

Intelligent Content Delivery and the Bottom Line: Smart Websites

By Joe McCormack

University of Advancing Technology

Development of Vtracker

A successful website is one that delivers the content a user expects to see without pushing content, such as the dozens of links, tickers and advertisements that most likely will overwhelm the user and may detract from the communication of the desired content. Large websites and website conglomerations (a network of websites owned by a single entity) can easily lose users through gigantic navigation themes, search results and a lack of structural uniformity between websites. I've found that the more options you give a user multiplies more ways you can lose that user.

As an active participant in rebuilding a portal website, I found that one of the most important objectives in this task was to be able to tie intelligent content delivery (i.e., delivering relevant content) to users, while incorporating many additional features such as these:

1. Remembering user surf patterns each time a website is visited, and maintaining surf histories for each user. This facilitates being able to tailor content for each individual user in real-time.
2. Remembering the user even if that user's IP address changes or if the user switches web browsers without heavy dependence on cookies throughout multiple sessions.
3. Being able to cast a user to a "group" or "genre" in order to readily determine content rolls and relevance to the website's purpose. This also allows for the delivery of focused navigation themes and content, thereby reducing site navigation complexity for the user.
4. Being able to accurately determine how many visitors visit a website, whether as a new visitor who has never been to the website before, or a visitor who has returned to the website several weeks later.

Project Vtracker was developed to meet the requirements of intelligent content delivery and also serve as a traditional statistical reporting application. The use of current statistical tracking software (i.e., a server-based application that tracks hits to web pages, browsers being used, etc.) at the inception of the project seemed to be a logical solution, in this instance described above. However, one of the problems with commercial tracking software, such as WebTrends, is that there are no mechanisms in place to dynamically influence website content delivery based on previous visit data on an individual user basis to individual users in real-time (this is one problem—I will discuss other significant and related issues as well).

The most challenging aspect of Vtracker, both during its inception and its later development, was envisioning how the four points above could be seamlessly integrated together into a robust

architecture by developing new, imaginative methods using current technology to derive a real-world solution to the deficiencies of other applications. In this case, I was able to rely on server-side technology only, without having to resort to installing any software on individual users' computers or becoming reliant on specific features of a handful of web browsers. Despite this, it took less than a month to move from inception to development and then completion of an application that would support the requirements. [Author's note: Although unseen, Vtracker is currently expanding, running quietly at <http://www.uat.edu/>.]

Vtracker Features

Vtracker—the “brain” of the portal site—can not only gather statistical data that is available with common tracking software, but can also directly control how a website serves content to each individual user; it also has the capacity to tailor such content delivery over time for each user as their habits change.

Because Vtracker is able to remember users across multiple sessions and maintain histories for each user (see Figure 1), website traffic and marketing effectiveness can be tracked with a greater degree of precision over traditional tracking applications.

VISITOR TRAFFIC / PROFILE ADMINISTRATOR [Main Page] [Help]			
USER DETAILS			
This section gives details relating to a visiting user of the selected domain independent of date so that the entire history of the user can be shown.			
User "Genre"	Last IP Used	Browser Used	Language
Game Design	217.237.95.XXX	mozilla/4.0 (compatible; msie 6.0; windows nt 4.0; t312461)	en-us
[Back Up] User Surfing Pattern On Site			
Date Hit	Web Page		
1/3/2005 10:57:21 AM	www.uat.edu/multimedia/gamedesign/default.asp		
1/3/2005 10:58:31 AM	www.uat.edu/faculty/subpages/credentials/default.asp		
1/3/2005 10:58:41 AM	www.uat.edu/faculty/subpages/bios/default.asp		
1/3/2005 10:59:57 AM	www.uat.edu/faculty/subpages/credentials/default.asp		
1/12/2005 2:42:44 PM	www.uat.edu/multimedia/gamedesign/default.asp		
1/12/2005 2:43:46 PM	www.uat.edu/careerservices/subpages/alumniintheindustry/default.asp		
1/12/2005 2:43:55 PM	www.uat.edu/careerservices/subpages/alumniintheindustry/subpages/featuredalumni/default.asp		
1/13/2005 8:15:52 AM	www.uat.edu/multimedia/gamedesign/default.asp		
1/13/2005 8:20:30 AM	www.uat.edu/faculty/subpages/credentials/default.asp		
1/13/2005 8:21:32 AM	www.uat.edu/faculty/subpages/bios/default.asp		
1/21/2005 9:36:54 AM	www.uat.edu/default.asp		
1/21/2005 9:36:59 AM	www.uat.edu/default.asp		
1/21/2005 9:38:24 AM	www.uat.edu/multimedia/gamedesign/default.asp		
1/21/2005 9:40:05 AM	www.uat.edu/multimedia/gamedesign/subpages/courses/default.asp		
1/21/2005 9:41:04 AM	www.uat.edu/multimedia/gamedesign/subpages/courses/default.asp		
1/21/2005 9:41:47 AM	www.uat.edu/multimedia/subpages/programcore/default.asp		
1/21/2005 9:42:25 AM	www.uat.edu/multimedia/default.asp		
1/21/2005 9:42:27 AM	www.uat.edu/multimedia/subpages/majors/subpages/gamedesign/default.asp		
1/21/2005 9:42:39 AM	www.uat.edu/subpages/programs/subpages/collegeofmultimedia/subpages/majors/subpages/gamedesign/subpages/courses/default.asp		
1/21/2005 9:42:53 AM	www.uat.edu/subpages/programs/subpages/collegeofmultimedia/subpages/majors/subpages/gamedesign/subpages/jobs/default.asp		
1/21/2005 9:45:38 AM	www.uat.edu/multimedia/gamedesign/subpages/resourcesfordiscipline/default.asp		
1/27/2005 1:13:17 PM	www.uat.edu/default.asp		
1/27/2005 1:13:39 PM	www.uat.edu/uatonline/degrees/gamedesign/default.asp		
1/27/2005 1:15:51 PM	www.uat.edu/uatonline/admissions/default.asp		

Figure 1

Example of a user's history across multiple sessions.

One of the reasons why Vtracker is more precise at delivering statistical tracking information (such as web page hits) is that it can distinguish between new and repeat users. Vtracker can remember exactly where and when a user does something on the website; this data is available under the section labeled “User Surfing Pattern On Site” (Figure 1). Vtracker easily remembers date of access, time and what page the user has hit, even across multiple sessions.

Maintaining histories is vital, particularly from a marketing perspective and from the perspective of those who monitor a company’s bottom line: the effectiveness of an advertising campaign can be more accurately determined by being able to separate new visitors (who may be visiting the website after seeing the ad) from visitors who are simply revisiting the site and may never have seen the ad. Vtracker also tracks referrer data in order to report referrals from other websites for online marketing campaigns.

Vtracker also categorizes visitors into interest groups automatically and can dynamically update categorizations in real-time as user interests change, as seen under “User Genre” (Figure 1). Because many advanced web development and maintenance issues are sensitive to the type of web browser the user employs, Vtracker remembers the web browser type. This, aside from assisting a web development team with development concerns, can also show web browser trends and market shifts. The same also holds true for the “Language” column (Figure 1); in this case, Vtracker has identified the user as an English speaker by “en-us”.

A side benefit of being able to remember and track users across multiple sessions is that a history of IP addresses the user has used can be referenced in the event of security issues that may arise (Figure 2).

1	www.uat.edu/multimedia/subpages/programcore/default.asp
1	www.uat.edu/multimedia/default.asp
1	www.uat.edu/multimedia/subpages/majors/subpages/gamedesign/default.asp
1	www.uat.edu/subpages/programs/subpages/collegeofmultimedia/subpages/majors/subpages/gamedesign/subpages/courses/default.asp
1	www.uat.edu/subpages/programs/subpages/collegeofmultimedia/subpages/majors/subpages/gamedesign/subpages/jobsgamedesign/default.asp
1	www.uat.edu/multimedia/gamedesign/subpages/resourcesfordiscipline/default.asp
1	www.uat.edu/uatonline/degrees/gamedesign/default.asp
1	www.uat.edu/uatonline/admissions/default.asp
1	www.uat.edu/admissions/subpages/admissionsrequirements/subpages/undergraduate/default.asp
1	www.uat.edu/admissions/subpages/admissionsrequirements/default.asp

[Back Up] Sites User Has Been Referred From	
Date Referred	Web Page Referred From
1/3/2005 10:57:21 AM	http://www.google.com/search?hl=en&q=%22game+design%22+%22degree%22
1/12/2005 2:42:44 PM	http://www.google.com/search?hl=en&q=%22degree%22+%22game+design%22
1/13/2005 8:15:52 AM	http://www.google.com/search?hl=en&q=degree+%22game+design%22
1/21/2005 9:36:54 AM	Visited site directly from typing in URL or from a bookmark or history in their web browser; OR the request came from a web bot.
1/27/2005 1:13:17 PM	http://www.google.com/search?hl=en&q=uat

[Back Up] Alias/User Categorization Information (to date)			
Assigned Alias	Group Affiliation	Site Association	Re-Associations To Site
UAT User	Internet	www.uat.edu	1

[Back Up] IP History	
Date of Use	IP Addresses Used
12/23/2004 7:18:09 AM	205.84.145.XXX
12/27/2004 4:39:25 PM	217.84.193.XXX
12/28/2004 1:58:13 PM	164.221.194.XXX
12/31/2004 9:52:12 AM	217.237.95.XXX

Figure 2

Tracking where the user has come to this web page from. Also, this captures an IP Usage History.

Vtracker also has the capacity to remember where a user comes from and can even track the time at which the user came from another website. This is illustrated by “Sites User Has Been Referred From” (Figure 2). This can impact marketing campaigns and the bottom line by showing which websites users are coming from. The more links that originate from a particular website may indicate that the website is generating a lot of interest; this may contribute to marketing decisions that may be made in relation to such a website.

Another statistical feature of Vtracker is that it tracks browser, user languages and web bot activity over time. This is a collective reporting feature available in a different section of the program (Figures 3 and 4).

VISITOR TRAFFIC / PROFILE ADMINISTRATOR [Main Page] [Help]

WEB BROWSER / BOT ACTIVITY
 This section gives details relating to the types of browsers / bots being used when browsing the domain selected independent of date. Scroll down (or click here) the page to see the summarizations for web browsers, bots and cumulative usage.

[Back Up] **Web Browser / Bot Variants**

Number of Uses	Browser / Bot Type
3646	mozilla/4.0 (compatible; msie 6.0; windows nt 5.1; sv1; .net dr 1.1.4322)
3625	mozilla/4.0 (compatible; msie 6.0; windows nt 5.1)
3218	mozilla/4.0 (compatible; msie 6.0; windows nt 5.1; sv1)
2115	mozilla/4.0 (compatible; msie 6.0; windows nt 5.1; .net dr 1.1.4322)
1676	mozilla/4.0 (compatible; msie 6.0; windows nt 5.0)
1660	mozilla/5.0 (windows; u; windows nt 5.1; en-us; rv:1.7.5) gecko/20041107 firefox/1.0
1080	mozilla/4.0 (compatible; msie 6.0; windows 98)
926	mozilla/5.0 (compatible; yahoo! slurp; http://help.yahoo.com/help/us/ysearch/slurp)
906	mozilla/4.0 (compatible; msie 6.0; windows nt 5.0; .net dr 1.1.4322)
396	mozilla/4.0 (compatible; msie 5.0; windows 98; digext)
379	mozilla/4.0 (compatible; msie 6.0; windows nt 5.1; sv1; .net dr 1.0.3705)
376	mozilla/4.0 (compatible; msie 6.0; windows nt 5.1; .net dr 1.0.3705)
360	mozilla/4.0 (compatible; msie 5.01; windows nt 5.0)
293	mozilla/4.0 (compatible; msie 6.0; windows nt 5.1; sv1; .net dr 1.0.3705; .net dr 1.1.4322)
265	mozilla/4.0 (compatible; msie 6.0; windows nt 5.1; .net dr 1.0.3705; .net dr 1.1.4322)
214	mozilla/4.0 (compatible; msie 5.23; mac_powerpc)
213	mozilla/5.0 (windows; u; windows nt 5.0; en-us; rv:1.7.5) gecko/20041107 firefox/1.0
210	mozilla/4.0 (compatible; msie 5.5; windows 98; win 9x 4.90)
208	mozilla/4.0 (compatible; msie 6.0; windows 98; win 9x 4.90)
190	mozilla/5.0 (windows; u; windows nt 5.1; en-us; rv:1.7.2) gecko/20040804 netscape/7.2 (ax)
175	mozilla/4.0 (compatible; msie 6.0; windows nt 5.1; funwebproducts)
158	mozilla/4.0 (compatible; msie 5.5; windows nt 5.0)
155	mozilla/4.0 (compatible; msie 6.0; windows 98; .net dr 1.1.4322)
150	mozilla/5.0 (windows; u; windows nt 5.1; rv:1.7.3) gecko/20041001 firefox/0.10.1
138	mozilla/4.0 (compatible; msie 6.0; windows nt 5.0; .net dr 1.0.3705; .net dr 1.1.4322)
120	mozilla/4.0 (compatible; msie 6.0; windows nt 5.0; .net dr 1.0.3705)
118	mozilla/4.0 (compatible; msie 5.01; windows 98)

Figure 3
 Example of user-agent gathering.

[Back Up] Web Browsers (browsers, client-based site index tools)			[Back Up] Web Bots (crawlers, spiders, etc)		
Number of Uses	Percentage	Browser Type	Number of Uses	Percentage	Bot Type
107	0%	at&t	40	2%	ask jeeves/teoma
3	0%	agilent	4	0%	asterias crawler
32	0%	alexa	5	0%	almaden
3	0%	amigaos	7	0%	baiduspider
755	2%	aol	23	1%	becomebot
7	0%	arachmo	20	1%	boitho
1	0%	asterias	2	0%	converacrawler
14	0%	avant	3	0%	cydralspider
23	0%	boeing kit	1	0%	deepindex
14	0%	bordermanager	756	33%	digext
2	0%	casperxp	1	0%	digger
5	0%	cerberian	1	0%	elinks
22	0%	crazy browser	2	0%	fastsearch
2	0%	crick	2	0%	fast-webcrawler
1	0%	cybercity	3	0%	findlinks
2	0%	epiphany	1	0%	fyberspider
9	0%	firebird	1	0%	gaisbot
2925	9%	firefox	1	0%	geonabot
1308	4%	funwebproducts	35	2%	gigabot
5	0%	galeon	11	0%	mediapartners-google
1	0%	genzyme	175	8%	googlebot
22	0%	grub	1	0%	holmes
263	1%	hotbar	1	0%	http retriever
7	0%	hot lingo	4	0%	infoave
2	0%	htmlab	3	0%	innerprisebot
23	0%	iopus	7	0%	jakarta
19	0%	java	2	0%	libby
56	0%	konqueror	1	0%	linkscan
475	2%	mac_powerpc	17	1%	libwww-perl
91	0%	maxthon	0	0%	lmaden
120	0%	media center pc	2	0%	mewsoft
8	0%	mra	19	1%	msiecrawler

Figure 4

Separation of users from web bots.

Vtracker is more precise at delivering statistical tracking information (such as web page hits) because it is able to differentiate real users from web bots—programs that may be used to artificially inflate the reported website traffic of a website—by separating them and reporting their percentage of bandwidth consumption (by number of requests and via percentage). This function allows quick identification of such activities so that *unbiased* traffic numbers can still be gathered. As indicated earlier, this summarization shows the popularity of various web browsers among the user audience and also shows the access rates of different web bots.

Being able to maintain surfing histories on each user can not only identify what web pages a user prefers the most (Figure 5), but also allows the fundamental process of delivering custom content and navigation, allowing the website architects and webmasters to do the following:

1. Identify the most visited web pages.
2. Identify probable content pages that a user may be confused with.
3. Identify how long a user stays on the website per session each time they visit.
4. Identify how users are going through the website per session each time they visit.

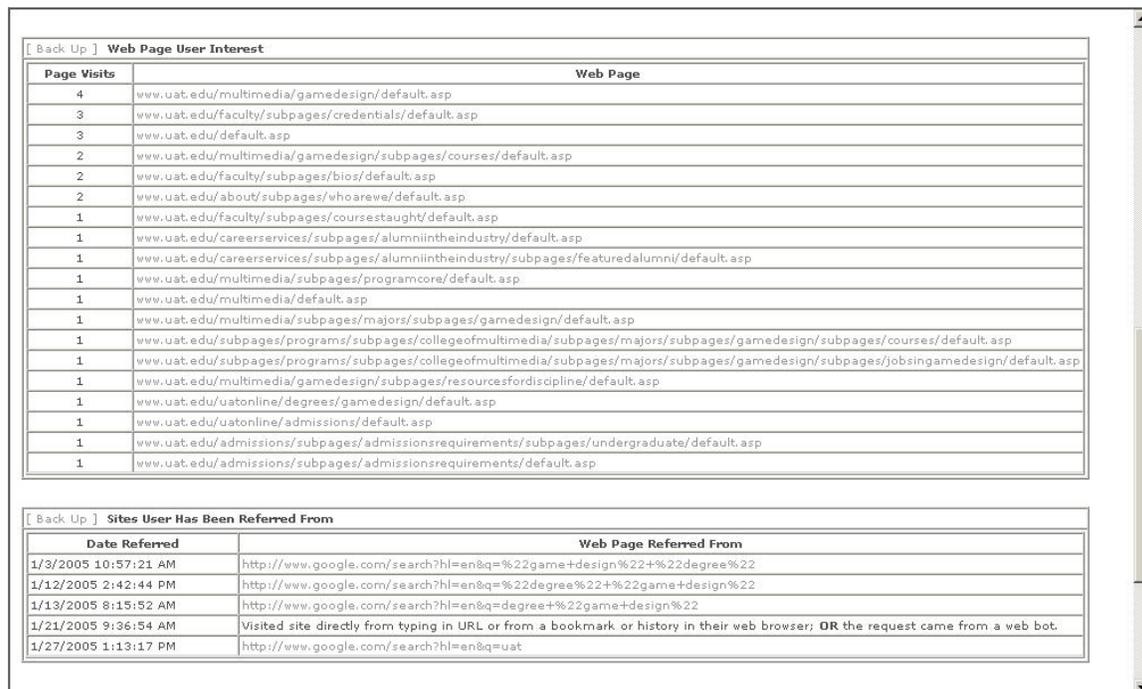


Figure 5

Ranking web page user interest.

Since Vtracker was developed around the core capabilities of being able to cast users into “groups” or “genres,” remembering users and their activities, areas of the website can be associated as belonging to a “group.” From a statistics standpoint, this allows Vtracker to report where a user goes after entering a key page associated to a “group” (Figure 6).

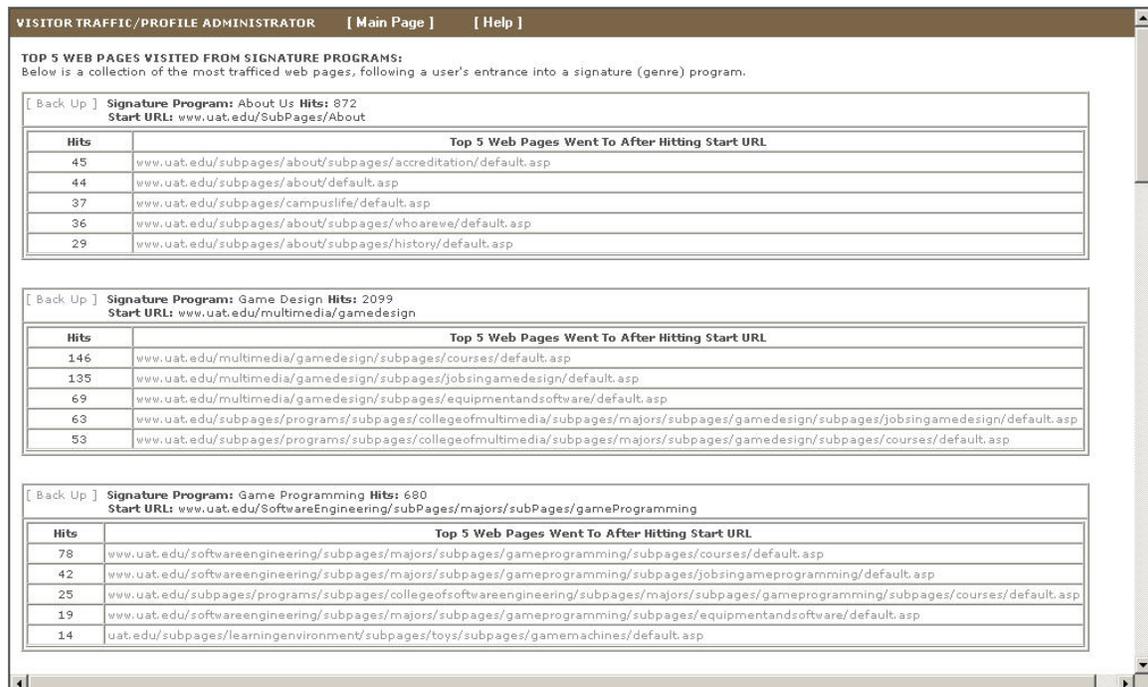


Figure 6
Reporting the top web pages users go to after hitting key pages.

The Future of Vtracker

The inherent abilities Vtracker currently has, i.e., being able to remember users across multiple sessions, as well as what they do once there, opens the door to web-based, persistent artificial intelligence systems scalable to individual users. Imagine a website-based AI system that could communicate in some manner with the user, then learn from that user and provide help to locate content for the user.

As a practical reference, which I know could be developed with technology that we currently have, let me put forth the following concept. Let's say that a visitor comes to a website and starts surfing. If no discernable patterns can be distinguished as to what they are doing, it may indicate that the visitor is curious, bored or simply confused. Vtracker could be programmed to invoke a trigger that would spawn a “virtual person” in the user's web browser. After the “virtual person” is active for that specific user, the “virtual person” might initiate dialogue via text or voice synthesis (if the user's connection

can support it). Because Vtracker can already associate the user to a history, Vtracker's "brain" could be easily expanded to support storing more information about the user, such as dialogue transactions; for instance, between the "virtual person" and the user. Through the dialogue transactions, the "virtual person" would be able to narrow the user as being either curious, bored or confused, and then could offer assistance accordingly.

More "building blocks" could be applied to Vtracker's foundation to push the envelope of a persistent artificial intelligence system. One block at a time, Vtracker and the "virtual person" visual representation of it could evolve in sophistication, action and reaction to users on an individual basis. Over time, Vtracker could also have in-depth knowledge of a user and be able to interact with the user on multiple levels that may not be directly related to the website's purpose. For example, Vtracker may learn that the user really likes a certain baseball team; using simple methods, Vtracker could access data on the baseball team, such as current standings, and communicate relevant data to the user the next time the visitor goes to the website.

Impossibility only exists if you can't change perception.

Biography

Joe McCormack, an alumni of UAT and instructor, has been involved with web development using a wide range of programming languages on different server platforms since the mid 1990's. Joe has published two books on the subject of server/web-based programming and actively develops new methods and processes to automate and streamline functions to improve performance and task-handling abilities of web-based applications on different server platforms. Joe is also the author of two books on Perl/CGI programming.