UNIT IV WEB ANALYTICS

Web Analytics - Present and Future, Data Collection - Importance and Options, Overview of Qualitative Analysis, Business Analysis, KPI and Planning, Critical Components of a Successful Web Analytics Strategy, Web Analytics Fundamentals, Concepts, Proposals & Reports, Web Data Analysis

---

DATA COLLECTION - IMPORTANCE AND OPTIONS

There are several important elements to consider when it comes to implementing an effective data capture strategy:

There are a number of ways to collect data as a customer interacts with our websites. There are web log files, web beacons, JavaScript tags, and packet sniffers. Some advanced e-commerce software can also leverage built-in data collection mechanisms such as event logging to collect important business events and contextual data.

A number of data collection mechanisms are in the mold of "you forget it and you are screwed"

Sometimes we need more than one method of data collection. We might use JavaScript tagging, to collect website behavior. All sources of data that we need for making effective decisions: data related to outcomes on the website, or the various types of qualitative data such as from surveys or usability studies that need to be collected to understand customer behavior, or data from other sources in the company such as your CRM or Enterprise Resource Planning (ERP) systems.

Clickstream Data

There are four main ways of capturing clickstream data: web logs, web beacons, JavaScript tags, and packet sniffing.

Web Logs

Web logs have been the original source of data collection from the dawn of the Web. They were originally developed to capture errors generated by web servers and overtime have been "enhanced" to capture more data as analytical needs shifted from technical to marketing based. Figure 2.1 shows a simple schematic of how web logs capture data.

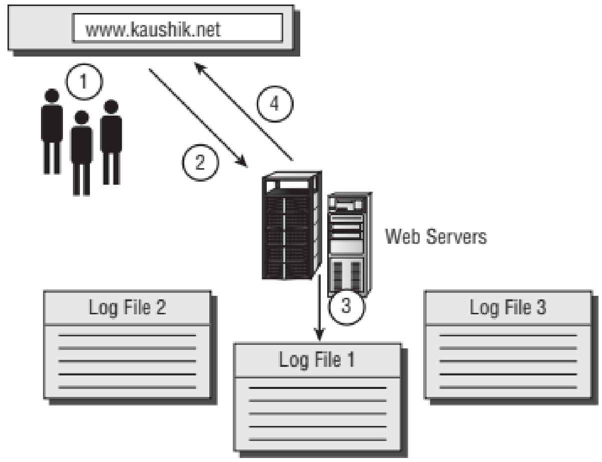


Figure 2.1 How web logs capture data

The data capture process is as follows:

1. A customer types your URL in a browser.
2. The request for the page comes to one of your web servers.
3. The web server accepts the request and creates an entry in the web log for the request.
4. The web server sends the web page to the customer.

In most cases, the web logs are taken from the server on a set schedule (say, nightly). A standard log-parsing tool or a web analytics tool can be pointed in their direction to analyze the web logs and produce standard reports.

Benefits of Using Web Logs as Your Data Collection Mechanism

- Web logs are the most easily accessible source of data. Every web server simply comes with mechanisms that collect the data and create web logs.
- Web logs are the only data capture mechanism that will capture and store the visits and behavior of search engine robots on your website.
- If you use web logs, you always own the data. With most other methodologies, the data will be captured, processed, and stored with your web analytics vendor, who operates under an application service provider (ASP).
- Web logs are primarily geared toward capturing technical information (404 errors, server usage trends, browser types, and so forth).
- If additional marketing and business information need to be captured, that capture requires a close collaboration with your IT team and a dependency on their release schedules.
- If the web server is not setting cookies, identifying visitors with any degree of accuracy is very challenging.
- Web logs were created to capture all the hits on the server. Therefore, when using logs, you have to be very careful and deliberate about applying the right filters to

remove image requests, page errors, robot traffic, requests for Cascading Style Sheets (CSS) files, and so forth, in order to get accurate traffic trends and behavior.

- Page caching by ISPs and proxy servers could mean that some of the traffic is invisible. With page caching common, your website pages are cached at the ISP or proxy servers. So when someone from that ISP's network requests your home page, it is served from the ISP and not your web server. Therefore, you will not have an entry for that request in your log files.

Web Beacons

Web beacons were developed during a time when banners ruled the Web as the prime way of "capturing eyeballs" and delivering those to "sticky" websites where we were measuring hits. A company would run banner ads across many websites, and often these would be similar banner ads. There was a distinct need to figure out not just how many people who saw the banner ads were clicking through, but also how many of those exposed to the ads were the same individual. Alternatively, if the same person was exposed to different creatives (banner ads, ad text, and so forth), which one worked best?

Web beacons usually are 1 × 1 pixel transparent images that are placed in web pages, within an img src HTML tag. The transparent images are usually hosted on a third-party server—different from the server that is hosting the web page. Figure 2.2 shows how data is captured by web beacons.

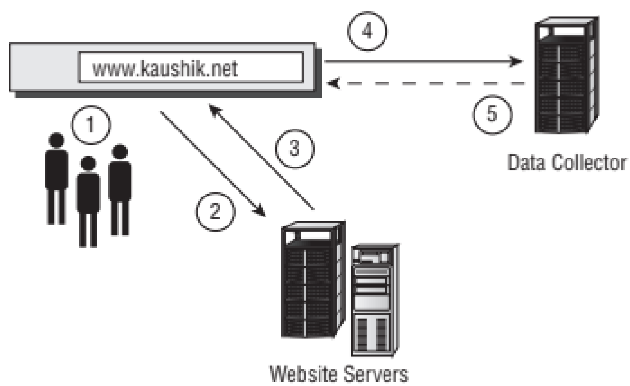


Figure 2.2 How web beacons capture data

The process is as follows:

1. The customer types your URL in a browser.
2. The request comes to one of your web servers.
3. The web server sends back the page along with a get request for a 1 × 1 pixel image from a third-party server.
4. As the page loads, it executes the call for the 1 × 1 pixel image, thus sending data about the page view back to the third-party server.

5. The third-party server sends the image back to the browser along with code that can read cookies and capture anonymous visitor data such as the fact that the page was viewed, the IP address, the time the page was viewed, the cookies that were previously set, and more.

Web beacons are also used in emails (such as email newsletters or promotional emails that we all receive). Here, just as in a web page, the transparent image is requested when the email loads in your email reader, and data about the email view is sent back and recorded. Typical data could include the fact that the email was read, by whom (email address), and any other parameters that might be appended at the end of the transparent image request embedded in the email. With the prevalence of JavaScript tagging, the use of web beacons has become less prevalent; they are mostly used to track basics around banner ads and emails.

Benefits of Using Web Beacons as Your Data Collection Mechanism

- Web beacons are easy to implement in most cases because they are just a couple of lines of code wrapped around an img src HTML tag request. Most of the “intelligence” in what to capture comes back from the server that received the image request.
- We can optimize exactly what data the beacon collects (for example, just the page viewed, or time, or cookie values, or referrers), and because robots do not execute image requests, you won’t collect unwanted data. This can keep your logs to a manageable size and won’t require complex filtering. Web beacons shine when it comes to collecting data across multiple websites or domains. The data captured is less deep than with other methodologies, but for targeted narrow purposes (banners, emails, and so forth), it works well.

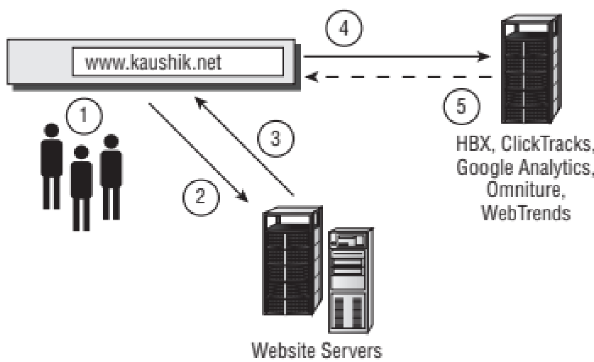
Concerns about Using Web Beacons as Your Data Collection Mechanism

- Beacons are most closely identified with advertising and advertising networks and therefore have a bit of a bad rap. As a result, many visitors apply global opt-outs, or the anti spyware programs automatically remove the cookies, which greatly hampers the ability to collect data.
- If image requests are turned off in email programs or some browsers, you can’t collect the data.
- Beacons are not as expansive and customizable as JavaScript tags in terms of the data they can capture. They capture less data but can do so across a broad range of websites.
- By their nature, beacons interact with third-party servers and for the most part set third-party cookies.

JavaScript Tags

JavaScript tagging is perhaps the favorite child of the industry at the moment. Most vendors and web analytics solutions are relying on JavaScript tagging to collect data.

After the beaconing season, JavaScript tagging allowed for more data to be collected more accurately and—very important—it ushered in new business models in the industry. Data serving was separated from data capture, hence reducing the reliance on corporate IT departments for various data capture requests. It also meant that data capture moved to third-party web analytics vendors in most cases. Now web pages could go out from the company servers, with no need to capture data, and be presented to website visitors. In turn, data about the visitor session would be captured on other servers (usually, third-party—for example, your favorite web analytics vendor's servers) and be processed there with reporting available online. No longer was there a need for companies to host their own infrastructure to collect data, a team to process the data, and systems to support reporting. Of course, nothing in life is perfect, and this rose has its own set of thorns. But let's first understand how tagging works (see Figure 2.4). Figure 2.4 How data capture works with JavaScript tagging



The process is as follows:

1. The customer types your URL in a browser.

2. The request comes to one of your web servers.

3. The web server sends back the page along with a snippet of JavaScript code appended to the page.

4. As the page loads, it executes the JavaScript code, which captures the page view, details about the visitor session, and cookies, and sends it back to the data collection server.

5. In some cases, upon receipt of the first set of data, the server sends back additional code to the browser to set additional cookies or collect more data.

Benefits of Using JavaScript Tagging as Your Data Collection Mechanism

- Other than web logs, this methodology perhaps has the easiest initial implementation effort. Adding a standard few lines of JavaScript code in a global site element (such as a footer) can instantly tag the entire site, and you can have massive amounts of data and standard reports 30 minutes later.

- If you don't have access to your web servers (technically) and/or your web server logs, JavaScript tagging is your only choice. You can install the tags easily (in your pages) and use an ASP vendor to do your reporting. This is particularly appealing to small- and medium-sized businesses.
- Page caching, either locally on a visitor PC or on cache farms such as those of Akamai Technologies, is not a problem for JavaScript tagging (as it is for web logs). Regardless of where your web page is being served from, the JavaScript tag will execute and your analytics tool will be able to collect the data.
- You have a great deal of control over exactly what data is collected. You also have the ability to implement custom tags on special pages (cart, checkout, order confirmation, knowledge base articles) that allow you to capture additional data for those pages (for example, order value, quantity, product names, and so forth).
- JavaScript enables you to separate data capture from data serving. When you use JavaScript tagging, your site releases will be a little bit faster because your IT department does not have to check anything relating to data capture, other than ensuring that your tag is on the page. (Data capture becomes your vendor's responsibility.) You don't have to trouble your IT department to set cookies to track sessions; your tool can do it now.
- Most vendors' innovation (new features, upgrades to how data is captured, and so forth) is happening in the JavaScript methodology. Most vendors stopped actively improving their web log tool versions. Many don't even offer a web log version of their tool.
- If you use third-party cookies (set by you or, as is usually the case, your vendor), tracking users across multiple domains becomes easier, because your third-party cookie and its identifying elements stay consistent as visitors go across multiple domains where your JavaScript tags exist.

Concerns about Using JavaScript Tagging as Your Primary Data Collection Mechanism

- Not all website visitors have JavaScript turned on, often for privacy or other reasons. For these users, your analytics platform will not collect any data. Benchmarks are hard to come by, but usually 2–6 percent of site visitors have JavaScript turned off. These visitors will be invisible to you.
- Data collected via JavaScript tagging is divorced from other metadata. Hence it is almost imperative that lots of thought and planning be put into creating the tag that will be capturing the site taxonomy and hierarchy, to allow for optimal analysis. This can be a strenuous process and requires regular maintenance as the site evolves.

- JavaScript tags collect data "browser side" and not "server side." Some websites, rather than storing some data in cookies or URL parameters, will store data on the servers during the visitor session. In this case, the tags will not capture essential data. If your IT strategy is to hold key data on the server, rather than on the browser/visitor machine, tags might not work for you (or you will have to go through the pain of changing your IT strategy).
- Capturing data in JavaScript tags about downloads (for example, PDFs or EXEs) and redirects is harder than with web logs, though some vendors are thinking of clever solutions.
- If your website is already JavaScript heavy, with lots of JavaScript on the site trying to do lots of clever things, your web analytics JavaScript tag could cause conflicts. In some cases, using tags to collect data might not even be possible (to allow your website to function).

Packet Sniffing

Packet sniffing technically is one of the most sophisticated ways of collecting web data. It has also been around for quite some time, but for a number of reasons it is not quite as popular as the other options outlined in this chapter. Among the vendors who provide packet-sniffing web analytics solutions are Clickstream Technologies. Some interesting ways of leveraging packet sniffers are also emerging—for example, SiteSpect is using the technology for multivariate testing, eliminating the reliance on tagging your website to do testing. Figure 2.5 illustrates the process of collecting data using packet sniffing.

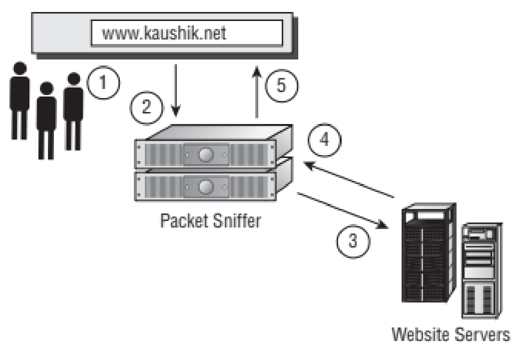


Figure 2.5 How data capture works with packet sniffing

There are a total of five steps to collect data:

1. The customer types your URL in a browser.

2. The request is routed to the web server, but before it gets there, it passes through a software- or hardware-based packet sniffer that collects attributes of the request that can send back more data about the Visitor to the packet sniffer.

3. The packet sniffer sends the request on to the web server.

4. The request is sent back to the customer but is first passed to the packet sniffer. The packet sniffer captures information about the page going back and stores that data. Some vendor packet-sniffing solutions append a JavaScript tag that can send back to the packet sniffer more data about the visitor.

5. The packet sniffer sends the page on to the visitor browser. A packet sniffer can be a layer of software that is installed on the web servers and runs “on top” of the web server data layer. Alternatively, it can be a physical piece of hardware that is hooked up in your data center, and all traffic is then routed to your web server via the packet sniffer solution.

Benefits of Using Packet Sniffers as Your Data Collection Mechanism

- Because all data passes through the packet sniffer, first it eliminates the need to use JavaScript tags for your website, or in theory, to touch your website at all.
- Your time to market is a bit longer than with JavaScript tagging because of the reliance on IT to approve and install the additional software and hardware in the data center—but it is less than the time required for other methods.
- You can collect lots and lots of data instantly, much more than with standard JavaScript tagging. For example, you can get server errors, bandwidth usage, and all the technical data as well as the page-related business data. It is fair to say that with packet sniffing you will collect the most comprehensive amount of data ever possible (every 0 and 1!).
- given the nature of the solutions, you do have the ability to always use the first party for cookies, and so forth.

Concerns about Using Packet Sniffers as Your Data Collection Mechanism

- For most companies, it is quite a struggle to make a case for and convince the IT department to add an additional layer of software on the web servers or to physically install hardware in their high-profile data centers and route all web traffic via these solutions. Some IT teams have a natural acceptance barrier for things they consider nonstandard. Packet sniffers also place a layer between the customer and the web page, a concept that while mostly benign can raise concerns and create hurdles.
- Remember, you are collecting raw packets of your Internet web server traffic.
- This poses two important challenges: First, nontrivial amounts of configuration work with your packet-sniffing solution to parse out just the needed data from all the raw data. The second challenge is privacy. In the raw data you will be capturing all the data, including PII data such as passwords, names, addresses, and credit card numbers. This last one, privacy, needs very careful stress testing and legal review.

But as you can imagine, using JavaScript tags to complement packet sniffers would expose you to some of the cons discussed earlier for tags.

- When using most packet-sniffing solutions, you would still need JavaScript tags to truly collect all the data that you will need for optimal analysis. For example, without JavaScript tags, the packet sniffer would not get any data for cached pages (because no request comes to the web server). Or consider your inability to collect data from Adobe Flash files or Ajax or rich Internet applications (RIAs): one deeply interactive file goes over to the visitor's browser, and then lots of interaction happens in the visitor browser that would all be invisible to a traditional packet sniffer (again, because the rich media interaction sends no request back to the server). Ditto for the inability to collect some of the core structure and metadata about pages via pure packet sniffer implementation.
- Packet sniffing can get expensive if you have many web servers (which is fairly common) or if you have web servers sitting on many networks. In these cases, you would have to install the software or the hardware on all the networks.

Outcomes Data

Regardless of what methodology you use from those listed earlier in this chapter, and no matter what web analytics tool you use, most of the data that you obtain will be based on customer clicks. You will, in the very short duration of one day, have access to between 80 and 300 reports (depending on the tool you end up using), all illustrating a combination of these facets of your website traffic:

- Visitors (visits, total, unique)
- Page views (individual, aggregates, averages)
- Time (overall, averages, slices)
- Referrers (counts, websites, keywords, trends)

Every report you see will illustrate one of the preceding metrics, or one of them multiplied by or divided by or subtracted from the other. The important question that is often forgotten in all this deluge of data is, so what happened? If all these people came and saw all these pages and spent so much time on the site, what was the outcome for the customer or the company? This is why it is extremely important to think really hard about your outcomes data strategy—which should begin with the question, why does your website exist? The answer to that is not a 500-page tome, but rather a few words that get to the heart of the site's existence. The answer could sound something like this:

- To make as much money as possible for our company, without causing undue harm to our customers

- To reduce the cost of servicing a support phone call by improving web self-help
- To improve the problem resolution rates for our most in-need customers
- To generate leads for our email database or sales prospecting efforts or future products not yet announced
- To create a customer experience that would reinforce our brand in the hearts and minds of a million people who don't know any better

After you have an answer to the question of why your site exists, it is imperative to investigate how your decision-making platform will capture the data that will help you understand the outcomes and whether your website is successful beyond simply attracting traffic and serving up pages.

E-commerce

For most e-commerce websites, it is now fairly standard practice to use custom JavaScript tags (or packet sniffers or web beacons) to capture data from the order confirmation page. The data you will capture most frequently via this custom implementation is as follows:

- Order's unique identifier
- Product or service ordered
- Quantity and price of each item
- Discounts and promotions applied
- Metadata about the customer session: A/B or multivariate test (MVT) IDs, cookie values, and so forth
- Metadata about the products or services: product hierarchy, campaign hierarchy, product attributes (all for sophisticated post-data-capture analysis)

This data is then incorporated into your standard clickstream tool, enabling you to report on this e-commerce data.

Lead Generation

For lead generation websites, you might be able to collect data on your website (on the "thank you" page, the one that the customer sees after submitting a successful lead), or you might have to partner with other websites that might be collecting and storing the leads on your behalf. Plan on identifying where the data is being captured and how you can have access to it (for example, via JavaScript tags or beacons or database exports).

Brand/Advocacy and Support

For both brand/advocacy websites and for support websites, the outcomes are less clear. A page view on these sites is nothing more than a viewed page. For the longest time, we have made the leap of faith that if the user sees a certain page, we can call it mission accomplished—mostly because we did not know any better. In these cases, we need to know the customer's perception or to ask the customer whether the page view resulted in their problem being solved. Outcomes in this case are harder to figure out. However, a great way to start is to put a surveying methodology in place on the websites that would continually ask a series of relevant questions to a statistically significant sample of site visitors to get their ratings on success. This can be a great complement to your clickstream data. An important facet of outcomes analysis is the ability to take key data out of a customer interaction that may or may not be available to a clickstream web analytics tool. We have all come to realize this, and so have the web analytics vendors. Increasingly, a cornerstone of an optimal web analytics strategy is having a data warehouse environment that allows you to have more-complex data in an easy-to-report-from environment. Data warehouses are typically very flexible when it comes to importing data from external sources and hence empowering you to do the required analysis with a view into the end-to-end customer experience. Vendors such as Omniture and WebTrends have created a V1 version of a true "data warehouse back end" in their tools. Alternatively, many large companies are choosing to build their own data warehouse environments (Figure 2.6) in which clickstream is just one data source. These companies have the ability to use standard software, methodologies, and business intelligence (BI) tools (such as Brio, Business Objects, Cognos, and MicroStrategy) to slice and dice the data. Having your own environment means that you have immense flexibility in terms of bringing in lots more sources of data (for example, event logs from your Flash or rich Internet applications, Google search data, metadata from other parts of the company, and CRM or phone channel data). This allows you to truly create an end-to-end (e2e) view of customer behavior and outcomes that can scale effectively over time. You can also use standard off-the-shelf tools, which can be a benefit.

Research Data

The core goal of measuring the why, or qualitative analysis, is to understand the rationale behind the metrics and trends that we see and to actively incorporate the voice of the customer (VOC) into our decision making. The following user-centric design (UCD) and human-computer interaction (HCI) methodologies are commonly used to understand the customer perspective:

- Surveys
- Heuristic evaluations
- Usability testing (lab and remote)
- Site visits (or follow-me-homes)

Although not a traditional UCD/HCI methodology, experimentation and testing (A/B, multivariate, or experience) increasingly falls into the research data category simply because it can often be the fastest way to answer a hypothesis we might form from quantitative analyses and get the customer perspective to validate or disprove that hypothesis.

There are three aspects related to research data that need to be strategically planned when it comes to research data: mindset, organizational structure, and timing.

1. Mindset

Our industry, traditionally called web analytics, is seeped in clickstream and outcomes analysis. We believe that analysis of numbers will provide all the answers. There is a huge mindset challenge to convince all the “quant jocks” (up and down the management chain) to see the value of qualitative data, and perhaps most important, to start to think differently because analysis of research data poses completely different opportunities and challenges.

The recommended approach is to first internalize the value of qualitative data yourself. Then prove the value by actually doing the research and sharing findings and do that a few times.

2. Organizational Structure

Most frequently the research team is either in a separate organization (closer to traditional market research or user research), or it is outsourced (typically to an agency or consulting company), or it simply does not exist. The action that needs to be taken in the last scenario is clear: create one. But in the other two it is less clear. My recommendation is to have a web research team and to have the team members, researchers, sit with and work alongside the web analysis team. The rationale for this is that each of the two pieces separately add up to one and a half, but when they are together, it is a case of one plus one equals four. The qualitative team can benefit from the insights that the quantitative team can provide in order to focus their efforts and have a real-world connection to what is going on (at times researchers can get “disconnected” from the real world). The quantitative team can benefit from getting much closer to the customer and their perspective than was ever possible (no matter how much they use solutions from Omniture or WebTrends or Visual Sciences). You will have to plan for having such an outcome-maximizing structure, sell it to your decision makers, and staff the team with the right leadership.

3. Timing

It is always hard for key company decision makers to know when is the best time to engage in research studies in order to maximize the learning potential. Some companies will conduct research on an ad hoc basis, others will do it prior to a big launch, and others when they find something interesting in the quantitative data. I recommend that you have at least one continuous listening program in place. Usually the most effective mechanism for continuous listening, benchmarking, and trending is surveying. It is extremely cost-effective to use one of the many industry providers (such as ForeSee Results or iPerceptions) to implement surveying that is geared toward teasing out critical facts—such as why people come to your website, what their task completion rate is, whether they are satisfied, whether they are more or less likely to buy as a result of their website visit, what they would like fixed, and more.

The other methodologies fall into the noncontinuous listening categories (that is, they are deployed on an as-needed or periodic basis). You would use the best-fit methodology based on the following:

- Scope (both size and complexity) of the problem you are trying to solve (entire website, core segments of experience, particular pages, and so forth)
- Timing (whether you need it overnight or over the next few weeks)
- Number of participants (how many customers you would like feedback from)
- Business desire (before and after a site launch, at some random time, or based on external triggers, for example) Although each methodology is unique, typically costs go up from heuristic evaluation to lab usability to site visits, and the opportunities for learning about your customers also increase in that order.

Competitive Data

The last data collection/analysis methodology is perhaps one of the most effective ways of gaining a strategic advantage. It is pretty easy to celebrate success (or acknowledge failure) of our websites based on just the metrics from our web analytics tools (Google Analytics, ClickTracks, IndexTools, HBX, or others). But if our visitors are doing better year after year, is that great? Or if your return on investment (ROI) is up for your pay per click (PPC) campaigns, is that fantastic? Or if your revenue has increased 30 percent from last month, is that success? In each of these scenarios, the answer could be a whopping yes. But it is missing the critical ecosystem context: what is happening in the landscape that could have caused these outcomes vs. what you are causing?

It could be that visitors on the Web went up 70 percent in your category. Or perhaps your ROI increased because your competitor stopped their campaigns. Or your competitor's revenue increased 80 percent because of all those new visitors, but you could manage only 30 percent. True delight comes from knowing how you are doing vis-a-vis your competitors or the industry as a whole. This competitive intelligence is key to helping you understand your performance in the context of the greater web ecosystem and allows you to better understand whether a certain result is caused by ecosystem trends or your actions (or lack thereof). Having a focused competitive intelligence program (which can be all of half a person in reality) can help you exploit market trends, build off the success of your competitors, or help optimize your search engine marketing program because you know exactly what your competitor is doing.

There are three main methodologies used to collect data that is then analyzed for competitive intelligence on the Web: **panel-based measurement, ISP-based measurement, and search engine data.**

Panel-Based Measurement

Panel-based measurement is very much inspired by traditional television Nielsen ratings systems, whereby in exchange for an incentive, the participant agrees to have their TV viewing behavior tracked. In that case, the individual tracks the programs, but for the Web this part can be automated.

A company called comScore NetWorks uses panel-based measurement to compile data that is used by many companies for competitive analysis. In exchange for an incentive (server-based virus protection or sweepstakes prizes), the panel member agrees to have all their web surfing activity monitored. comScore accomplishes this by installing monitoring software on the panel member’s computer and then funneling 100 percent of the surfing via comScore’s proxy servers. All the customer data (HTTP, HTTPS, PII, credit card, social security numbers, and so forth) are captured by comScore.

As of December 2006, comScore’s global network is 2 million panelists (though the Media Matrix audience measurement is 120,000 US panelists, and Media Matrix Global services is 500,000 outside the United States).

Benefits of Using comScore (Panel-Based Measurement)

- comScore has in-depth browsing behavior data from their panel and hence the analysis they provide can go really deep in terms of website data.
- comScore can provide metrics such as conversion rates or purchasers. They monitor 100 percent of the traffic for their panel and they apply aggregations and sophisticated computations to approximate these metrics.
- comScore can break down some websites into deep embedded pages; for example, they can measure microsoft.com/office/avinash/information.html, a potential page that could be embedded deep into a directory and can’t be tracked effectively by other methodologies.
- comScore can do more-custom work on your behalf because they have every page, HTTP or HTTPS, from their panel members and all the data associated with that panel member and their surfing habits.

Concerns about Using comScore (Panel-Based) Data

- There are approximately 200 million people on the Web in the United States and approximately another 700 million worldwide. Sample size could be a concern with comScore, since you are extrapolating the behavior of 200 million people from that of several hundred thousand.

- comScore offers incentives (virus protection, sweepstakes) that might appeal to a certain type of the Internet population, so sample bias could be an issue.
- Because comScore monitoring software is considered invasive (sometimes spyware) by corporations, it can't be installed in a work environment. This also inserts additional sample bias by not accounting for work surfing, which by many estimates accounts for more web usage (even for personal purposes) than home web usage.
- With most panel-based measurement systems, a small percentage of web users represent the general population. This means that for the most part a website with huge amounts of traffic might have reasonable numbers represented by the panel, whereas websites will smaller traffic (say less than a million visitors a month) would not be represented by accurate numbers.

ISP-Based Measurement

The second method of collecting data for competitive analysis uses anonymous data that is captured by various Internet Service Providers (ISPs). While all of us surf, all our data is funneling through the ISPs that we all use to connect to the Internet. Companies such as Hitwise have agreements with ISPs worldwide whereby the ISPs share the anonymous web log data collected on the ISP network with Hitwise. This data is analyzed by Hitwise. The data is further combined with a worldwide opt-in panel to produce demographic and lifestyle information.

According to Hitwise, the company has roughly 10 million US and 25 million worldwide users who provide data (as of December 2006).

Benefits of Using Hitwise (ISP-Based Measurement)

- The sample size is much bigger; this is a huge benefit.
- The basic data capture mechanism means they have a much more diverse pool of people providing data. Participants don't have to agree to have their anonymous data analyzed, and there is a higher likelihood that the ISP-based measurement system covers a significantly more diverse set of surfing habits and a more diverse population.
- ISP-based methodologies such as those used by Hitwise also have much deeper and richer search engine traffic data.
- The psychographic (demographic, lifestyle) data Hitwise provides via the Prizm database is significantly better than self-reporting of such data.
- Hitwise has a lot more on-demand reporting available through their web access interface, which is very much amenable to self-service.

Concerns about Using Hitwise (ISP-Based Measurement)

- Hitwise data offers more breadth (websites, participants, search engines, and networks), but it does not go too deep into an individual website (deep pages on the website, insights into HTTPS sessions, and so forth). Hence it can't, yet, provide rich insights into behavior deep into a website.
- Conversion rate–type metrics are best obtained from services that use panel based methodologies (such as comScore) because they capture every click for the sites the users on their panel use.
- There are certain types of PII data (payment types used or types of credit cards, for example) that can't be obtained from ISP-based methodologies.

Search Engine Data

This is the newest kid on the blog and perhaps the most underutilized source of information about competitive behavior. As you can imagine, search engines collect massive amounts of data related to searches. They also often know information about their users (in the case of MSN, via the account information they have for Hotmail email or Passport / Windows Live ID login systems). Google and MSN have recently opened up lab/beta environments where they enable users to run queries against their databases to glean insights into competitive information. On Google Trends (www.google.com/trends), you can enter one or more search phrases and Google will indicate the total number of searches done over time for those key phrases, the number of times the phrases have appeared in stories in Google News (it also shows the news story), and the top regions, cities, and languages for the search phrases you typed in. Figure 2.7 shows a Google Trends report. On Microsoft adCenter Labs (adlab.msn.com/demo.aspx), you are able to do even better. You can predict any website user's age, gender, and other demographic information; do tons of analysis related to search keywords, such as keyword clustering, keyword forecast, search funnels (what people search for before and after they search for your keywords), and keywords expansion; and detect the commercial intent of visitors for any website (for example, are visitors to your website or your competitor's website more likely to buy?).

Benefits of Using Search Engine Data

- Access to the data and analysis is completely free.
- Because most web surfers tend to use search engines, this data represents a huge number of web users.
- Search engines are a great source of detailed data related to keyword behavior (adCenter especially has some amazing tools you can leverage).

Concerns about Using Search Engine Data

- The amount of analysis you can do is limited and not close to what Hitwise or comScore can offer.
- The tools are still maturing and in a beta phase.