Webmunk: A New Tool for Studying Online Behavior and Digital Platforms

Chiara Farronato, Andrey Fradkin and Chris Karr*

May 23, 2024

Abstract

Understanding the behavior of users online is important for researchers, policymakers, and companies. But measuring behavior online and conducting experiments is difficult for independent researchers, who do not have access to the user bases or software of technology companies. We introduce Webmunk, an open-source tool designed to make conducting online studies much easier. The user facing side of Webmunk is a browser extension that can track consumer browsing behavior and experimentally modify consumers experiences as they browse the Internet. It can be installed just like any other browser extension. Through this extension, researchers can collect a host of consumer data, from URLs to web page HTML elements, clicks, and scroll positions. The extension can also modify information and change the look of a web page, allowing for researchers to implement interventions that vary across study participants. A key advantage of this approach is that interventions occur while participants are engaging in real world activities such as shopping, browsing the news, using social media, or searching for information. We demonstrate the power of Webmunk by discussing two studies in progress.

The internet has become central to modern economic and social activities. As such, understanding online behavior is critical for policymakers and businesses alike. Researchers have

*Farronato: Harvard University, CEPR, NBER, cfarronato@hbs.edu. Fradkin: Boston University, fradkin@bu.edu, Karr: Audacious Software, chris@audacious-software.com. We thank Tesary Lin and Alex MacKay for productive collaborations using Webmunk and for comments on the white paper. We thank Jean-Michael Lambert and Wenhan Wang for additional software development. We thank James Dana, Yutao Chen, Yunjie Song, Richard Xu, and Hannah Zhang for outstanding research assistance. We thank Stefan Bucher for comments on the draft. We thank the Digital Business Institute at Boston University and the Internet Society Foundation for helping to fund the project.

been increasingly interested in exploring consumer behavior online, yet, the tools available to them for conducting experimental studies remain limited. In this paper, we introduce Webmunk, an open-source tool that allows researchers to manipulate a consumer's online experience and track its effects on the consumer's behavior.[1] Webmunk is designed to be a general purpose tool for researchers to extend and customize. We describe two applications of the technology to illustrate Webmunk's capabilities.

Researchers of digital behavior have started to use custom software to conduct studies. Under this paradigm, recruited participants install software on a device such as a laptop or mobile phone, and this software tracks behavior and implements interventions in the course of typical online behavior. This approach has recently been used to study social media (Allcott, Gentzkow and Song (2022), Aridor (2022), Levy (2021), Beknazar-Yuzbashev et al. (2022)). Although these studies require a similar technology, there is no software infrastructure that allows researchers to easily run such studies. Developing this type of technology is costly–both financially and time-wise–, which inevitably creates entry barriers for academics to develop online research studies with credible random variation.

To overcome the challenges imposed by data availability and software development, we have developed and open-sourced Webmunk, a web browser extension and data collection platform to conduct observational and experimental studies on the Internet. Webmunk consists of two components: a web browser extension and cloud server infrastructure. The browser extension component is designed for the Chrome and Edge browsers. The extension can modify the content of web pages in real time and collect data as users browse the Internet. By keeping track of (anonymized) user identities, the extension can implement different web page manipulations across users or over time, which enables researchers to assign different experimental interventions that they may want to compare. The extension uses various techniques to engage study participants in specific tasks beyond their regular browsing activity, such as completing surveys or visiting designated websites.

The cloud server component of Webmunk encompasses two main elements. First, an enrollment server handles the generation of random user IDs and the allocation of experimental treatments and configurations across users and over time. Second, a data transmission server serves as the central hub for collecting both passive and experimental data, which can be stored and analyzed at a later stage. Both servers use the Passive Data Kit (PDK), a comprehensive data collection framework.

All components of Webmunk are open-source and designed to be easily extended. Additional functionality can be added to Webmunk through the use of modules. Modules are

[1] Webmunk is available at `www.webmunk.org` and `https://github.com/Webmunk-Project`

self-contained and re-usable components that work with Webmunk. In our projects, we have designed modules for identifying products on Amazon, for retrieving users' Amazon order histories, and for pushing pop-ups for the selection of cookie tracking preferences. Modules are particularly useful for implementing manipulation and tracking that is specific to a web domain or that broadly applies to many domains.

This paper, along with relevant companion documentation available at `webmunk.org` and `https://github.com/Webmunk-Project`, aims to provide researchers and software developers with the necessary information to use Webmunk for conducting new research studies. While most researchers may need the support of a software developer to deploy Webmunk, leveraging its existing infrastructure can reduce time and other costs associated with creating software from scratch.

We showcase the functionalities of Webmunk using two of our ongoing projects. The first project (Farronato, Fradkin and MacKay 2023, *Amazon project* henceforth) explores the impact of vertical integration on Amazon. Specifically, we investigate the role of the platform in displaying its own products and those of third-party sellers to customers. Understanding this dynamic is crucial when a digital platform competes directly with third-party sellers who depend on it to reach their customers. The second project (Farronato, Fradkin and Lin 2023, *Cookie project* henceforth) is motivated by the fact that websites often obfuscate cookie preference policies and consent forms in order to nudge users to share their data more than they would otherwise do, a practice known as "dark patterns." We explore the effects of dark patterns on consumers' privacy choices and the consequences of these choices on the types and quantities of ads consumers receive. Importantly, unlike the Amazon project, the cookie project is cross-site.

Our work contributes to the broader Open Science movement (Nosek, Spies and Motyl (2012), Spellman, Gilbert and Corker (2017)) which calls for researchers to make research data and software widely available. In economics, development of open-source software is becoming part of a researcher's professional activity. For example, Conlon and Gortmaker (2020) develop PyBLP, an open-source package to estimate demand models, and Shen et al. (2021) develop a novel framework to improve the efficiency of layout detection and extract text from historical documents.

In addition to software-based studies, there are other paradigms for studying digital behavior. One paradigm is to use data collected by third-parties or scraped by researchers.[2] For

[2] Although the legality of scraping has faced some recent challenges (`https://techcrunch.com/2022/04/18/web-scraping-legal-court/`).

example, Santos, Hortaçsu and Wildenbeest (2012) use data from ComScore[3] to study search behavior online, Calder-Wang (2021) uses Airbnb data scraped by AirDNA[4] to study the effects of Airbnb entry on the rental market, and Lewis (2011) uses scraped data from eBay motors to study asymmetric information and disclosure. The limit of this approach is that third-party or scraped data may measure a limited set of variables and may not contain experimental variation of research relevance.

Another approach to studying digital behavior is to use confidential data directly from the company that generates it. This approach has two main advantages. First, it is less susceptible to measurement error and selection issues (e.g., Farronato and Fradkin 2022). Second, it allows researchers to use randomized controlled trials conducted by the company in order to identify causal effects (e.g., Blake, Nosko and Tadelis 2015). Despite these advantages, most researchers do not have the option of using confidential data. Companies are hesitant to share these data for a variety of reasons, including the perception of regulatory, legal, or competitive risk. When companies do collaborate with researchers, researchers may be limited to studying topics that may benefit the company, or, at the very least, will not hurt it. This creates a selection problem for the type of questions that researchers can hope to answer through company collaborations. A key advantage of our software based paradigm is that it only requires permission from the researchers' Internal Review Boards (IRB) and from study participants.

The rest of the paper proceeds as follows. Section 1 illustrates Webmunk's functionalities, and Section 2 describes the ways in which Webmunk is used for the Amazon and Cookie projects. Section 3 discusses the technical skills required to use Webmunk for new studies. Section 4 presents Webmunk's technical aspects and its modular structure. Section 5 concludes.

1 Webmunk's Functionalities

In this section, we provide a comprehensive overview of Webmunk's functionalities. Our aim is to offer researchers who are considering deploying Webmunk a clear understanding of the possibilities it enables. By delving into the details, we aim to showcase the full range of capabilities that Webmunk offers for conducting research studies. The list presented here is not exhaustive, but rather is a function of our needs in the Amazon and Cookie projects. We hope that by leveraging this technology, other researchers can add to Webmunk's existing functionalities.

It is useful to compare Webmunk to an ad blocker, which many readers may be familiar

[3] `https://www.comscore.com/`.

[4] `https://www.airdna.co/`

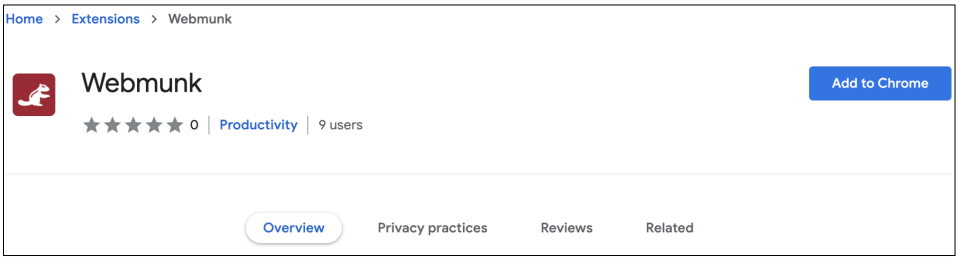


Figure 1: Webmunk in the Chrome Extension Store

with. From the user's perspective, Webmunk can be found on the Chrome store and can be installed like any other Chrome extension (see Figure 1). After installation, the user is prompted to enter their email address (see Figure 2), which will create a unique (anonymous) identifier.[5] An enrollment server, separate from the main server where data are collected, will store the mapping between the encrypted email addresses and the anonymous identifiers. Emails are used to send compensation, typically in the form of gift cards, for participating in research studies. After enrollment, Webmunk operates in the background while the browser is open, and can be found in the list of installed Chrome extensions (Figure 3).[6]

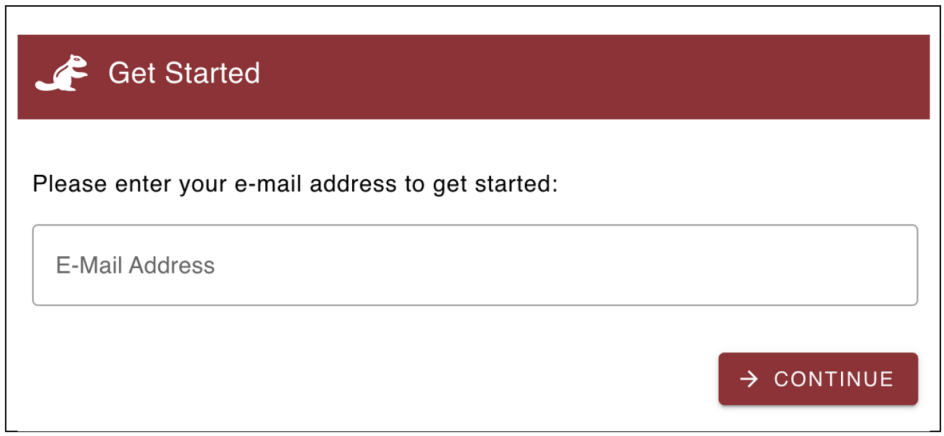


Figure 2: Webmunk Enrollment

The capabilities of Webmunk can be broadly classified into two main groups: tracking capabilities and manipulation capabilities. We devote a subsection to each.

[5] Webmunk can also handle other identification schemes such as usernames or phone numbers.
[6] Extensions on the Chrome browser can be accessed by navigating to `chrome://extensions/`.

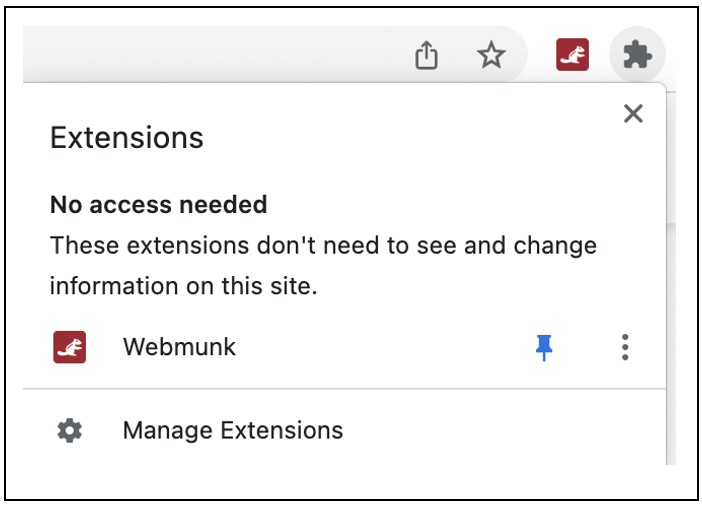


Figure 3: Webmunk after Installation

1.1 What Can Webmunk Track?

Webmunk can track a variety of digital data that the consumer generates while browsing the Internet. We have included a number of tracking modules that we developed for our ongoing studies. Researchers can easily include a subset of the existing data tracking capabilities or develop other modules to track additional behavior and data.

When it comes to tracking user behavior online, researchers need to strike the right balance between tracking everything, which may put private data at risk or dissuade users from participating in a research study, and not tracking enough, which prevents the researchers from conducting a thorough study of user behavior. The approach we take with Webmunk is a conservative one, where the default is not to track, unless explicitly stated otherwise. This requires considerable pre-analysis of the behavior that researchers are interested in collecting. Technically, this means that Webmunk includes an extensive list of conditions that need to be met, jointly or independently, for data to be tracked.

In addition to ex-ante conditions for tracking, researchers can ensure the privacy of certain confidential information ex-post. For example, researchers can implement models to identify searches that likely contain personally identifiable information (such as addresses or first and last names) and remove them from the data ex-post. Researchers should also consult their respective Institutional Review Boards and information technology support resources to ensure data are stored in accordance with best practices.

We classify the types of data that Webmunk can track into five categories: browser settings, web history, website content, user activity, and cookies. We describe each of them below.

Browser settings. Webmunk can track the settings of a user's browser. Browser settings include anything that applies generally to the user's browsing activity and is specified at the level of the browser (see Figure 4). For example, Webmunk can see a user's default search engine (e.g.; Google, Bing, DuckDuckGo). Other settings of potential interest to researchers include accessibility settings, whether the current browser is the default browser, and general cookie settings (such as whether the user has selected to block all third-party cookies).

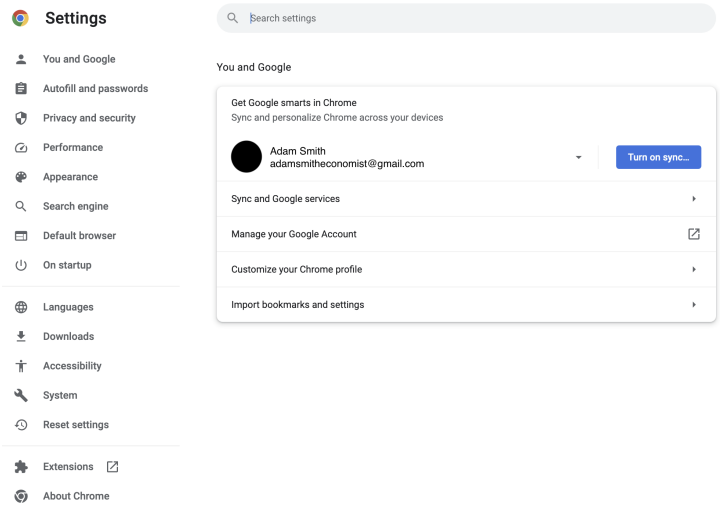


Figure 4: Browser Settings on Chrome

Web history. Webmunk can track URLs that users visit, as well as associated metadata such as page titles and times of visits. This is already a very useful set of data because it contains referrer information, which can help researchers identify where a user clicked to visit the current page. Athey, Mobius and Pal (2021) use this type of data to study the effect of news aggregator on the consumption of news. In our Amazon project, we use referrer URLs to infer how users reach product pages on Amazon, and find that almost half of all product pages are visited by clicking on a search result on Amazon.com.

In addition to the use of web history data on their own, information contained in URLs can also help validate other data collected by Webmunk. For example, we describe below how Webmunk collects the terms used for searches on Amazon. Since Amazon search result URLs contain the query parameters, we can validate that the search terms collected by Webmunk constitute the complete set of search terms used by a specific user.

Website content. Webmunk can also track information included in the websites that a user visits. Two types of website content are of particular interest: content generated by the website itself and content generated by third parties (e.g., ad exchanges). The content generated by the website itself is frequently HTML-based, which makes it easy to track. One could simply track

all HTML content contained in a page, or further separate it into different types by identifying specific HTML tags. For the Amazon project, for example, it is very convenient to identify the HTML snippet constituting an individual product presented in a search results page (Figure 5). Webmunk identifies each individual search result and stores its HTML separately from that of other search results, which can be particularly useful for ex-post processing if researchers are interested in identifying the products shown when consumers search on Amazon and their characteristics. Because it also records the individual product's position on the webpage, researchers can also study the position of each HTML element relative to one another, which is important when researchers want to study how products are ranked by Amazon (Farronato, Fradkin and MacKay 2023).

In addition to making ex-post data processing easier, a main advantage of identifying elements of a web page that researchers are interested in tracking is to avoid the collection of confidential data. Take checkout pages on Amazon.com for example. Those pages typically contain the buyer's full name, payment information, and shipping address. The risks of confidentiality breach if researchers tracked that type of information are often too high relative to the benefits of collecting it. A tool that is able to exclude the tracking of that type of data is certainly valuable both to incentivize users to participate in the study and to comply with IRB requirements.

Figure 5: A Search Result on Amazon

The second type of content is generated by third parties and embedded in the web page visited by the user. For example, online ads are often displayed on a web page by a supply-side platform—a third party—that collects ads from ad exchanges and displays them to the final user who visits the web page. Such content is often contained in inline frames (iframes), or HTML elements that load other HTML content within the document (effectively, a web page within a web page). The hierarchical structure of iframes, combined with the fact that each level of the hierarchy is controlled by an intermediary trying to protect their own data from

being accessible to other players in the chain for security or strategic purposes, [7] makes it challenging to efficiently track the content of iframes. Webmunk can capture the position and size of an iframe, any text and URL contained within the iframe, including the URLs of images or videos displayed.

The combination of these data allows us to partially learn about an ad served in an iframe, which we use in our Cookie project to identify the effects of cookie preferences on the type and quantity of ads shown to a study participant. A limitation of this strategy for tracking ads is that we do not know the final URL a user would land on if they clicked on an ad. The final destination is often obfuscated by a sequence of URLs, or re-directs, that are loaded in sequence when a user clicks on an ad. The iframe only contains the first URL in that sequence, and that first URL is typically uninformative about the identity of the advertiser or the content of the ad. This is where other digital traces collected by Webmunk can become helpful. Conditional on a user clicking the ad, the final URL in the sequence of re-directs an be tracked by Webmunk as an element in the web history (described above). This URL contains important referrer tags that allow researchers to identify when a user clicks on an ad located on a specific web page.

In addition to the static content of a web page, Webmunk captures the dynamic events leading to the display of such content on the page. In practice, this means that we know a page life cycle, such as the time when a page is uploaded and the time when it is updated (for example, because the user scrolls past the initially loaded content). This allows us to identify on which portions of the page the user stops in order to distinguish between content that is loaded but unlikely to be seen by the user, and content that is loaded and visible to the user.

User activity. Since users interact with the content of web pages, it is often important to collect information on user activity. Clicking, scrolling, and adding text to forms embedded on a webpage are the types of activities that allow us to study, for example, how consumers search online (which terms do they search for?), the extent of their search efforts (how far down do they scroll in search results?), and the products they visit (which search results do they click on?). Webmunk has extensive selector language to identify which types of clicks, scrolls, and forms to track to avoid the collection of confidential data (such as payment information) that are unnecessary for the research purpose.

Cookies. As a user browses the Internet, the web servers they interact with generate cookies. These cookies, stored by the user's web browser, help websites be informed about the user, for example, by remembering the login credentials or the products in a shopping cart so that the user does not have to sign in or add products to the cart again. Webmunk can track all the cookies that a user has saved on their browser, which allows us to better understand how

[7] Reis, Moshchuk and Oskov (2019).

websites use cookies for advertising purposes, and how cookies change as a function of users' privacy preferences. We will come back to cookies in the next Section, where we describe the ways in which Webmunk can block or allow cookies depending on users' states preferences.

Crawling. The types of tracking functionalities described so far all require the user to actively browse the Internet. However, Webmunk can also crawl the Internet on the user's behalf to track additional information. We use this functionality in the Amazon project to obtain information on a user's past purchases on Amazon,[8] or in the Cookie project to obtain existing cookies stored by a user's browser. Note that this is a type of functionality that can be especially intrusive if conducted in the background, so it should be used sporadically and with the user's explicit consent. In our Amazon project, for example, we let the user click on the task "Upload your Amazon order history" (see Figure 8), after which we crawl the URL `https://www.amazon.com/gp/your-account/order-history` on their behalf while updating the user on our progress.

1.2 What Can Webmunk Manipulate?

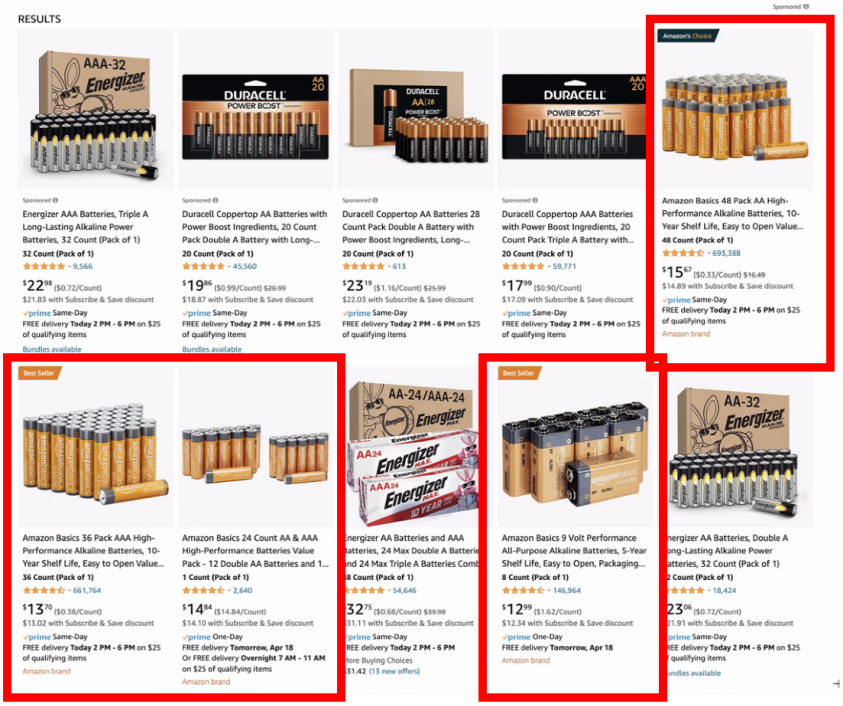
Browser extensions have the capability of manipulating many aspects of a user's experience while using the Internet. In this section, we describe the capabilities we have developed so far for our studies.

Removing content. Webmunk has the ability to arbitrarily modify the HTML code that determines the content of a web page. This will change how a website looks for the end user. One possible modification is to hide parts of a web page whose HTML matches a specific pattern. For example, suppose that an e-commerce site denotes parts of the page that contain products with a specific HTML tag or link URL. It is simple to specify the HTML tag or URL, and to tell Webmunk to hide or highlight any parts of the website that contain a match. Figure 6 shows an example in which Webmunk identifies Amazon brands and hides them. For this to happen, Webmunk identifies HTML components denoting each individual search result. For each search result, if the product title contains a list of pre-determined words associated with Amazon brands, the extension can remove the entire product from appearing on the page. This is similar to the type of logic used by ad blockers to prevent ads from appearing on a web page.

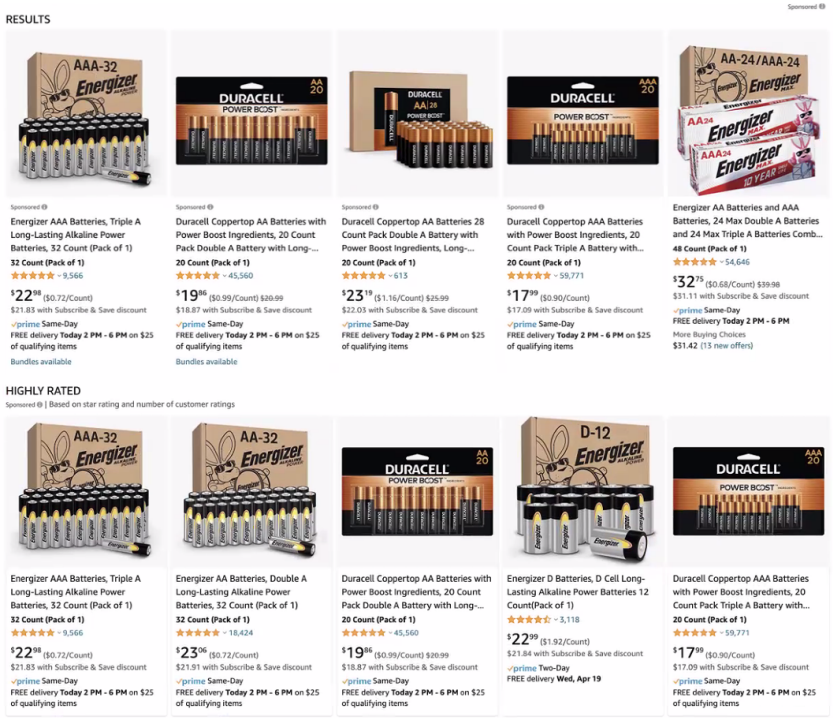
Changing the look and feel of website content. It is also simple to use Webmunk to change the look of specific content associated with a tag by, for example, highlighting it, changing its color, or changing its font.[9] This can be particularly useful not just for functional-

[8] `https://www.amazon.com/gp/your-account/order-history`.

[9] The mechanism for these manipulations is the addition of a special CSS tag into parts of the HTML. The manipulations are specified in a CSS stylesheet. Section 4 describes the technical aspects of this process.



(a) Amazon Brands Highlighted



(b) Amazon Brands Removed

Figure 6: Amazon Brands on a Search Results Page

ity purposes, but also to debug and validate the extension's intended objectives. For example, prior to removing parts of a web page, it is often useful to highlight them to ensure that all and only the parts intended to be removed are properly identified.

Another related capability is to change the position of existing content. Websites often present content such as search results, products, and social media posts in sequence, where some results are given higher priority than others. Many research questions concern the role of the content position on the web page on how consumers engage with that content. For example, sponsored search ads are often placed higher than organic content, and researchers are interested in whether and how much this positioning advantages the sponsored content (Blake, Nosko and Tadelis, 2015). One way to study this, which Webmunk can easily implement, is to shift the position of sponsored search ads to be below some of the organic content.

Inserting new content and new interactions. In addition to changing existing content on a webpage, Webmunk has the ability to inject new content and interactions into a website. One use case for this is for light-weight surveying of participants as they are browsing specific websites. For example, suppose we were studying perceptions of news articles. Webmunk can insert a survey every time a user landed on a page with a news article. Such a survey can be made to appear overlaid on the webpage or can appear as a separate pop-up.

In the cookie project, we are asking people about their cookie tracking preferences. We do so using a similar injection of content. In particular, upon navigating to a website, a cookie form appears while the website content fades to gray in the background (see Figure 7). Users must select an option or click 'x' to continue to the website content. Webmunk records the user's selection.

Changing cookies. When people browse the Internet, the websites they visit place cookies onto their browser. These cookies are used for a variety of reasons, e.g., to insure automatic logins and to track user behavior. Webmunk can interact with cookies loaded onto the browser. In particular, it can remove any cookie that has already been loaded onto Chrome. This allows Webmunk to prevent the tracking of a user by specific websites.

Acting as a user proxy. Webmunk can act on behalf of the user in filling out forms and selecting options on a webpage. For example, many pages contain cookie consent forms. Webmunk can identify such forms, and can select a preferred option automatically.[10] As another example, Webmunk can automatically click out of promotional banners on e-commerce websites that aim to collect a user's email address.

[10]Several popular browser extensions, such as Consentomatic, implement this type of functionality.

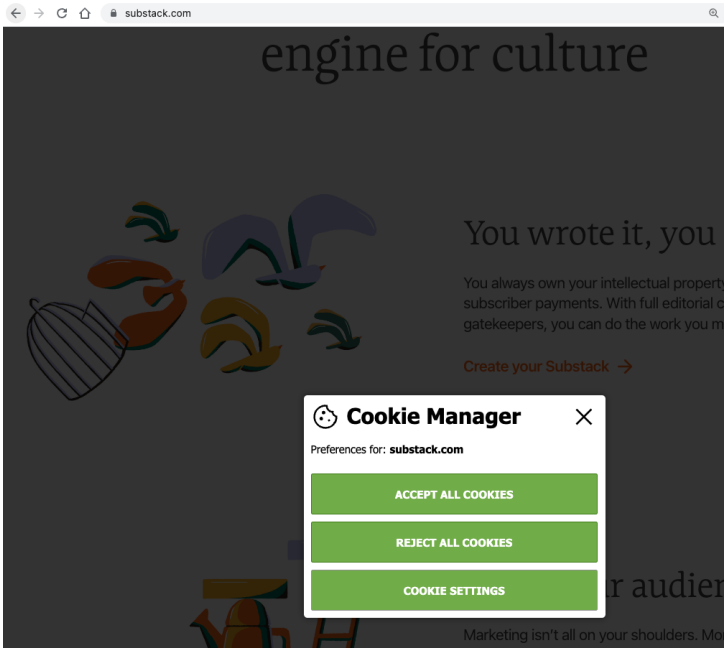


Figure 7: Custom cookie consent form

2 Case Studies

In this section, we describe two research questions under studies for which Webmunk has proved useful in collecting user behavior. We start with an application to Amazon to study self-preferencing, and continue with a study of cookie perferences.

Self-Preferencing on Amazon. Technology platforms such as Amazon and Google frequently offer their own products alongside products sold by competitors. This ownership structure creates the potential for platforms to give their own products an advantage over others, a practice often referred to as "self-preferencing." In Farronato, Fradkin and MacKay (2023) and ongoing work, we seek to study whether Amazon engages in self-preferencing on its own marketplaces and how this affects consumers. To do so, we customized Webmunk to track user behavior on Amazon and to experimentally modify the products displayed when users visit Amazon. Here, we describe how Webmunk allows us to collect data for this research objective. [11]

Users in the study install Webmunk from the Chrome Web Store (Figure 1) and register with their email address (Figure 2). The email address serves two purposes. First, we use it to check that the user gave their explicit consent to participate in the study, by matching the email address to answers to the initial survey via the Qualtrics API. Only users who are eligible,

[11] The study was approved under Harvard IRB21-1677.

consented, and gave matching emails in the initial survey and on the browser extension are enrolled into the study. Second, we use the email address to send participants gift cards as compensation for participating in the study.[12]

Enrolled participants must complete a series of tasks, such as surveys, that are part of the study. These tasks are shown in an extension pop-up window (Figure 8). Users click on each task in order to start or continue it. After completion, the task disappears from the pop-up window. The server can be configured to send participants email reminders about the remaining tasks.

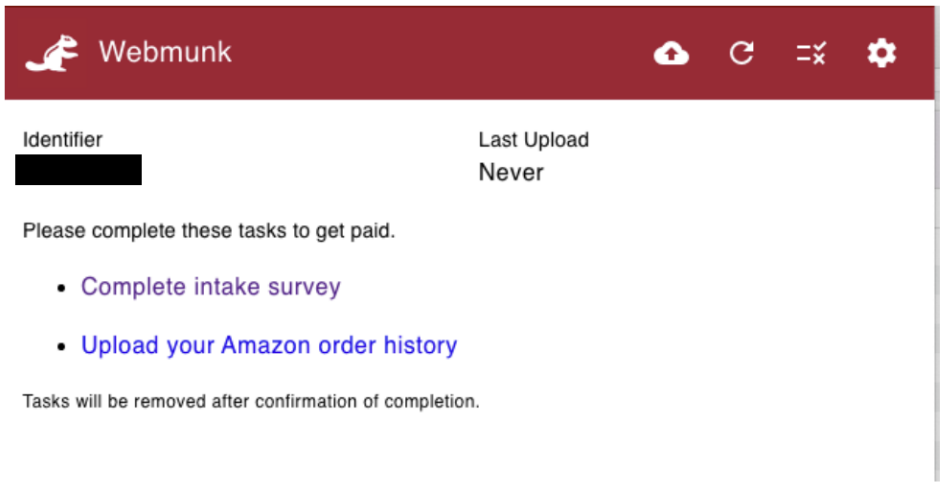


Figure 8: Webmunk Tasks

For the Amazon project, Webmunk's tracking abilities are focused on the amazon.com web domain. Webmunk tracks the information displayed on Amazon pages and user actions on these pages, with the exception of what we ex-ante identify as payment and shipment details. For every Amazon search results page for example, Webmunk records the search terms and each product displayed (we use Amazon Standard Identification Numbers, or ASINs, to identify products). Product characteristics such as product title, price, shipping options, and reviews, are tracked by storing the HTML portion of the page corresponding to each product. Webmunk also tracked the extent to which participants scroll on each page, and updates product positions accordingly upon scroll. Any time a user clicks on an item or adds an item to cart, the behavior is logged. Tracking works similarly for other Amazon pages, including product pages, checkout pages, and wishlist pages.

[12] Webmunk employs industry-standard encryption protocols so that personally identifiable information about users and their actions is not stored in plain-text or observable in transit over the Internet. Additionally, to obfuscate identities further, the mapping between the email address and the user's anonymous identifier is stored separately from the user data collected by Webmunk, which are associated to the anonymous identifier.

Webmunk manipulates the availability of products across participants. Each user is assigned to one of three treatment arms. Participants in the control arm see all the products that Amazon selects to display. For participants in the *Amazon treatment*, Webmunk removes products carrying a brand owned by Amazon, such as 'Solimo' or 'Amazon Essentials.' For the removal to occur, Webmunk checks for the presence of terms matching Amazon brands in the HTML portion of each product. Any white space generated is seamlessly filled by the automatic re-positioning of nearby items (Figure 6). The third and final treatment condition is one where Webmunk removes a random subset of products from those available. As before, any HTML element where multiple products are presented jointly is deleted from product pages.

Cookie Consent. Data are critical to the functions of modern businesses, whether as an input into decisions such as pricing or search ranking algorithms, or for the purposes of targeted advertising. At the same time, the collection and use of data is a threat to consumer privacy. Over the past decade, regulatory and societal pressures have led companies to offer consumers choices about the collection and usage of their data. In order to incentivize consumers to share data, businesses often structure these choices to make it hard for users to select options that limit data sharing, a phenomenon called 'dark patterns' by user interface designers.

Figure 9 provides an example of a dark pattern on Amazon.co.uk. Users are offered two options: "Accept cookies" or "Customize cookies." The option to reject non-essential cookies is hidden under the option to customize them. This setup makes it much easier for users to accept cookies rather than to search through the 'customize cookies' settings. This type of pattern is considered by some to be a substantial problem, and the European Union's Digital Services Act explicitly bans these practices.[13]

In an ongoing project, we are interested in assessing the impact of different types of dark patterns on consumer privacy choices. To the extent that consumers have difference preferences over data tracking by small and large companies, we also explore how dark patterns may exacerbate or ease data advantages of large companies.

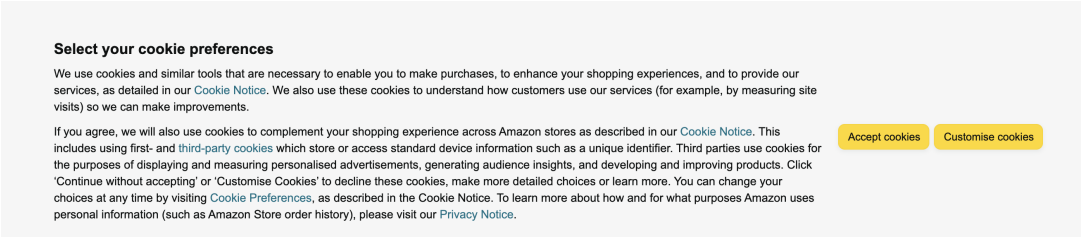


Figure 9: Cookie consent form on Amazon.co.uk

[13] https://ec.europa.eu/commission/presscorner/detail/en/QANDA_20_2348.

In this project, Webmunk allows us to change the cookie consent interfaces seen by users as they browse the internet. Our interface can appear on any website. Figure 10 displays some of the interfaces that we designed.

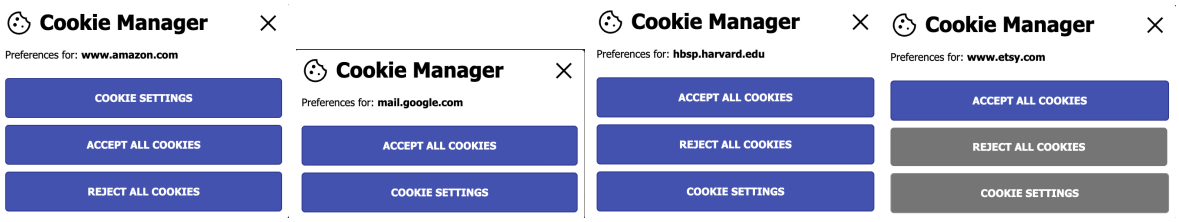


Figure 10: Examples of Webmunk's cookie consent interfaces

For participants in the study, upon navigating to a website, a cookie form appears while the website content fades to gray in the background (see Figure 7). Users must select an option or click 'x' to escape the form and continue to the website content. Webmunk records the user's selection, and enforces their choices to the extent possible through string matching on the website.

The experimental component of this study consists of randomly assigning cookie interfaces, and the frequency with which these interfaces appear. For every user and web domain combination, Webmunk selects one of six different interfaces, four of which are displayed in Figure 10 as examples. During participants' organic browsing, the pop-up appears every 10 minutes for half of the participants, while it appears every 60 minutes for the other half.

The study's experimental variation allows us to estimate the causal effects of the interfaces on cookie choices, and how this varies across users and domains. Additionally, the random frequency of cookie pop-ups allows us to identify how users' choice change as a function of how frequent they are.

3 Webmunk Project Requirements (what to look for in hiring a developer)

We anticipate that most social science researchers will need to hire a software developer or partner with a computer scientist to customize Webmunk for their research purposes. This section provides guidance regarding what technical skills are necessary to work with Webmunk. Additional technical details are available in the documentation for Webmunk. Three main technical skillsets are required: server administration, Django development, and Chrome web extension development.

First, the Webmunk cloud infrastructure is built on the open-source Django web application server, backed by a Postgres database for storage. This deployment assumes a POSIX-compliant modern Unix operating system, such as Linux or FreeBSD. To successfully set up the server foundation, an experienced Unix administrator is required to initially set up the server, install the Webmunk prerequisites and server software, and configure the necessary web endpoints and background tasks required by the Webmunk system. Note that the entire Webmunk platform is constructed with popular modern open-source components, so deploying to a recent release of a popular Linux distribution such as Ubuntu, CentOS, or Red Hat Enterprise Linux is advised.

The requirements for the server (or servers) itself will vary based on the size of the research population, quantity of data collected, and duration of the study. For a simple deployment of 50 participants collecting basic browsing data (URLs, page titles, click events, scrolling), four processor cores, with 16 GB of RAM, and 64 GB of disk storage should be sufficient. More involved studies involving thousands of participants and more involved data collection (such as large copies of page content or capturing embedded media resources) may need to split the participant load across several servers. In our Amazon study, we deployed five Amazon xlarge-class servers with several terabytes of storage to accommodate our participant load. Since server requirements will vary dramatically based on the actual study design, initial pilot studies with a larger pilot pool are recommended to generate specific per-participant usage estimates for the larger study.

The second technical competency required is that of a competent Django developer, ideally with additional experience with data processing and mathematics platforms such as Pandas, NumPy, and SciPy. The responsibility of this developer will be to fine-tune the default Webmunk server platform for specific study needs, create any new data exporters to feed observational information into the study's analysis pipeline. Given that the volume of data gathered from Webmunk can be substantial, an experienced Django developer with experience optimizing large database queries and data processing algorithms in Python will save the study significant time and resources compared to an entry-level or junior developer without experience managing and running a Django site at scale.

The final technical skillset required is that of an experienced Chrome web extension developer. Since Webmunk is a platform for creating extensions, as opposed to a large monolithic all-in-one extension, studies will need someone capable of registering, building, and maintaining extensions on the Chrome Web Store. This not only includes the technical skill to build the extension to submit, but comfort and competency navigating Google's extension registration process and dealing with any issues that their reviewers may raise.

On the technical front, the extension developer will be responsible using standard Javascript infrastructure such as WebPack to customize and build the extension. Should a study's data needs not be entirely met with an existing Webmunk extension, this developer will be responsible for developing the base extension that hosts the various data and manipulation modules (using a Webmunk-provided template), collaborating with their Django counterpart to set up and test the enrollment and identity verification, as well as implement any custom user interfaces for the extension itself that may be required by the study. In the event that a Webmunk module does not yet exist to address a specific data collection need, this developer will be responsible for creating new Webmunk modules to capture the data of interest and integrating it into the hosting extension. We designed the Webmunk extension components so that a competent Chrome extension developer should be able to understand the overall platform quickly and begin working with it using the same tools and techniques that they would use in building any other Chrome Manifest V3 extension.

4 Technical and Implementation Details

The Webmunk platform has been designed from the start to be a reusable tool, not just a single web browser extension created for the purposes of a single study. Drawing upon over a decade of experience of conducting observational web studies through browser extensions, proxy servers, and external software tools, Webmunk embodies the lessons learned from past endeavors and anticipates future research questions and studies to which it will be applied.

4.1 Technical Details About Webmunk Browser Extensions

Browser extensions are bundles of software intended to affect or enhance the web browsing experience in some specific way. The most well-known type of extension is the ad blocker, which is designed to eliminate advertisements and other tracking technologies from web pages where the ad blocking extension is installed. Webmunk extensions use the same techniques and application programming interfaces (APIs) and techniques employed by more prominent extensions to create a platform that both enables observational studies at the page level and provides mechanism for implementing experiments and interventions within subjects' browsers to test behavioral hypotheses.

Every browser extension features two main functional components: content scripts, which interact with web pages and the users directly, and service workers, which provide functionality and services in the background, divorced from any direct user interaction or specific pages

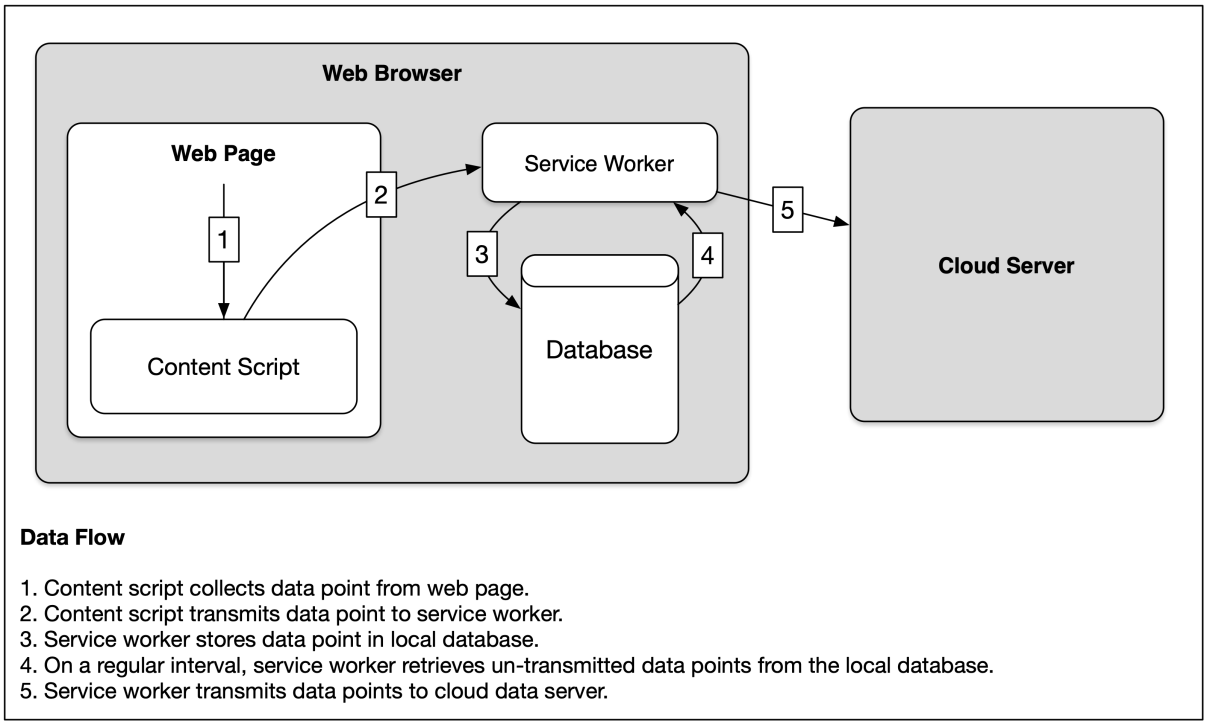


Figure 11: Webmunk Data Flow Diagram

loaded. For a variety of security and functional considerations, the browser permits each to accomplish specific tasks. A content script may directly observe and manipulate a page element in the page it is loaded, while a service worker can transmit data to a server, even if no pages are open. In most cases, an extension's functionality is implemented using a combination of the two components, which may communicate via messages sent to one another.

For example, in a basic Webmunk extension, the browser notifies the service worker when a new page is opened and then the service worker issues the appropriate commands to inject the content script into the newly-opened page. Note that the service worker may also make a decision to not inject the content script into the page, which may be important if the extension is targeting a narrow set of sites.

Once the content script is injected into the page, it may observe and manipulate the page's basic structure, the Document Object Model (DOM), a tree-like structure representing the contents of the page. Perhaps the simplest example of this capability is recording the page title and reporting it back to the study data collection server. This process happens in four steps.

1. The content script will send a message back to the service worker to store the data.

2. The service worker receives a message from a content script to store the data.

3. The service worker stores the data in a local database within the browser.

4. A background process runs within the service worker to inspect the local database for any pending data points, and if any are encountered, to transmit that data to the server. This is done on a set interval (typically every 5 minutes).

5. If the transmission is successful, the service worker removes the data from the local database.

This approach minimizes the potential for any data loss, is robust against problems in the network infrastructure, and allows the Webmunk extension to function in offline configurations as well as online ones.

An example of a module's content script, which identifies and send back the page's title is shown below. Note that it uses jQuery to identify a page's title and that it sends a message back with a particular data structure.

```
(function () {
  window.registerModuleCallback(function (config) {
    // If enabled for this site...
    if (config.enabled) {
      // Fetch page title using jQuery...
      const titleValue = $('title').val()

      // Transmit page title to the service worker as a
      // "webmunk-page-title" data type, including the
      // URL of the page...
      chrome.runtime.sendMessage({
        content: 'record_data_point',
        generator: 'webmunk-page-title',
        payload: {
          'url*': window.location.href,
          'page-title': titleValue
        }
      })
    }
  })
})
```

In the example above, the extension transmitted the title of the page. The extension's content script may also install various event listeners within the page to detect user activity like mouse clicks, page scrolls or other interactions with the page. This event data is different from the title data, and as part of the process of requesting the service worker store the data for transmission, the content script will provide a data identifier to distinguish one type of data from another. The title data might be identified by `webmunk-page-title`, while the user interaction events may provide several identifiers for each interaction such as `webmunk-mouse-click`, `webmunk-page-scroll`. Additionally, meta-data about the user's actions is returned as `passive-data-metada`

Each of these different data types' payloads will have their own structure, which we show below for illustrative purposes.

A page title:

```
{
  "date": 1695655181303,
  "generatorId": "webmunk-page-title",
  "page-title": "Example Page Title",
  "url*": "https://www.example.com"
}
```

Page scrolling:

```
{
  "date": 1695681303551,
  "generatorId": "webmunk-page-scroll",
  "scroll_top_offset": 900,
  "scroll_left_offset": 0,
  "viewport_width": 1920,
  "viewport_height": 1080,
  "page_width": 1920,
  "page_height": 2160
}
```

Clicking on a button that is labeled with the `id` attribute "submitButton" on the page.

```
{
  "date": 1695655181303,
  "generatorId": "webmunk-button-click",
```

```
    "page_top_offset": 642,
    "page_left_offset": 45,
    "clicked_element_id": "submitButton"
  }
```

In addition to user actions and page contents, we also store standardized metadata about the time at which actions happen and about the user's browser. An example is shown below.

```
  "passive-data-metadata": {
    "source": "20989909",
    "generator-id": "webmunk-extension-matched-rule",
    "generator": "webmunk-extension-matched-rule:
    Study Browser Extension/0.38 Mozilla/5.0 (Windows NT 10.0; Win64;
↪ x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/115.0.0.0
↪ Safari/537.36",
    "timestamp": 1692096550.764,
    "timezone": "America/Chicago"
  }
```

Note that the date field above is expressed in millisecond-level Unix epoch time, which counts the number of milliseconds since Jan. 1, 1970 at 12am in the GMT time zone. This representation is converted later into actual date types within the destination database, and is used as a simple and unambiguous method for representing when something occurred.

The metadata allows the upstream PDK server to file the incoming events in its database. The source field identifies the Webmunk participant, in this case with a remotely-assigned random identifier. The generator-id field encodes the data identifier, and the generator field acts as an effective user-agent, identifying the particular Webmunk module, containing extension, and web browser that generated the data point. The timestamp field is the standard method for timestamping data points. If the date field is omitted from the event, a current date will be added as part of the process enqueuing event for transmission. Finally, the timezone field records the local time zone as the browser is currently configured. This is included so that the universal timestamps included with the event can be translated into the participant's local "clock time".

In addition to information transmitted as a tree-based JSON structure, Webmunk can also transmit binary file payloads such as images and videos. The PDK server will unpack these payloads and store the binary data in files accordingly.

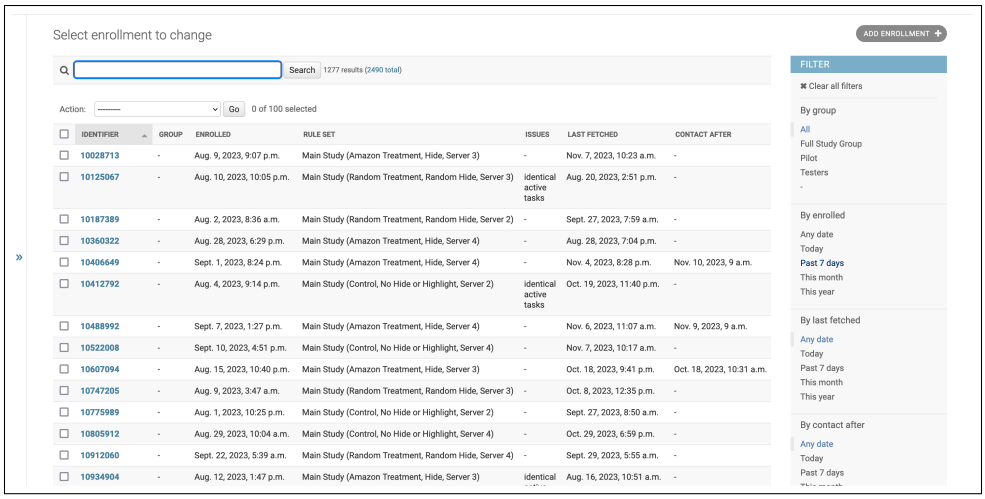


Figure 12: Enrolled participants and their assigned experimental arms

4.2 Technical Details About Webmunk's Cloud Infrastructure

Webmunk cloud servers are used to specify the configuration of the browser extension and to collect data uploaded from users. There are two servers that are part of each project, one that handles enrollment of participants and another that collects data. These are separated for privacy reasons.

The enrollment server validates a participant's identity, assigns them a suitable opaque identifier to attach to the data (the source field in the examples above), and provides updated configurations during the course of a study. We've typically tracked identity using users' emails. Other identity methods are simple to incorporate. Each email is mapped onto an id, which is then used by the data server. In instances where the participants are enrolled before installing the extension, study identifiers may be uploaded to the enrollment server, and only participants entering a valid identifier provided to them will be allowed to use the Webmunk extension.

When a Webmunk extension validates an identifier successfully, the enrollment server returns a configuration containing the assigned identifier, as well as any additional settings to apply to the extension or its modules. For example, in our Amazon study, we could configure which click elements to track as part of our configuration:

```
"rules": {
  "actions": {
    "webmunk-add-to-wishlist-button": {
      "on-click": [
        {
          "name": "log-click"
```

```
        }
      ]
    },
    "webmunk-checkout-element": {
      "on-click": [
        "log-click"
      ],
      "on-hide": [
        "log-hidden"
      ],
      "on-show": [
        "log-visible"
      ]
    }
  }
}
```

The `on-click`, `on-hide`, and `on-show` parameters instruct our Amazon module to send the appropriate events to the server when elements they are attached to (identified by the `webmunk-add-to-wishlist-button` and `webmunk-checkout-element` HTML classes added to elements elsewhere in the extension) are triggered. We used other configuration directives to specify when to inject content scripts into pages and when to skip those injections, such as when a user visits Amazon's Prime Video streaming service on their browser, which was not relevant to us in our particular study.

The configuration directives also specify to which data server the extension should transmit its data. This allows studies to scale horizontally as new participants begin to exceed existing server resources. New data servers may be brought online and new participants' extensions will be configured to send to those new servers instead of the fully-utilized existing servers. This horizontal scaling capacity can be used to add more capacity to existing studies, or may be used to segregate different users' data to different servers to implement any local access control or to implement study-specific experimental blinding requirements.

Configuration templates may also be used assign participants to different experimental arms. In our Amazon study, we had a control condition that did not change the participant's Amazon experience and only logged data, one experimental condition that hid Amazon-branded products, and one condition that hid both Amazon and third-party products. New participants were assigned to an appropriate configuration template on initial enrollment. The

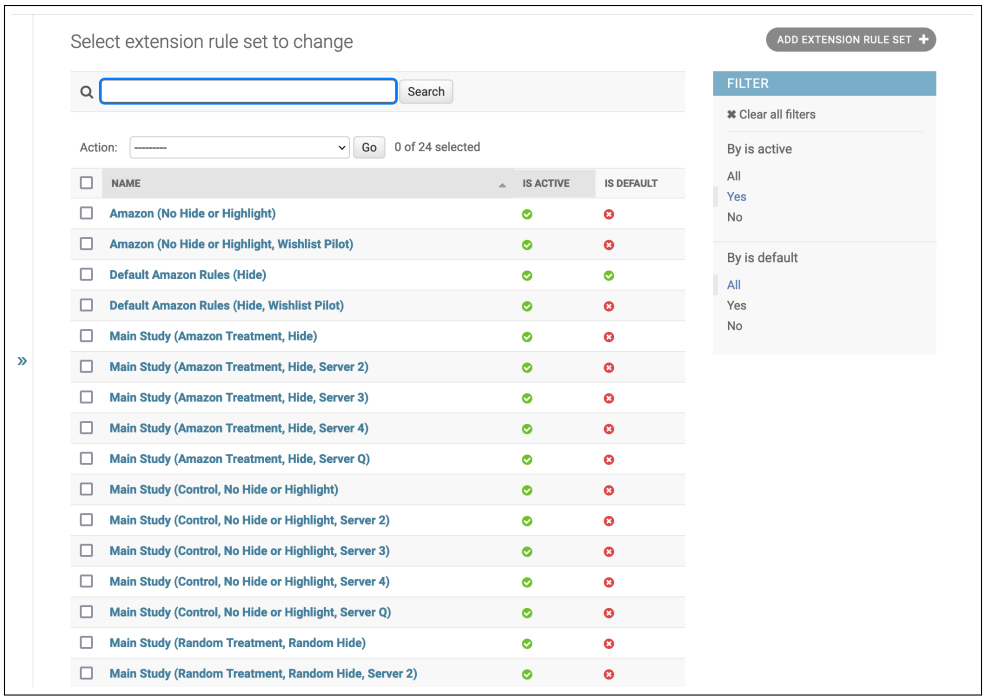


Figure 13: Various experimental arms used in the Amazon study

parameters implementing the experimental arms were entirely expressed in the configuration that the extension fetched from the server, not hard-coded within the extension itself. This allowed us to modify and adapt the extension in the field without the need to encode the details of the study's experimental arm design in the extension itself. This allowed us to fine-tune and adapt arm parameters as the study progresses without the delay and overhead of needing to submit new versions of the extension to Google for review every time that we needed to make a change.

In addition to configuring the behavior and appearance of the extension, we also implemented a simple URL-based task system. This allowed us to schedule activities for the participants to complete – such as filling out a mid-study Qualtrics survey or uninstalling the extension at the end of the study. Tasks appeared within the extension's window as clickable links that open in the browser. Tasks appear as soon as an activation date arrives, and do not go away until validated as completed, such as the server pulling down a completed set of responses from the Qualtrics server. As long as study tasks can be expressed in URLs (and corresponding labels), this is an effective mechanism to engage participants outside the extension itself.

5 Conclusion

This paper has presented Webmunk, a new open-source framework for conducting digital experiments. Webmunk consists of a web browser extension that is extensible via modules, and a data collection platform. Webmunk is able to comprehensively track user activity and website content, making it suitable for observational studies. Webmunk can also change the content of websites, modify cookies, and remind users to take specific actions, making it suitable for sophisticated experimental designs.

This paper has explained how we've used Webmunk in studies concerning Amazon's self-preferencing, and data sharing online. However, the goal of this paper is to give other researches the information necessary to adapt Webmunk for new research designs. To aid in this, we've described the basics of the browser extension and server side infrastructure built into Webmunk. We've also described the three skills reserchers should look for when hiring software developers to customize Webmunk. These are server administration, Django development, and Chrome web extension development.

The paper concerns the software used to run web studies, but there are other elements of web research studies that warrant further exploration. One such direction is the best way to preserve user privacy while sharing data with other researchers. Here, differential privacy seems a promising direction. Another area of interest is in efficiently recruiting users to web browser studies. Several recent studies, including our own, have recruited via advertising on Meta, but this recruitment is very expensive. More work is needed to develop best practices for finding research subjects.

Lastly, Webmunk is built for laptops and desktops. The browser extension approach we take will not work on phones, since most phone activity happens via apps. We need better technologies and research designs to study mobile phone behavior, given its importance.

References

Allcott, Hunt, Matthew Gentzkow, and Lena Song. 2022. "Digital addiction." *American Economic Review*, 112(7): 2424–63.

Aridor, Guy. 2022. "Drivers of Digital Attention: Evidence from a Social Media Experiment." *Available at SSRN 4069567*.

Athey, Susan, Markus Mobius, and Jeno Pal. 2021. "The impact of aggregators on internet news consumption."

Beknazar-Yuzbashev, George, Rafael Jiménez-Durán, Jesse McCrosky, and Mateusz Stalinski. 2022. "Toxic Content and User Engagement on Social Media: Evidence from a Field Experiment." *Working Paper*.

Blake, Thomas, Chris Nosko, and Steven Tadelis. 2015. "Consumer heterogeneity and paid search effectiveness: A large-scale field experiment." *Econometrica*, 83(1): 155–174.

Calder-Wang, Sophie. 2021. "The distributional impact of the sharing economy on the housing market." *Available at SSRN 3908062*.

Conlon, Christopher, and Jeff Gortmaker. 2020. "Best practices for differentiated products demand estimation with pyblp." *The RAND Journal of Economics*, 51(4): 1108–1161.

Farronato, Chiara, and Andrey Fradkin. 2022. "The Welfare Effects of Peer Entry: The Case of Airbnb and the Accommodation Industry." *American Economic Review*, 112(6): 1782–1817.

Farronato, Chiara, Andrey Fradkin, and Alexander MacKay. 2023. "The Welfare Effects of Vertical Integration: Evidence from an E-Commerce Platform." *In Progress*.

Farronato, Chiara, Andrey Fradkin, and Tesary Lin. 2023. "Dark Patterns and Privacy Preferences: Evidence from a Field Experiment." *In Progress*.

Levy, Ro'ee. 2021. "Social media, news consumption, and polarization: Evidence from a field experiment." *American economic review*, 111(3): 831–870.

Lewis, Gregory. 2011. "Asymmetric information, adverse selection and online disclosure: The case of eBay motors." *American Economic Review*, 101(4): 1535–1546.

Nosek, Brian A, Jeffrey R Spies, and Matt Motyl. 2012. "Scientific utopia: II. Restructuring incentives and practices to promote truth over publishability." *Perspectives on Psychological Science*, 7(6): 615–631.

Reis, Charles, Alexander Moshchuk, and Nasko Oskov. 2019. "Site Isolation: Process Separation for Web Sites within the Browser." 1661–1678. Santa Clara, CA:USENIX Association.

Santos, Babur De los, Ali Hortaçsu, and Matthijs R Wildenbeest. 2012. "Testing models of consumer search using data on web browsing and purchasing behavior." *American economic review*, 102(6): 2955–2980.

Shen, Zejiang, Ruochen Zhang, Melissa Dell, Benjamin Charles Germain Lee, Jacob Carlson, and Weining Li. 2021. "LayoutParser: A unified toolkit for deep learning based document image analysis." 131–146, Springer.

Spellman, Bobbie, Elizabeth Gilbert, and Katherine S Corker. 2017. "Open science: What, why, and how."