

Managing Click-Stream Data



By Craig Abramson and Kenny Kistler

Introduction

The wealth of click-stream data gathered from your site can help provide insight into the behavior, buying habits and preferences of the prospective customers who visit your web site. To understand what type of information can be gathered, consider the behavior of a certain John Smith, who decides to buy a new pair of shoes online. He signs onto the Internet and uses a search engine to find what sites sell his favorite brand. As the results come up on his screen, he clicks on the first link. This takes him to an online shoe store, and he begins to browse the site. John comes across the style and size that he wants and adds it to his shopping cart. He's ready to checkout and enters his personal information, credit card number and shipping address. The next screen displays the order information and total cost. After seeing how much the company charges for shipping and handling, John decides to cancel the transaction and go back to the search engine to find his shoes at a different online store. During this entire process, John's click-stream data has been collected by the Web retailer, providing a detailed look at how he got to the site, the Web pages he viewed, the merchandise he considered buying, and the point at which he left the site.

The specific click-stream data that can be collected includes such information as the client IP address, hit date & time, HTTP Status, bytes sent, download time,

HTTP method (get, post), target page, user agent, query strings, server IP address, and cookie data. (Johnson, 2000) If the user found your site through a search engine such as AltaVista, you'll usually be able to determine the referrer page and the search words entered, and you'll be able to access the page of search results on which your site appeared. (World, 2000) You'll also be able to track e-mail click-streams. This data is generated when an individual receives an HTML e-mail with scripting and clicks on a link or an ad. (Johnson, 2000)

The click-stream data typically looks like:

```
dial1-30-45.nbn.net - - [2/Feb/2001:19:54:14 +0000] "GET/html/win95_updates.htm HTTP/1.0"
200                                                                                               54
http://www.infoseek.com/Titles?qt=%22EM+service+release+2%22&col=New+Search&og=%22service+release+2%22&sv=N4&lk=ip-nofra mes&nh=10 "Mozilla/4.01 [en] (Win95; I)"
```

What does this all mean? Taking a look at this example, you'll notice that the IP address of the user visiting your site is listed first (dial1-30-45.nbn.net). The next field sometimes shows the login ID of users who have entered a password-protected area of your site. Many times this is an unused field indicated by - - . The date and time of the page request is listed next in Greenwich Mean Time (GMT). Following this is the name of the page viewed on your site. Then the referrer field is given which tells you what page the user came from. In this case, the visitor found your site by using Infoseek.com. The user agent field ("Mozilla/4.01 [en] (Win95, I)") is last, which shows you what browser the visitor was using. In this example, it's Netscape (code-named Mozilla), version 4.01,

English, International version under Windows95. (How to Internet Your Business, 2001)

The Flow of Click-Stream Data

You've seen what a click-stream record looks like, but it's also important to know how they are generated and stored. Click-stream data is created using a corporate information infrastructure that supports a Web-based eBusiness environment. (Bill Inmon, 2001) To begin the process, as soon as a user enters your site, a dialogue manager takes over and determines if this is a repeat or first-time visit. If the user has already been to the site, a personalization of the dialogue is done. If not, the user is sent a standard dialogue. Once the dialogue is completed, it is broken down into click-stream records, which are then stored in Web logs. These records are sent to the granularity manager where they are edited, aggregated, re-sequenced, and summarized. This helps to reduce the volume of data and organize it into a meaningful format and structure. The records are then entered into the data warehouse, usually on a historical, customer basis, where the data can periodically be refined and entered into the global operational data store (ODS). (Inmon, 2001)

If needed, the click-stream records can also be entered into the local ODS, which is the ODS that resides inside the Web site. Such information would include simple transactions so that if a user came back to your site later in the day, the local ODS would remember the previous activity. The click-stream records in the

local ODS can be used by the dialogue manager to tailor a dialogue for a user. This allows you to provide personalized messages for repeat users, which makes your site more appealing. It's important to remember that data passing from click-stream records into the local ODS remains in the Web environment and never enters the granularity manger. Aggregation, editing, selection and other processes have to be done manually. (Inmon, 2001)

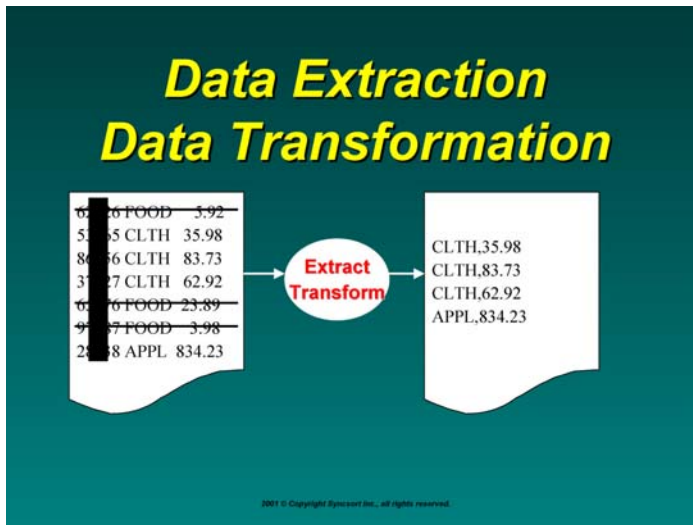
This click-stream data can be analyzed to determine such things as the number of return visitors, purchases made by first-time buyers, the pages that receive the most hits, how much time is spent on a page, which are the best-selling products, and much more. You'll also discover which advertising or e-mail messages are successful, and which ones aren't creating much of a buzz. All of this information can then be used to tailor your messages and greatly enhance a customer's online experience by focusing on the materials that users like best. If done properly, this analysis should lead to increased traffic and sales. But these priceless nuggets of information about customer behavior can be very difficult to find because there are as many as a billion records of raw Web data being generated every day on a popular site.

Achieve Better Performance and Improved Click-Stream Data Management

To find the golden nuggets, you first have to reduce the amount of data to analyze. There are a variety of other techniques that you can follow to optimize the performance of your system and improve the management of your click-

stream data. These techniques include:

Figure 1: Data Extraction / Data Transformation



- *Data Extraction / Data Transformation* - You can reduce the total data to be processed by using Include/Omit processing to quickly identify the exact records that you need from the Web logs. This can help you eliminate large amounts of data and increase the speed required to process queries. To achieve more efficient load performance, you can then transform the record layouts by deleting or reformatting specific fields.
- *Merge* - Similar data can be merged together into a single file to analyze. During this process, you can specify how the data is ordered. For example, if you are merging two lists of names, you can order the data alphabetically.

- *Join* - By joining data, you're matching keys in multiple files and creating one record from two records that have a common key. For example, if one record contains Doe, Jane, 65 Any Street, 07677 and another record contains Woodcliff Lake, NJ 07677, joining the data will create one file that contains Doe, Jane, 65 Any Street, Woodcliff Lake, NJ 07677. To optimize the record format, you can perform inner, outer, left and right joins.

- *Pattern Matching Field Extraction* - Another technique is to search for patterns anywhere in specified fields. Once a pattern is found, you can then extract portions of the field. For example, you can search through a list of referrer Web sites and extract only those that end in HTML.

- *Output Record Numbering* - To make it easier to sort through data, you can add a unique record identifier number to the output. This record number can start at any value and can be useful for databases with a uniqueness constraint. For example, you can add a record identifier to the following data:

HJ28983,shirt,red,12.49,2

HJ28983,shirt,blue,12.49,1

IL93848,pants,blue,32.11,3

With the record identifier number, the data now reads:

0001,IL93848,pants,blue,32.11,3

0002,HJ28983,shirt,blue,12.49,1

0003,HJ28983,shirt,red,12.49,2

- *Web Log Format* - There are two standard Web log formats that you can utilize to ease the processing of your click-stream data: The Microsoft Standard format (Microsoft Professional Internet Services format) and the National Center for Supercomputing (NCSA) Common Log File Format. The field definitions of the click-stream data are now already defined for you which simplifies and reduces development time when dealing with these formats.
- *Presort, Bulk Load* - For high performance relational databases such as Oracle, Sybase, and SQL Server, a bulk load followed by a separate create index step provides you with the fastest way to load data. Keep in mind that all of these database manufacturers recommend presorting the data before the bulk load, so that the create index can bypass the sort operation. This can cut the total load time in half.
- *Presort, Regular Insertion Load* - When a bulk load is not applicable, the load step can still be accelerated by presorting the data, using the clustered index as the sort key. Significant savings in load time can be achieved.
- *Extract/Unload* - To accomplish the fastest large data extracts, remove all GROUP BY, ORDER BY and DISTINCT clauses from the SQL SELECT statement that unloads the data, and then perform those operations with a special-purpose sort tool on the unloaded file. Keep in mind that SUMMARIZE

processing is a much faster equivalent to the GROUP BY and DISTINCT clauses, while the KEY option is a faster equivalent to the ORDER BY.

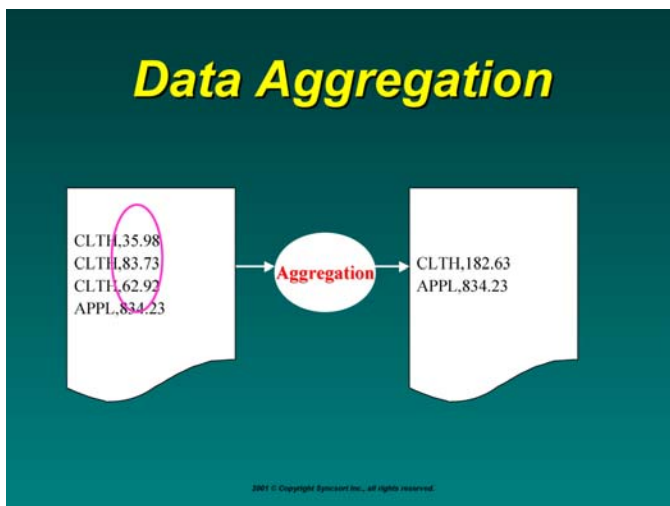
- *Reorg* - Reorg processing that consists of an unload, sort, reload sequence will run significantly faster when the other techniques are used for the unload and reload and a special-purpose sort tool is used for the sort.

Improve Query Analysis and Runtime Performance

“... expect anywhere from a tenfold to a thousandfold improvement in runtime performance by having the right aggregates available ...” Ralph Kimball

In order to analyze all of the click-stream data that's stored in the data warehouse, it's important to create aggregates. This will help you to dramatically improve query and runtime performance.

Figure 2: Data Aggregation



- *Aggregate Building* - You can use SUMMARIZE processing to generate pre-stored aggregates. This technique will help you optimize query processing and data warehouse response time.
- *Data Partitioning* - To split your data into separate partitioned table ranges, you can use multiple OUTFILE processing. Then you can utilize Pre-Sort to accelerate the database load itself. Oracle, Sybase and others recommend pre-sorting for the fastest possible index creation times.

Corbis Corporation Reduces Time to Process Click-Stream Data by Almost 80 Percent

Corbis Corporation is taking advantage of its Web servers to optimize its click-stream processing. As a provider of digital images for consumers and creative professionals, Corbis uses its Web site to do business with a growing number of online shoppers as well as the world's most popular publications. Behind the scenes at corbis.com, Web logs record click-stream data including such details as the number of unique visits to the Web site, the most popular pages, the most purchased products, and the pages that seem to be "session killers," where visitors frequently stop the session and leave the site.

Every day, Corbis experiences nearly half a million visits to its Web site during which customers browse through extensive online art galleries, download pictures, order framed prints, or license specific images for repeated use. Given

the dynamic nature of this Web traffic, the massive server logs record more information about the Web site and its visitors than analysts at Corbis can realistically use. Every link, picture, and page that is accessed by each visitor is recorded. At the end of a typical business day, there are literally hundreds of megabytes of information to sort through to access the hidden customer data that analysts covet most.

Identifying the crucial information in these Web logs, isolating it, and preparing it for warehousing and analysis is a task requiring time, resources, and expertise. C.J. Venkataraman, Senior Software Architect at Corbis, supervises this process. For over two years, Venkataraman has managed the flow of data from the company's operational data sources, between its e-commerce systems, and into its warehouses. With his team of programmers and designers, he defines the technology architecture at Corbis prior to its implementation by the business systems group. The platform consists of dual proxy Compaq Proliant servers with 1.5 GB of memory and Pentium III 600 processors running Windows 2000 and the Microsoft SQL Server database engine. A key function of this architecture is to facilitate Web log processing.

“We have 26 Web log servers at Corbis, and we copy each server’s log files onto one huge server,” says Venkataraman. “Basically, we merge them all, filter the resulting file to include only the customer and Web site information that we want to keep for analysis, and then compress it for storage. Once it is stored and

analyzed, we can more clearly see how many visits occur to our Web site, what our customers are doing on the site, and the top domains that our visitors are coming from.” In addition to customer analysis, Corbis uses the Web logs to evaluate its partnerships with Yahoo! and AltaVista. “Our click-stream data provides insight into who has been sending us the most Web traffic,” says Venkataraman.

Managers and analysts at Corbis can also use Web log information to research how their customers think. For example, click-stream data may reveal that many visitors are filling online shopping carts, but leaving before actually purchasing the items. After further investigation, Corbis may decide that by redesigning the checkout page, the percentage of visitors who follow through with purchases can be increased.

In theory, this method of customer research sounds simple and practical enough. But when the Web log files contain data from some 500,000 hits a day, the resulting numbers can be overwhelming. Each log file at Corbis swells to at least 200 MB in size before it is merged with the other Web logs. The resulting merge can easily exceed five GB daily, considerably slowing down the filtering and compression process. “End to end, it was taking close to five hours a day to complete,” says Venkataraman. “We don’t need most of what we have in our Web logs, so sifting through all of these files every day is a long process. We were using a tool that we made in Visual Basic, but it wasn’t getting the job

done.” Concerned that excessive time and resources were being spent converting daily Web logs to a single, compressed flat file for their analysts, Corbis began to search for alternatives.

That is when Venkataraman turned to a business associate from Microsoft's iDSS group, who suggested that they try SyncSort, a high-performance sort, merge, and copy tool from Syncsort Incorporated in Woodcliff Lake, New Jersey. "They use SyncSort at Microsoft iDSS, and my friend was the one who actually recommended my going down that path," says Venkataraman.

When Microsoft was building an integrated data warehouse, they incorporated SyncSort for several steps in their Web log processing, and were able to turn a billion records of raw Web data into 500 MB of clean data to upload into their warehouse. Hearing about this, Venkataraman was intrigued.

After receiving and implementing a demonstration version of SyncSort, Corbis believed that they had found their solution. "The speed at which SyncSort merged and sorted our log files was impressive," says Venkataraman. "I was surprised how it could copy, merge, and sort so many huge log files with additional filtering added in, and still process so quickly. It brought the daily routine down to one hour, which saves us about four hours on average per day." In fact, Corbis did not explore other products after testing SyncSort, because it met the key criteria they set. "Its performance was great. It was easy to use and

customize, and my contacts at Microsoft were quite enthusiastic about their results with it."

In addition to the merits of the software itself, Venkataraman experienced another asset of the SyncSort package. "When we were in the evaluation period, we had a lot of questions that Syncsort's support team helped us answer. They followed up on a regular basis, promptly and helpfully. The technical support made our decision to incorporate SyncSort easier."

Since making that decision, Corbis has implemented SyncSort into its regular development and production cycles with welcomed success. "Performance was our main goal with SyncSort, and now we're experiencing significant time savings of almost 80 percent with the same amount of data as we had before. We've been using it for several months now, and we process all our daily Web logs right through it."

In addition to the strides made in Web log processing, Corbis is developing ways to refine the collected data for maximum advantage. Venkataraman adds, "Our use of SyncSort will be expanding in the near future. We have many data quality projects that are starting up soon, and we're looking forward to using SyncSort to run data quality checks in our warehouses."

Corbis has been able to use this specially-crafted combination of technological architecture and high-performance tools to turn a potential corporate advantage into an actual one. By overcoming the obstacles in their Web log processing, Corbis can more efficiently use the rich mines of data in its Web logs to better understand customer behavior and, in turn, increase profits.

Summary

Click-stream data can be a marketer's dream. It offers a detailed glimpse into the activities of a visitor to your site. Once analyzed, this information can help you enhance their online experience and hopefully increase sales. But to achieve this analysis in a fast and efficient way, you'll have to cut through the enormous amounts of data collected each day. Using high-performance application acceleration software will help reduce the volume of data and organize it into a meaningful format and structure. There are also numerous techniques you can use to make this data more manageable, including data extraction, data transformation, merge, join, pattern matching field extraction, Web log format, data aggregation and data partitioning. By incorporating the various tools and techniques discussed, you'll be able to optimize the performance of your system and improve the management of your click-stream data. Best of all, you'll quickly be able to uncover the golden nuggets of information about customer behavior that's buried inside.

References

Inmon, William H. "An Architecture for Managing Click Stream Data," BILLINMON.COM, (March, 2001), 5-7, 10-11

Johnson, Rich. "Webhousing and Making Sense of Clickstream... A Marketer's Dream!," ebi Conference PowerPoint Presentation, (2000)

"Web Logs, Statistics & Sniffers: How to Understand the Raw Data," How to Internet Your Business, www.howtointernet.com/rawdata.html, (2001)

World, Jim. "An Introduction to Web Site Traffic Analysis from Your Server Log Files," www.virtualpromote.com/trafficreports.html, (2000), 2-3

Biography

Craig Abramson is a technical analyst at Syncsort Incorporated, focusing on the latest data warehouse performance solutions. He has over seven years experience in the field working on projects dealing with data warehousing, database management and Web log processing.

Craig Abramson

Technical Analyst

Syncsort Incorporated

50 Tice Boulevard

Woodcliff Lake, NJ 07677

201-930-9700 ext. 308

Email: cabramson@syncsort.com

Kenny Kistler is a technical analyst at Syncsort Incorporated, focusing on the latest tools for Web log processing and data quality management. He has over four years experience in systems implementation and documentation for a variety of operating platforms.

Kenny Kistler

Technical Analyst

Syncsort Incorporated

50 Tice Boulevard

Woodcliff Lake, NJ 07677

201-930-8233

Email: kkistler@syncsort.com

For more information
or to arrange a free trial,
call Syncsort at
(201) 930-8200
or visit the Syncsort
web site at
www.syncsort.com