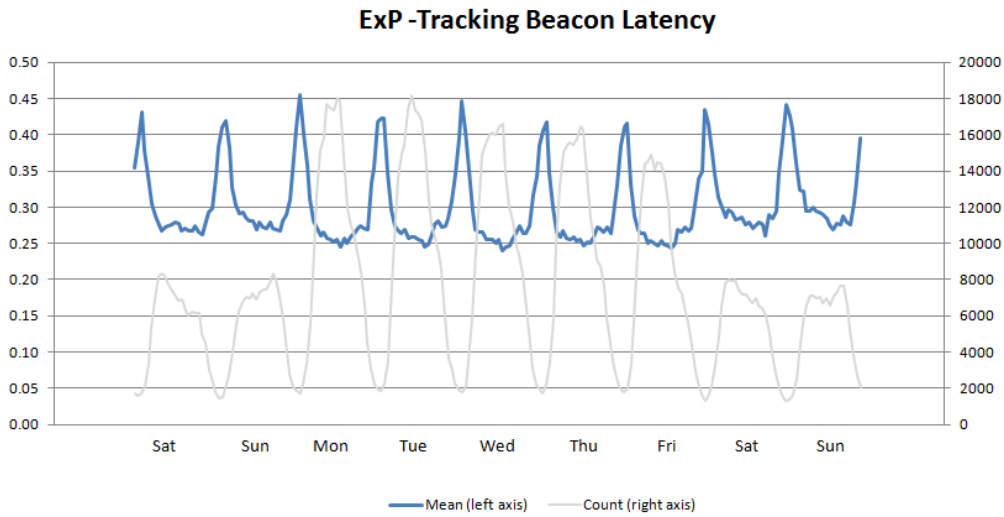
4 Actual Tracking Times and Time Outs

4.1 Actual times for ExP tracking system

Treatments with fixed times should take about the time designated in the fixed-time parameter. For Out-Of-Band treatments, we found for 50% of users the time for the beacons is less than 200msec, but for 1% of users, waiting for the beacon entails times of 1800msec (the wait is the max of the beacons). The graph below shows the timing for different browsers

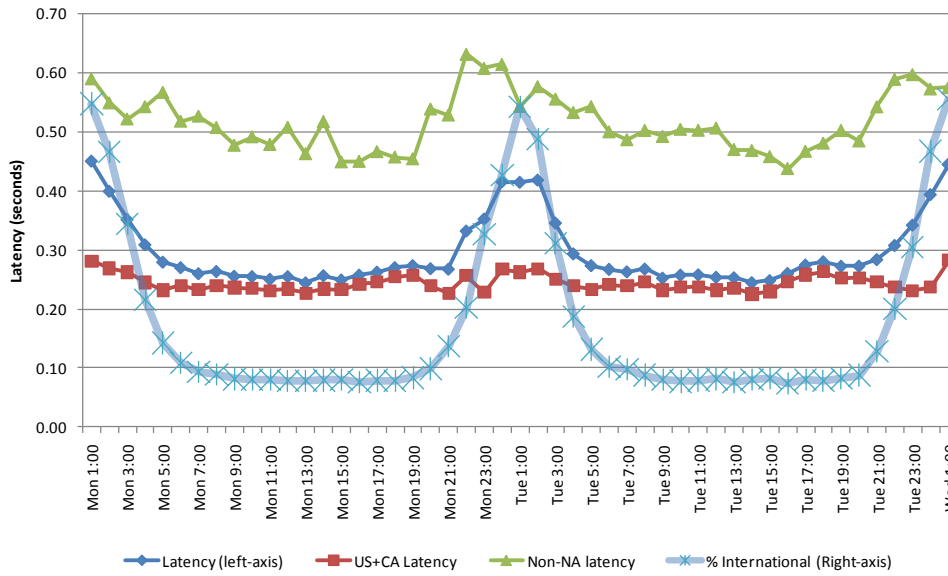


It's important to remember that the timing reflects both the browser efficiency and also the network bandwidth and geography. For example, it may be that IE6 users are laggards more likely to use slower/older machines and perhaps slower network connections. Conversely, Chrome users may be early innovators and hackers with fast network connections. The graph below shows latency throughout the day, i.e., how long the client had to wait for the Exp Beacon, where timeouts result in a delay of 2 seconds:



The beacon latency in blue shows that mean latencies are around 280 to 450 msec with some peak reaching 500msec. The grey line shows the number of beacons sent during each hour for a sample. We can see the daily traffic patterns and the differences between weekends at the beginning of the graph and every week thereafter, and the weekday patterns, which are more volatile and reach higher peaks. Zooming into a day, the lowest latency is actually during peak times and vice-versa: the longest latencies are during 10PM to 4AM, a non-intuitive fact. There are several hypotheses that can explain this, but the strongest is that this is due an increase in latencies and timeouts from far-away countries. The following graph shows the latencies from IPs in the North America (US and Canada) vs. non-NA based on reverse-IP lookup. The red line represents North America latencies, which are small (about 250msec) and very stable throughout the day. The green line shows non-NA traffic with latencies that are much higher (450-600msec) but also relatively stable. What changes the aggregate latency in blue is the ratio of the two, represented by the thick transparent blue line.

Exp Beacon Latency: North America vs. Non-NA



For non-IE browsers, there is probably a bias in recording: fewer clicks may be recorded for traffic that is “far” and has longer latencies and more timeouts. Looking at user’s countries based on ip-to-geo, the 15 slowest countries with more than 500 clicks in the two days are as follows. Note the significant percentage of 2-second timeouts.

Country	Mean	% Timeouts
Tanzania	1.64	43%
Kenya	1.58	37%
Nigeria	1.48	38%
Iraq	1.40	34%
Iran	1.23	27%
Lebanon	1.17	21%
Syria	1.15	27%
Ghana	1.11	17%

Country	Mean	% Timeouts
Ethiopia	1.05	17%
Sudan	1.03	14%
SURINAME	1.03	18%
BANGLADESH	0.97	16%
BRUNEI	0.95	13%
Kuwait	0.95	13%
Oman	0.88	13%

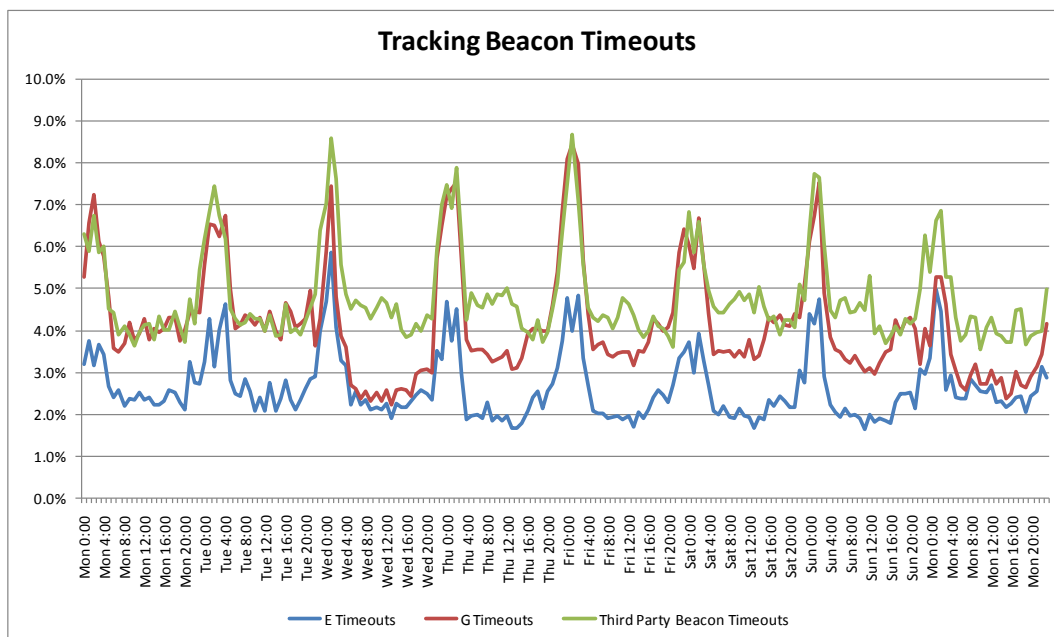
They’re shown on the map below relative to Tukwila, Washington where the data center was at the time of this study.



Note that speed-of-light alone accounts for a small amount: half the earth's circumference is 20,000 KM and speed of light is 300,000 KM/second, which translates into 66msec. Common US coast-to-coast delays, one quarter of 20,000KM, are about 70-90msec because of backbone traversals, indirect paths, and other routing delays, which is about 5 times the speed-of-light, so actual network delays to these countries under reasonable "us quality" conditions should be around 350msec.

4.2 Time outs

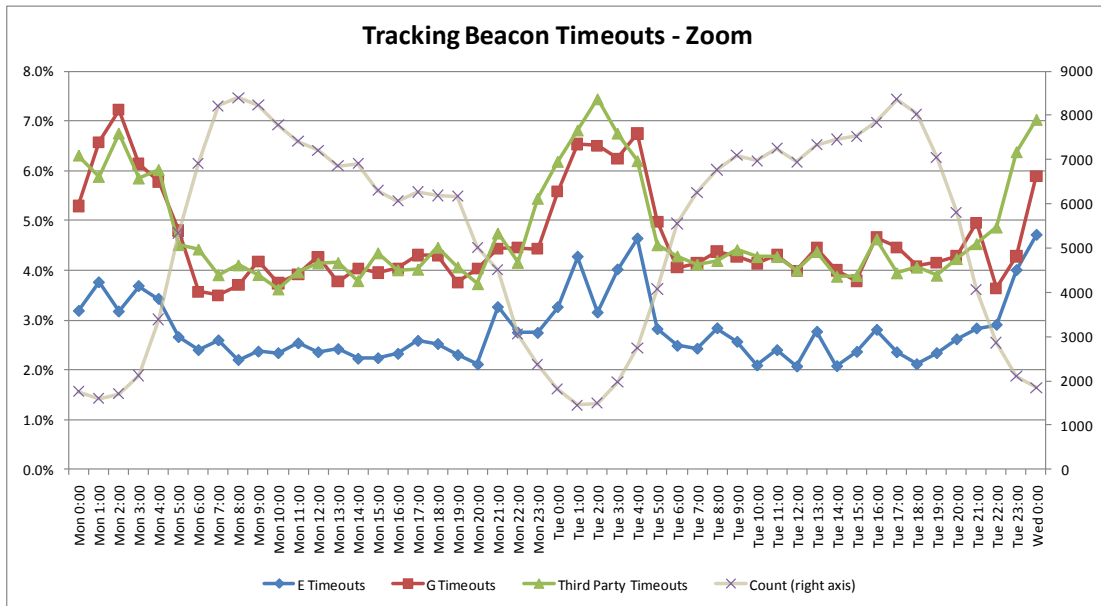
This section reports the percent of requests time out at the maximum allowed 2000msec. Since the special treatment called all systems with a 2-second timeout and reported the times (with a 500msec timeout for that), we are able to provide the percentage of requests that time out. The following graph shows the timeouts for the three tracking systems



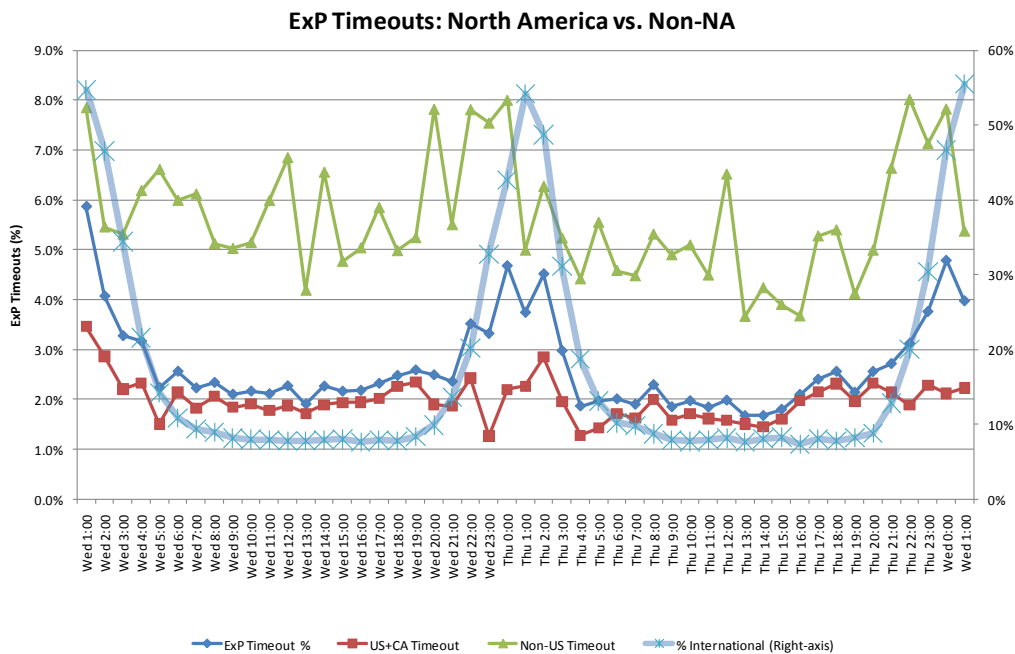
The results are likely to be a slight underestimate because if there is a timeout in the initial request, there is some increased probability that the 2nd reporting call to ExP will fail. However, this is a 2nd order effect for the MSN Tracking

System and Omniture. It may be a larger effect for ExP, which has a larger bias against itself (if the 1st request fails with a 2-second timeout, the probability that the 2nd request will not register with the same system increases more).

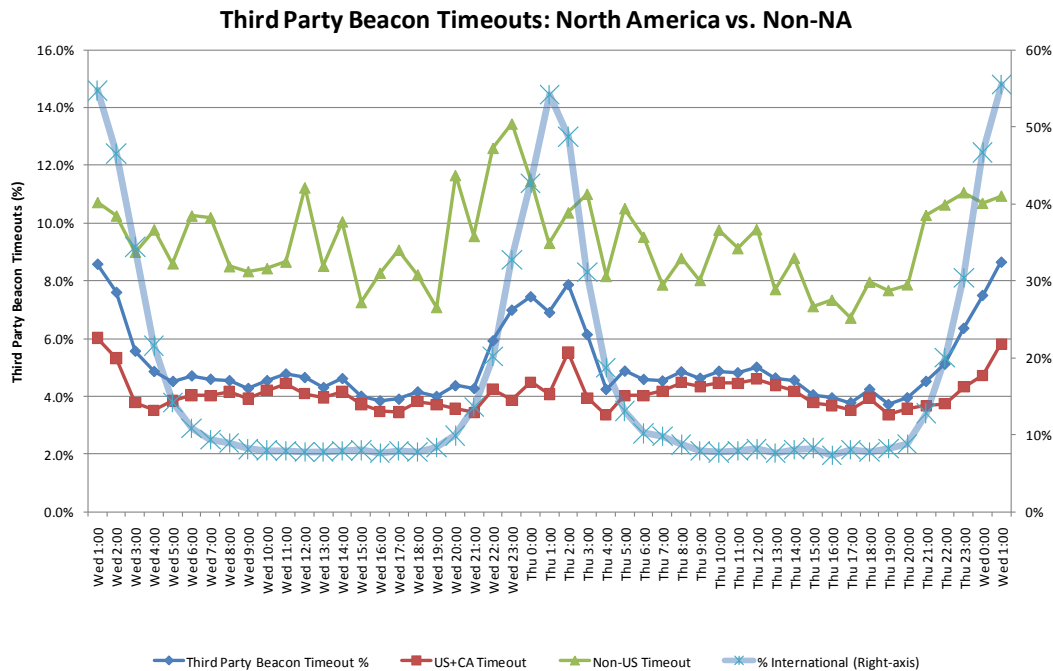
As with the latencies, the timeouts occur at the higher frequencies during the night, and as with latencies, we'll show that these are due to international traffic. However, even during the day, the level is about 4-5% for Omniture and 2-4% for the MSN Tracking System, as shown below.



As with latencies, the following graph shows the percentage of ExP timeouts by geography. In North America, the percent of timeouts hovers around 2%; outside NA, it is about 5-6%.



Omniure is similar as shown below, except that the North America timeout rate is about 4% (double that of ExP) and the non-NA rate is about 8-10% (also significantly higher than ExP), but this is again the time to remind the reader that ExP may have an inherent advantage in terms of the bias.



5 Conclusion:

Based on the experiment, tracking should not wait at all with the IE family of browsers. There was no change in loss when an IE browser waited longer, a surprising fact, which ran counter to all expectations. As with many experiments, the results are humbling and lead to changes that cannot be foreseen. The evaluation focused on form tracking, mostly search. We believe the results should generalize for link tracking.

For other browsers we recommend a wait time of 150msec. For non-IE browsers, there is a tradeoff between waiting longer and data loss. The 150msec presents a reasonable tradeoff.

There were more than 20 million users included in this experiment. To detect a 0.5% change, you need 100 times more users than to detect 5% change; small sites typically aim for large improvements, and so they might need 200K users to detect a 5% delta, but sites like the MSN home page, which are better optimized, have huge monetary gain from 0.5% changes and therefore need a large number of users in experiments to get the sensitivity levels down.

While the experiment focused on understanding the loss with Search as a destination, the tracking experiment ran on all form submits and clicks on the page. Data showed that the whole-page clicks per user increased 0.4%, which translates to significant annualized revenue for MSN HP, which has implemented the recommendations here.

Acknowledgement

This paper is the result of 3-way collaboration between the MSN Home Page team, Microsoft Experimentation Platform team, and Bing search team. Multiple people were involved, but we especially want to thank Ron Logan, Bruk Belachew, Izi Behar, and Tyler Landas from the MSN home page, Zijian Zheng, Max (Wei) Zhang and Eric Schurman from Bing Search, and Brian Frasca and Tamir Melamed from Experimentation Platform. Thanks also to Jim Pierson, Bruce Dawson, Anuj Nigam who helped with Gomez, the IE team including Sarah Bowers, Zeke Odins-Luca, Venkat Kudallur, Adrian Bateman, Christian Stockwell, and others.

Bibliography

Box, George E.P., Hunter, J Stuart and Hunter, William G. 2005. *Statistics for Experimenters: Design, Innovation, and Discovery*. 2nd. s.l. : John Wiley & Sons, Inc, 2005. 0471718130.

Kohavi, Ron. 2007. Practical Guide to Controlled Experiments on the Web: Listen to Your Customers not to the HiPPO. 2007. <http://exp-platform.com/Documents/2007-10EmetricsExperimentation.pdf>.

Kohavi, Ron, Crook, Thomas and Longbotham, Roger. 2009. Online Experimentation at Microsoft. *Third Workshop on Data Mining Case Studies and Practice Prize*. 2009. <http://exp-platform.com/expMicrosoft.aspx>.

Kohavi, Ron, et al. 2009. Controlled experiments on the web: survey and practical guide. *Data Mining and Knowledge Discovery*. February 2009, Vol. 18, 1, pp. 140-181. http://exp-platform.com/hippo_long.aspx.

Mason, Robert L, Gunst, Richard F and Hess, James L. 1989. *Statistical Design and Analysis of Experiments With Applications to Engineering and Science*. s.l. : John Wiley & Sons, 1989. 047185364X .