WHAT MARKETERS REALLY WANT TO KNOW

TRACKING USERS

BY DAN R. GREENING

Marketers don't want to measure raw hits on a Web site. Increasingly, they want to categorize visitors and measure the significant things those visitors do on the site. Many want to track the effectiveness of promotions in real time and make adjustments instantly. They often want to take data about Web activity offline, combine it with their traditional data, mine it, and report on it. They want to improve advertising effectiveness, visitor loyalty, purchase rates, cross-sells, and up-sells. All this is fueling demand for a new generation of Web-site analysis tools that represent visitor behavior in terms that marketers understand.

Web logging was originally designed for Web engineers to diagnose problems and measure total throughput, not to provide insights that could improve the marketing performance of a site. If the data the marketer wants is logged at all, it has to be mapped to a readable form. But the most useful marketing data isn't recorded at all.

For example, the common log format (CLF) written by all Web servers contains only the fields shown in Table 1. This isn't enough information to identify the referring site, track cookie-connected sessions, or identify page-views (rather than file "hits"). Web-server manufacturers have added mechanisms to log referrer data, but this data is stored differently on the various servers.

The extended common log-file format (ECLF) includes referring pages and cookie identification, but still fails to provide important content and visitor information. In particular, high-level marketing demographics, dynamic content subjects, advertising effectiveness, and revenues don't make it to Web-log reports.

Because much of the information a marketer needs is not available in logs, specialized monitors have emerged to observe higher-level events and deliver more relevant data. Monitors can track events in Web servers, network interfaces, or e-commerce engines. We'll discuss each type of monitor, then show how the highest-level events can be used to improve marketing on e-commerce sites. We'll talk about category and cluster modeling, two different techniques that traffic analyzers use to segment visitors. Finally, we'll outline the key steps to use to decide which traffic analyzers are appropriate for you.

Server Monitors

A server monitor typically runs as a plug-in to NSAPI, ISAPI, or Apache, getting information about each event through an API. Server APIs are proprietary, so the events and data seen by server monitors depend on the Web server release. In most cases, a server monitor can gets unique visitor IDs, referrer pages, and all the information in Table 1.

Some data isn't available to server monitors. For example, when a visitor interrupts the transmission of a page

FIELD	CONTENTS
remotehost	Browser hostname (IP number if DNS hostname is unavailable)
rfc931	The remote log name of the user (almost always "-" meaning "unknown")
authuser	The Web server authenticated username
date	Date and time of the request
"request"	The request line exactly as it came from the client
status	The HTTP status code returned to the client
bytes	The content-length of the document transferred

Table 1: Common log file (CLF) format fields and their contents.

```
<title "MovieCritic: The Guns of Navarone">
<meta action,drama>
<!--aria add-content-category action -->
<!--aria add-visitor-category soccer-mom -->
<!--aria add-persona lastvisit="3/22/99" -->
```

Example 1: Andromedia Aria dynamic-tracking tags.

Ad click

Promotion view

Promotion click

Product navigation

Detailed product view

Shopping-cart insert

Shopping-cart delete

Upsell view

Upsell click

Shopping-cart checkout

Example 2: Marketing events.

by hitting the stop button, clicking on a Web shortcut, or typing in a new URL, this has the effect of a "stop request" being sent to the Web server. The Web server then interrupts the transmission of the page being sent. Interrupted transmissions are very informative events: They may indicate that a particular resource is taking too long to generate, or that the whole Web server is overloaded. Unfortunately, Web servers typically don't notify plug-ins or log when transmissions end prematurely.

Server-side page generation from dynamic content servers, such as BroadVision One-to-One, Allaire Cold Fusion, or Vignette StoryServer, can make it impossible to reliably categorize the content being delivered from any specific URL. The only portable process that works across these systems is to actually look at the generated content, searching for specific HTML tags. Unfortunately, many Web servers fail to provide an interface that lets plug-ins easily view content during delivery.

Installing a server monitor introduces a slight risk to the Web server: If something goes wrong in the monitor, it could crash the Web server. A server monitor that directly calls a database server introduces a higher risk because more code is involved. If the Web site is mission-critical and requires a server monitor, use an architecture in which the monitoring and recording processes are separated. Some Web servers can isolate a server monitor in a separate process, which significantly reduces the risk.

Network Monitors

Network monitors perform "packet sniffing." All major operating systems allow an application to register a function called to view each packet when it crosses the wire. Because traditional Ethernet LANs broadcast every packet to all computers attached to the same subnet, a single network monitor could report on every HTTP event on that subnet.

The best approach, however, is to install a network monitor on each Web server. Network interfaces give you a lousy choice: Sniff packets from a single machine, or sniff all packets on the wire ("promiscuous mode"). When network monitors operate in promiscuous mode, they consume much more processor time. High-traffic sites often segment their networks in a way that makes it impossible to see packets from every Web server, anyway. In this case, the only solution is to install a network monitor on each Web server machine.

A network monitor can see everything, including client requests, server responses, cookies, and HTML files. It can track stop requests issued from the browser, making it possible to list the pages that are taking too long to generate. It can measure the Web server's response time to different requests. Certain network monitors can report on content-related HTML tags, such as `<TITLE>`, `<META>` or comment tags. They can even capture "form data" transmitted via a POST request when the visitor hits a submit button.

Example 1 shows different values that can be set and tracked through Andromedia's Aria network monitor. When configured to track `<TITLE>` tags, the traffic analyzer can report on pages using the same reference point delivered to visitors in the browser's title-bar. When configured to track `<META>` tags, the traffic analyzer reports on pages using the content categories provided to search engines.

Analyzer-specific comment tags deliver the greatest flexibility, because these tags have no special meaning to other applications. When configured to track comment tags, the traffic analyzer reports on the categories that a marketer wants to track. Furthermore, the tags can direct the analyzer to perform special functions, such as setting the visitor ID, changing demographic values in the visitor profile, or setting a product attribute.

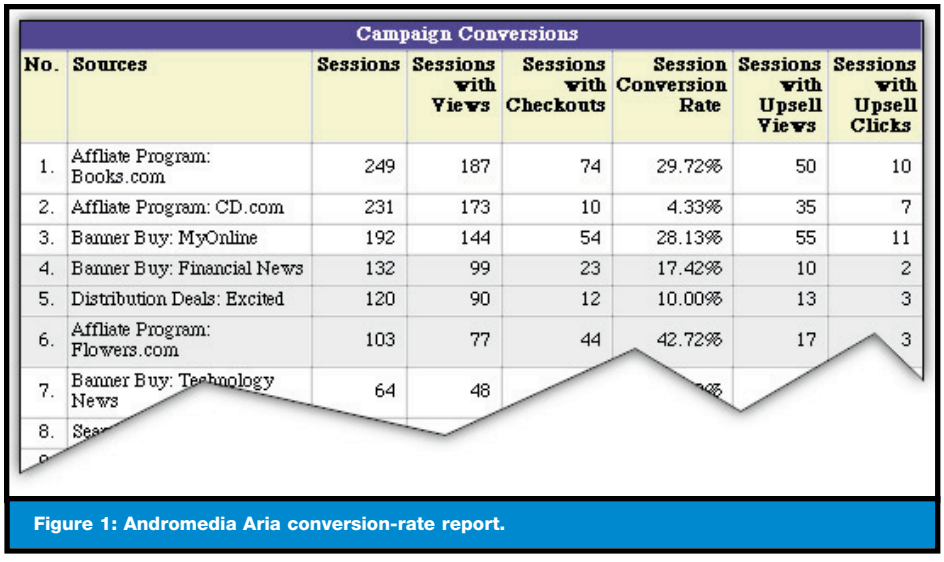
This eliminates the middleman, allowing dynamic content servers to generate dynamic tracking tags automatically, bypassing the architectural limitations of the Web server. For example, by adding a subclass to the Vignette StoryServer profile-mark function that inserts Aria-specific comments, Web developers can use Aria to report on visitation patterns using the categories already declared in their StoryServer-driven sites.

Network monitors significantly reduce risks to Web server operation. A network monitor can be placed on its own machine to isolate the traffic analyzer from the Web server. If the network monitor crashes, the Web server continues to run properly. Even when they're running on the same machine the risk is extremely low, because the network monitor is a separate process operating independently from Web-server processes.

There is one drawback to network monitors: They can't track encrypted information from secure Web servers. Because of the many advantages of network monitors, high-end Web sites usually install server monitors only on secure Web servers, and install network monitors elsewhere.

E-Commerce Application Monitors

A needs-driven approach to network-traffic analysis starts with the goal and works back to the solution: E-commerce marketers want to make more money. Therefore, the best marketing reports reveal where the money comes from, who the money comes from, and what marketers can do to improve their revenues. Marketers can use this information to increase advertisements on sites that reach the most interested parties, provide a better selection of products for



Campaign Conversions

No.	Sources	Sessions	Sessions with Views	Sessions with Checkouts	Session Conversion Rate	Sessions with Upsell Views	Sessions with Upsell Clicks
1.	Affliate Program: Books.com	249	187	74	29.72%	50	10
2.	Affliate Program: CD.com	231	173	10	4.33%	35	7
3.	Banner Buy: MyOnline	192	144	54	28.13%	55	11
4.	Banner Buy: Financial News	132	99	23	17.42%	10	2
5.	Distribution Deals: Excited	120	90	12	10.00%	13	3
6.	Affliate Program: Flowers.com	103	77	44	42.72%	17	3
7.	Banner Buy: Technology News	64	48				
8.	Sea						

Figure 1: Andromedia Aria conversion-rate report.

different types of visitors, or offer better service to their most valuable visitors.

To provide this information, e-commerce monitors look at visitors' interactions with an e-commerce site—specifically their interactions with marketing elements on the site, such as advertisements, promotions, detailed product views, purchases, and shopping-cart interactions. E-commerce monitors observe these marketing events by hooking into an e-commerce engine, such as BroadVision One-To-One or Open Market Transact. At present, these interfaces are highly proprietary.

Some examples of marketing events are shown in Example 2. The most significant subset of events follows a visitor's interactions with a stock-keeping unit (SKU)—a site-specific number that merchandisers use to refer to a specific product—and a visitor ID.

Suppose Sally clicks on an advertisement and goes to www.suzeshoes.com. When Sally arrives on the site, the e-commerce engine might generate an "ad click" marketing event, with a reference to the ad campaign and referring site. This click-through behavior may be the first important event in a series of steps that lead to a purchase, so e-commerce analyzers record and report on it.

Suppose www.suzeshoes.com contains an onsite advertisement for black canvas shoes. This generates a "promotion view" event referring to the black-canvas-shoe SKU. If Sally clicks on the promotion, a "promotion click" event with the same SKU is generated. Now that Sally is looking at the promotion page for the shoe, she might click to see a review or specification list; this generates a "detailed product view" event. Finally, she decides the shoes are cool, and clicks to add the shoes to her shopping cart, generating a "shopping-cart insert" event.

On the shopping-cart page, she might see another promotion asking whether she'd rather buy some higher-priced black leather shoes. This is called an "upsell" by marketers, and generates an "upsell view" event with the original and upsell SKU numbers. If Sally clicked on the upsell, the e-commerce engine would generate an "upsell click" event, but let's suppose she clicks the "checkout" button instead. Finally, she enters her credit card and submits her order; this generates a "shopping-cart checkout" event.

The importance of e-commerce reporting is illustrated in Figure 1, a "Campaign Conversions" report for a fictional BroadVision One-to-One-driven online store. Marketers use the term "conversion rate" to refer to the ratio of marketing successes (purchases) to marketing attempts (banners displayed).

The "Sources" column contains the name of the ad campaign driving the visitor to the site. The "Sessions" column shows the number of visitors directed from a particular ad campaign. "Sessions with Views" indicates the number of visitors clicking to see detailed information on a product. "Sessions with Checkouts" is typically a marketer's ultimate goal: driving online purchases. In this example, "Session Conversion Rate" means the ratio of checkouts to sessions.

What does this report tell us? Our affiliate program with Flowers.com drove the highest conversion rate. When we get

Online

Accrue Insight
www.accrue.com

Allaire Cold Fusion
allaire.com

Analog
www.statslab.cam.ac.uk/~sret1/analog

Andromedia Aria
www.andromedia.com

BroadVision One-To-One
www.BroadVision.com

Marketwave Hit List
www.marketwave.com

Microsoft SiteServer
www.microsoft.com/siteserver

Net.Genesis Net.Analysis
www.netgen.com

Open Market Transact
www.openmarket.com

Personify Essentials
www.personify.com

Vignette StoryServer
vignette.com

Webstats
www.bbs.com.au/webstats.htm

WebTrends Professional
www.webtrends.com/default.htm

Wusage
www.boutell.com/wusage

WHAT MARKETERS ASK	WHAT MARKETERS MEAN
Who visited?	Visitor categories (demographic or behavioral) sorted by visit frequency
Where did they come from?	Ad campaigns or inbound hyperlinks sorted by visit frequency
What did they do?	Content category, for each visitor category, sorted by page-view frequency
How did they use the site?	Traffic patterns, next-click or previous-click from each page, sorted by frequency
Why did they leave?	Exit pages, for each visitor category, sorted by visit frequency

Table 2: Information that marketers need to know about Web sites, translated into categories.

Flowers.com buyers to visit our site, more than 42 percent of them decide to buy. And despite the fact that CD.com brought more visitors to the site than Flowers.com, in fact Flowers.com generated more conversions. It would be wise for our company to deepen its relationship with Flowers.com.

Modeling Visitor Behavior

Now that we've monitored our events, it's time to figure out what happened—that is, construct a statistical model of visitor behavior. All traffic modelers share the same basic goal: They want to present a concise report on visitor behavior that shows who visited, where they came from, what they did, how they used the site, and why they left.

Category Modelers aggregate behaviors based on preidentified categories. Categories are important, because they help avoid drowning our marketing friends in details. Furthermore, marketers have some favorite categories, such as demographics and psychographics, that they can use to relate Web behavior to other marketing media.

Some of us technologists have been known to respond to "Who visited?" with a huge file containing all the visitor profiles. Then when marketers protest, we've said, "Don't you know how to mine data?" To help you avoid such faux pas (and the resulting hurled keyboard), I've provided a helpful translation.

The simple questions in Table 2 can be extended in different ways to form a use case, which generates a schema for reporting usage behavior. There are visitors, visitor categories, content, content categories, and relationships between them. In the most detailed view of "what happened," the relationships between visitors and content are represented by raw events, each an n-tuple: *(visitor, content, action, time)*.

From those raw events, the modeler can generate statistical relationships. Some relationships are obvious, such as *(content-category, visitor-category, count, time-span)*, which reports the most frequent visitors to a set of Web pages by demographic classification. Some relationships are less obvious, such as *(page, next-page, count, time-span)*, which reports next-click and previous-click statistics from each page.

A small combinatorial explosion occurs if you try to generate all the possible relationships, so all Web-traffic analyzers limit which relationships can be generated. To provide flexibility to SQL-savvy marketers, many traffic analyzers provide for data export.

Cluster Modelers turn the modeling phase upside down. Instead of asking the marketer to identify visitor categories up front, cluster modelers create a behavior profile for each visitor indexed by content-category, then try to find "clusters" of visitors whose profiles look similar. The center of each cluster becomes a "centroid" or "beacon" profile. A marketer typically then gives each centroid a name, such as "impulse buyer" or "loyal hardware hobbyist." These centroid profiles are then saved for use in categorization.

As new visitors use the site, they are assigned the nearest centroid profile. This lets a cluster modeler report that "impulse buyers" are looking at certain parts of the Web site, or that "loyal hardware hobbyists" are arriving more frequently from a particular ad campaign.

Over time, the centroids drift from the arbitrary labels assigned by a marketer due to new types of visitors or to newly added content. What was once a "loyal hardware hobbyist" centroid may later include mostly "gun-toting survivalists." To avoid mislabeled clusters, when new content is added to the site or when new ad campaigns are run, marketers must wait several days to allow new visitors to interact with the new content, then must recluster the behaviors and relabel the centroids. In short, to get decent results, cluster modelers require more marketing insight and effort than category modelers require. There's a delay required to gather enough data to "train" a cluster modeler.

Clustering algorithms form an academic industry of their own, variously called data mining, operations research, or combinatorial optimization. Because clustering can be slow, cluster modelers work best on sites with few content categories. As traffic increases, sampling is used to reduce computation time. Cluster modelers are only as good as the algorithms they use and the relevancy of the behaviors they measure. Unfortunately, those details are often hidden behind reports that use arbitrarily assigned, possibly stale labels.

As cluster modelers mature, category modeling and cluster modeling will likely be used side-by-side in Web marketing, just as demographics and data mining are now both firmly entrenched in direct-mail marketing. However, cluster modeling may be more useful offered as a service than in the form of software, because human decisions made at the Web site—such as data categorization and centroid labeling—largely determine the success of a cluster modeler. Evidence for this trend is appearing, with some portal companies acquiring cluster modelers to provide targeting services, such as Yahoo's acquisition of Hyperparallel, while category modelers have remained successful independent software vendors.

High-Traffic Recording

On popular Web sites, traffic is increasing exponentially. Traditional Web-log analysis can take too long to read and process large log files. Even recording the raw data in a database is too slow.

Data-cube recorders don't actually store the raw event data at all. Instead, the recorder creates new visitor and content categories on the fly, and assembles a statistical model of visitor behavior as event data flows in (see Figure 2). In OLAP lingo, the statistical model is called a complete or partial "data cube." It lets marketers rapidly roll-up and drill-down to see different views of the data. This is the method Andromedia Aria has adopted and it accounts for the product's unique analysis and reporting capabilities.

The advantage of data-cube recorders is that reporting on preanalyzed data can be very fast. Furthermore, the statistical behavior model is typically much smaller than a collection of raw events, requiring less disk space. Finally, because a data-cube reporter writes less data and makes less frequent commits to the database, it can keep up with extremely popular sites where other recording techniques have difficulty.

The downside of data-cube recorders is that raw data isn't saved. If the recorder was not set up to generate the desired visitor or content categories automatically, it can be impossible to go back and regenerate the statistical behavior model after the fact.

To enable regeneration, Aria also provides a log recorder that creates compressed output files—optionally deleting files older than a preconfigured retention period—along with a log reader. Compressed-log recorders are inappropriate for most production traffic analysis, because decompression consumes precious processor time during the (also processor-intensive) analysis phase.

However, sites can run a data-cube recorder and a compressed-log recorder simultaneously, giving them the best of both worlds. The data-cube recorder provides on-the-fly data analysis for realtime reporting, while the compressed-log recorder lets a Webmaster restructure categories and regenerate the statistical model afterwards, if necessary. In practice, the combination is not used that often. Most Webmasters set up category-generation correctly in advance, and don't want to waste disk storage and processor time creating compressed log files. —DG

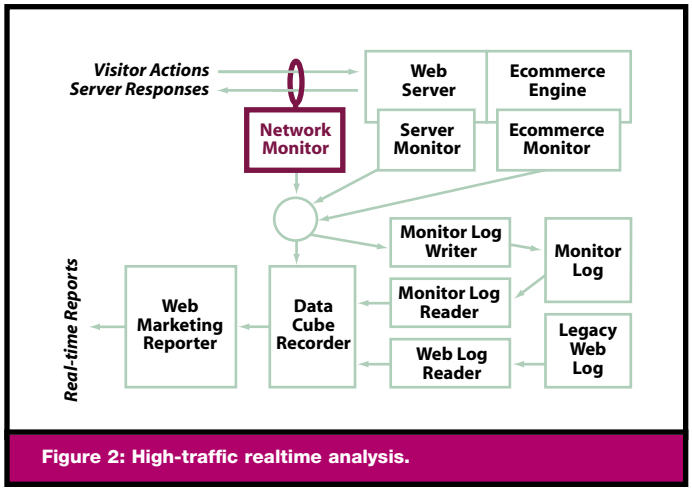


Figure 2: High-traffic realtime analysis.

Tracking Users

Key Decisions

Within the marketing world, there are many possible goals for traffic analysis, so deciding what to measure is especially important. You can compare ad campaigns to emphasize the most effective. (Does "effective" for you mean more click-throughs, more loyal visitors, or more purchases?) You can determine the most popular areas so that content can be improved. (How do you know your content is "improved"? Do you get more page views, purchases, or more people filling out an evaluation form?) You can get visitors to stay at your site longer by providing more next-clicks of strong interest, by reducing stop-requests, or by identifying "popular" exit pages and adding attractive hyperlinks to other parts of the site. (But is there a good marketing reason to simply keep people on your site?) All of these goals should be part of a marketing plan for your Web site.

There are numerous Web-traffic analyzers on the market (see "Online"). Your choice will depend on your needs and budget. The least expensive Web traffic analyzers use Web log files. They typically cost $5000 or less; some, like Analog, Wusage, and Webstats, are free. They're trivial to install and use, but the information they provide is limited. If you haven't determined any marketing goals, and your traffic is under 100,000 hits per day, an inexpensive or free Web-log analyzer can be a good place to start.

The most expensive Web-traffic analyzers provide both server and network monitors, and either database or data-cube recorders. (See the box titled "High-Traffic Recording" for a discussion of data-cube recorders.) They also have loosely coupled multiprocessing, multithreaded architectures. These analyzers can handle complex visitor and content categories, and track dynamic content generated from a variety of application servers. Database recorders can keep up with moderately high traffic, but may incur a few hours delay before generating reports on current traffic. Data-cube recorders generate traffic reports within minutes, and can be effectively deployed on highly trafficked sites.

High-end solutions are not for dabblers. The software and attendant configuration services are typically priced starting at $40,000 and can approach $500,000, especially if application or e-commerce monitors are included. Heavily trafficked sites often deploy the recorders, modelers, and reporters on separate enterprise-class multiprocessor hardware, adding $100,000 or more to the cost.

At the same time, the value of knowing who is visiting your site, which people are generating the most revenue, and what advertisements are working most effectively can often generate profits and savings far exceeding the cost of expensive analyzers.

Dan holds a Ph.D. in computer science from UCLA, emphasizing statistical optimization. He is currently vice president of advanced solutions at Andromedia. He can be reached at greening@andromedia.com.