Paper 017-31

Web-Site Analytics – A Land Mine or a Gold Mine?

Jerry J. Hosking, SAS Institute Inc., Cary, NC

ABSTRACT

Do you have the information that you need in order to optimize the decisions that you make about your Web site? Can you tell if your Web site is effectively influencing its visitors?

This paper is a practical guide to avoiding the hidden land mines of Web-site analysis and providing a gold mine of information about your Web site and its use. Technical issues related to using the Web log and other data to understand activity on your Web site are discussed. This paper also outlines major steps for implementing a Web analytics solution to provide guidance about your Web content. Topics include data accuracy, suggestions for automating processing, domain experience on issues such as tagging, cookies, standards and definitions, and specific practices based on a real-world implementation of SAS Web Analytics. The techniques presented can help you discover the critical nuggets of information buried in your Web data.

INTRODUCTION

Web sites have become an essential channel for communication. Whether or not they are used directly for commerce, they are a fundamental avenue between your organization and the people—your customers, vendors, partners, and employees—with whom you need to connect. By using Web analytics, you can learn about the activity on your Web site. That knowledge will guide you in making decisions that will improve your Web site's performance. Each topic included here is an issue I've addressed in the Web analytics implementation at SAS. In essence, this is a map to help you avoid the land mines that I've been dodging. After reading this paper, you will be prepared to carry out an analytics solution for your site. You will understand the implications of both business and technical issues that are relevant to designing and maintaining a Web analytics implementation.

CONTINUOUS IMPROVEMENT

Is there any organization today that doesn't have a Web site, whether it is an Internet or intranet site? If your organization invests in a Web site, there is the assumption that you expect to accomplish something from its presence. The anticipated result might be additional revenue from commerce, better-served customers, greater brand awareness or name recognition, increased employee productivity, or a clearly communicated corporate culture. Just as any other delivery process in your organization can be evaluated and improved upon, your results and your Web site's activity can be measured and strengthened. Web site analytics is a key element in the continuous improvement cycle of delivering an effective Web site.

If your organization already has a culture of **continuous improvement**, your efforts to adopt analytics to evaluate and improve your Web site will be easily understood and accepted. If the idea of continuous improvement is new, you can be a pioneer in introducing such an approach. There are numerous books and Web sites that formally define and fully discuss the concepts of continuous improvement (Value Based Management.net 2005). As applied for our purpose of Web analytics and Web site evaluation, we've adopted this definition modified from Wikipedia (Wikipedia October 3, 2005) and other sources (Alberta Government 2004).

Continuous improvement suggests that a process or product should always get better as knowledge about it and experience with it accumulates over time. Simply put, *continuous improvement* is the ongoing evaluation and resulting change of processes, products, programs, and services to make them work better.

This concept is depicted as a continuously repeating cycle of steps for the measurement, evaluation, and change implementation surrounding a specific business goal (Figure 1).

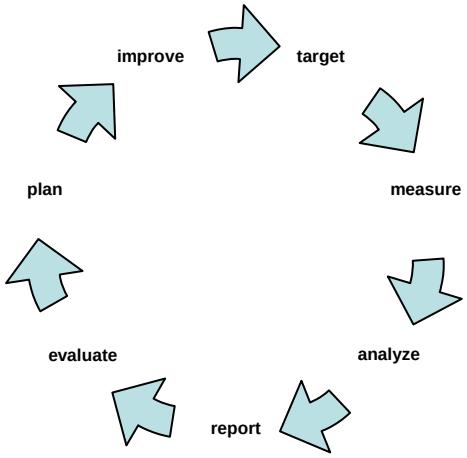

Figure 1. Web Site Continuous Improvement Model

A Web site is a perfect model for applying continuous improvement. The nature of a Web site—an entity comprised of many parts—lends itself to the idea of making incremental changes, capturing the effect of those changes through Web analytics, and continuing to plan and implement additional changes based on those results. But beware! Expect the results of your analytics to have an impact. When you demonstrate more about the use of your Web site, expect there to be greater demand to change the Web site to address the issues that are revealed in the results.

LAND MINE OR GOLD MINE?

So when is a Web site and its associated analytics a land mine and when is it a gold mine? More appropriately, it might be seen as a gold mine surrounded by land mines as barriers to accessing the wealth within. Let's take a look. Both land mines and gold mines have a great deal of uncertainty and unknown about them. But the resulting outcomes from each can be drastically different. At the least, land mines create obstacles to getting to a destination. They are hidden, with locations unknown or undocumented; they activate without warning and ultimately can be devastatingly destructive. Gold mines might also be hidden. But in legend, there is always a map. This map might lead through treacherous terrain and dangerous digging, but when correctly deciphered ultimately delivers a mother lode of treasure.

What are some of the land mines associated with Web analytics? Primarily, they fall into two categories, **business issues** and **technology issues**. For many of you, the landscape of the technology issues is familiar -- problems that can be solved with programming or processing solutions. The business issues might present challenges that will pull you into uncharted territory and cause you to form partnerships with colleagues in non-information technology business units of your organization.

BUSINESS ISSUES

GET A GOAL!

Understanding your Web site's specific business objectives is fundamental to determining relevant metrics to be gathered.

Proceeding without explicitly-stated goals can be a land mine, depending on how it is handled. Consider this conversation between Mark, a marketing strategist, and Wanda, the Web analyst, at the water cooler (Sullivan 2005).

> Mark says, "Hey Wanda! You have that Web metrics thing. What cool data can you get me?"
> Wanda replies, "Well, what sort of stuff do you want?"
> Mark explains, "Well anything really good that will be helpful."
> Wanda responds, "Well, what would be helpful?"
> Mark states, "Anything really. What have you got?"
> Wanda questions, "What do you need?"
> Mark suggests, "Show me what you've got and I'll tell you."
> Wanda counters, "Tell me what you want and I'll see if we can get it."

Mark persists, “How do I know what I want unless you can tell me what you’ve got?”
Wanda entreats, “How do I know what to give you unless you tell me what you need?”
Mark: (sounds of hair ripping)
Wanda: (sigh)

Knowing your Web site’s specific business objectives increases the likelihood of finding success with the site. I recommend collaborating with colleagues in the business units of your organization. They can provide guidance about the intended achievements from your Web site. You might feel that you are experiencing the chicken-and-the-egg dilemma, as depicted in the scenario with Wanda and Mark. Do your best to figure out one purpose that you understand your Web site is supposed to accomplish. Use this goal as your guiding principle to determine relevant metrics. Based on that one stated objective, what if anything on your Web site indicates that your visitor has performed an action that you recognize as achieving that objective? This becomes the most important action to measure. Trust me, when doing Web analysis, you are going to have more numbers and more metrics than you can digest. Keep focused on a small number of critical indicators of success. You deactivate the “meaningless number” land mine by focusing on relevant and actionable metrics.

Some business objectives are directly addressed by measures of activity on your Web site. For example, an objective to increase the number of people who request further information about the treasure maps you sell can easily be answered by measuring the use of a “contact me” form. An objective to increase name recognition for your organization, “The Lost Dutchman,” as the supplier of the world’s most reliable treasure maps can be addressed by measures from your Web site as well as supplemental data. You have to decide what on your Web site indicates that your organization’s name recognition is increasing. An increase in new visitors could indicate that more people are aware of your organization. An increase in the unique number of Web sites sending traffic to your site could indicate that more people are aware of your organization. Those are both measures available directly from your Web site. But increasing name recognition is an objective that would be a good candidate for supplemental measurement such as a survey or interview that directly assesses unaided recall or top-of-mind awareness of organization names.

TEAM UP!

Expertise and input from varied perspectives and disciplines will improve your Web analytics implementation.

Excluding any of the roles involved in the delivery and analysis of a Web site is a land mine. The consequences of missing the participation of all the functions—such as Web site strategy, content production, creative design, application development, Web server system administration, and Web analytics—will be a site that is less conducive to the production of useful analytics. You can analyze any Web site. Decisions about site design and techniques used throughout site development will have an impact on the ability to produce useful analytics from the Web site.

Enlisting the participation of colleagues in the continued development of the Web site in ways that make the site more analyzable will produce more accurate and helpful results. Unless you perform all of the roles in delivering your organization’s Web site, become friends with the people who produce the content, develop the applications, and run the Web servers. How they choose to set up and deliver the site will contribute to producing the data you analyze. Whether it takes brownies, beer, or other enticements to ensure their cooperation, the investment is worthwhile.

What kinds of issues can they help you address? As mentioned in the previous section, “Get a Goal,” the strategist is important in identifying specific business objectives. These strategists should also be some of the primary consumers of the analysis you produce. The insights that the Web analytics produce should help guide them in decision making about the Web site.

Content production staff can make or break your data by including, or failing to include, code in your pages that is essential to identifying visits to your site. This is discussed later in the “Sessionizing” section.

Creative design team members can exercise discretion about the way navigation initiates new browser windows that affect the identification of referring traffic. For example, if they choose to use JavaScript coding to open a new window as the result of clicking a link on a page, the resulting Web data does not reflect any value for the referrer. In essence, you’ve lost information about the path your visitor has taken. Simple HTML coding with target= “_blank” will retain the referring page in the Web data. There might be compelling reasons to choose one method over another; but in the absence of those reasons, design and code in favor of producing data that is best suited for analysis.

The application developers are key to determining what data is actually included in the Web server logs for analysis for URLs used in the applications they design. For example, in most situations, the developer has the option of coding the application to use either the GET or POST (Korpela 2003) method for HTML forms. Choosing the GET method retains the significant query string portion of the URL in the Web data. The POST method does not. The availability of query string data is critical to knowing the actual content of the page viewed. Generally speaking, unless the purpose of the form is to update critical data, the GET method can be safely used.

The server administrators are the linchpins for delivering the data to be analyzed. Whether you are analyzing server logs, application data logs, or page tagging logs, the server administrators are the likely sources for controlling the format and content of those logs through the server configuration. Producing the logs in a standard format, such as Extended Log Format (Apache 2006), will eliminate the need for customizations to process them with SAS Web Analytics. It also produces logs with the richest data record. If, however, your systems administrator has augmented the log format with additional fields, those fields can be ignored or included in the processing, if they are valuable. Systems administrators also affect all kinds of activity on the site through the server configuration that can affect the flow of traffic through your Web site.

BE RELEVANT!

Frame the analytics results in context that will resonate with your audience.

An analysis that is just a deluge of numbers, lacks an apparent relationship to business objectives, or has no implication of connecting activity on the Web site to actions of people is a land mine.

Sift through the analysis you produce. Choose a small number, maybe even one, telling metric. This should be a measure that speaks to the business objective you have worked with the marketing strategist to identify. Distill it to communicate the essence of what this analysis tells you. As applicable, present it with a germane representation: people, if your metric reports influencing behavior of people on your Web site; currency, if your metric is connecting activity on the Web site to revenue. Be prepared to handle loads of deeper questions this analysis provokes.

SPREAD THE WORD!

Let people know you can share intelligence that will help them be more effective.

The perception that the Web analytics will reveal bad news or everything that is wrong with the Web site can be a land mine. To a certain extent, whether or not this attitude is present might be a reflection of your corporate culture as it relates to assessment and continuous improvement at a general level. If revealing that something isn't working as well as you desire is threatening at your organization, you will have a challenge.

A counterpoint to the hesitancy to embrace Web analytics due to a fear of revealing unsuccessful efforts is the promise to identify actionable measurements that can lead to improved results. Be a proponent for the opportunity to use genuine instrumentation rather than just "best guesses" in determining design, usability, and content issues about your Web site. Get your relevant metric (described above) in front of everyone. Like all superheroes, take an oath only to use your powers for good purposes.

TECHNICAL ISSUES

WEB DATA 101

To be sure that our discussion is based on a common understanding, I am providing a few basic definitions of terminology in wide use in the Web analytics industry.

URL (Uniform Resource Locator) is an address that specifies the location of a file on the Internet or intranet.

Requested file is the entity or component, such as a page, partial page, graphic, or application requested via the client from the server. The request is communicated as a URL. Although sometimes referred to as requested page, requested file usually indicates a more specific entity that produces a part of the overall content of a page viewed. In fact, most views in the client represent displays of many requested files to compose a page including navigation, graphics, and text content.

Client is typically a browser, but can be any device used to access Web content.

Client ID is an Internet protocol (IP) address or host name represented by that IP address.

Referrer is the URL of the Web page where a visitor clicked a link to come to your site.

Referring domain is the host name part of the referrer URL.

Query string is the part of a URL, beginning with a question mark (?), that is composed of name-value pairs of input, separated by ampersands (&), that conveys parametric data to the server. For example,

```
?firstname=Jerry&lastname=Hosking.
```

User agent is a text string used to identify a browser or other client accessing a Web page.

An excellent method for finding relevant definitions for any Web-related terminology is to use the *define* keyword offered by Google at www.google.com. Simply prefix your query term with the word *"define:"*. For example,

```
define: web analytics
```

SAS WEB ANALYTICS 101

The two-step extract/transformation/load (ETL) process of SAS Web Analytics provides intervention points within the first step (referred to in SAS documentation as *usermods*) to allow tremendous flexibility and power for customizations to your Web data (Figure 2). You can ignore or limit your use of this feature to the extent appropriate for your implementation. The weblog detail data set (indicated as #1 in Figure 2) is an interim data set, produced by the first step of ETL, that reflects the reorganization of your individual Web server records into a series of events attributable to a visitor. The second analytical ETL results in the production of summary tables that are report- and analysis-ready. The major data sets of the Web data mart are:

2. a date-based data set containing an observation that represents each visit to the site (dated)
3. a date-based data set that contains an observation that represents each request made to the server (detail)
4. a series of data sets representing activity summarized for days, weeks, months, quarters, and years (summary)

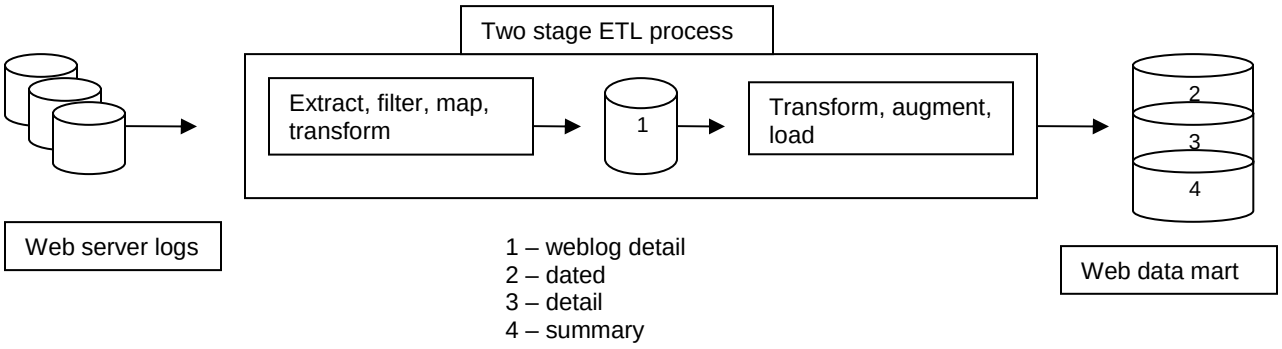

Figure 2. SAS Web Analytics ETL Process

DATA ACCURACY

By its nature of anonymity, Web data can seem rife with inaccuracies. In reality, the data you collect for analysis is quite accurate. Web servers are quite accurate in recording exactly what was requested of them. It's what is requested of them that can be manipulated, purposefully or unintentionally, that creates a question of confidence in the results. This section describes common data manipulation land mines that can blur the picture of activity on your site and the recommended defusing techniques. If your purpose is to analyze, report, and predict based on exactly what occurred on your Web server, then the issue of data accuracy is moot. Collect the data in your logs and have at it. If your purpose is to understand, improve, and influence behavior on your Web site, then you will want to edit your data to produce the most exact representation of human behavior that occurred on your site. Your objective is to cull any data that is not activity performed by true visitors who viewed your content.

REFERRER SPAM

Referrer spam results when techniques are used to make requests at a Web site using a fake referrer. In essence, this produces data that says a visitor arrived at your site as a result of using a link from the referring site. There are three primary purposes for referrer spam. Probably the most prevalent is to cause the appearance of high value for the referral site to search engines. There are many sites that publish referrer statistics for their own site. When they do this, they often produce a link to those false referrers. This benefits the spammer because of the free link. It also gives the spammer's site improved position on results pages from search engines due to link-counting algorithms used by search engines (Wikipedia December 19, 2005). False referrer data also results when people fake the referrer so as to prevent knowledge about Web site use, browsing patterns, or just plain old paranoia. Occasionally, robots (see the section "Robots, Spiders, and Crawlers") set false referrers (such as your home page) that are used while they traverse your site. This results in data that shows a referral from your home page to a subordinate page on your site, even if there is no link on your home page to that subordinate page.

There is nothing in the data that clearly distinguishes a request that uses a fake referrer from a request that does not use a fake referrer. Data with referrer spam from true spammers or robots, as opposed to the individual who is simply masking information about their Web use, can often be detected due to the sheer number of requests from the client and the rapidity of those requests. In addition, requests from robots sometimes use identifying information in the user agent posted in the request. See the section "Robots, Spiders, and Crawlers" for more suggestions about identification and removal of undesired data from automated sources.

There are three main opportunities to intervene against referrer spam. All three require identification of either IP addresses or host names of the originator of these offending requests. There are many inventories of automated clients available on the Internet that can be used to seed a blacklist for filtering your Web data. An example specific for referrer spam is ReferrerCop at www.referrercop.org/blacklist.php. In addition to using such lists, you will want to examine the SAS logs from your daily processing for the identification of other possible automated clients.

The first intervention opportunity is at the Web server level. Using server configuration settings, you can direct Web data generated as a result of requests from identified automated clients to a secondary Web log (Apache 2006). This preserves the availability of data from those sources, but separates that data from data generated as a result of requests from true visitors. Alternatively, also using server configuration settings, you can deny access to the Web site (or specific areas of the Web site). This generates data with a status code denoting that access was forbidden. This data cannot be distinguished from any other requests that generate forbidden access. If you are not concerned about the reason for creation of forbidden-access requests, it is a reasonable approach to separating requests from automated clients. Both of these methods are going to require cooperation from the Web server systems administrator for implementation. Remember, **TEAM UP!**

The second intervention opportunity is at a pre-Web Analytics processing point. Again, maintaining a list of IP addresses or host names to be used for excluding from your data, you can use filtering scripts that operate on your raw Web logs to produce a cleaned raw Web log.

The third intervention opportunity is during the first step of ETL of your Web analytics processing. As part of your Web data mart configuration, a special clients configuration data set is maintained. You can populate this data set with the IP addresses (or range of IP addresses) for the referrer spam originating hosts. By including the IP addresses, requests from these clients can automatically be excluded from the Web data mart during ETL. This is a straightforward, self-contained solution that gives you total control of your filtering list.

SESSIONIZING

To understand behavior on your Web site, you must include a full following of the activity ascribed to each person. Again, the basic anonymous and stateless nature of the Web presents challenges to connecting all the activity of a single person yet separating it from the activity of another person. The act of grouping activity from a specific visitor for a certain period of time is *sessionizing*.

Accurately connecting all the requests for a visitor for a prescribed time can be a land mine. Even if your site requires security (such as a login) that provides authentication, you are faced with using methods to sessionize activity from a visitor yet the methods don't require undisputable identification of the person. Challenges related to sessionizing include:

- Assigning acceptable, yet anonymous, identification to a visitor
- Re-linking requests from the visitor that occur prior to the assignment of that identification with requests that contain the identification.

The two most recognized methods for collecting data for analyzing activity on Web sites are the raw Web server logs and page tagging server logs. Web server logs are created automatically; their contents are controlled by the configuration of the Web server that produces them. Page tagging server logs can be created through the introduction of a small amount of JavaScript code added to your Web pages. Each method has its advantages; deciding on which one to use is a decision point for you to consider. Table 1 highlights the differences between the two implementations:

Issue	Web Server Logs Method	Page Tagging Logs Method
Maintenance	No additional effort	• Additional effort • Can be minimal for sites already using standardized content sections • Include call to tag code in existing header
Data loss	Loss of data due to page caching (proxy server/back button)	• Loss of data due to failure to tag pages (correctable) • Client and server error reporting not included • Loss of search engine spider data
Data enhancement	Extensive data, every request to the server	Can include additional data from meta element's properties and values

Table 1. Differences between Web Server Logs Method and Page Tagging Server Logs Method

You can use either method or a hybrid of the two, depending on the effort available for your implementation and the required results. A good starting approach is to use the Web server logs and include a page tag to enhance sessionizing. This provides some of the advantages of the page tagging method with the ease of the Web log method.

Optimally, your site is organized so that a file containing a routine or template header file is included as part of every Web page. In this header file, include a call to a file containing JavaScript that (1) checks for the existence of a cookie containing a Web analysis identification value, and (2) sets a Web analysis identification cookie (if not present). In the future, if you decide to take further advantage of page-tagging analysis, you can modify this JavaScript code to request a GIF file. This request will include additional parameters of interest, such as page title or metadata that describes the content of the page being viewed. You will need the services of either your content production staff or application developers to modify your header file or create the JavaScript code. You'll also need to work with your Web server administrator to make sure that the server configuration includes cookie logging. This is not writing the cookie values to another log; it is writing the value of the cookie in the Web server access logs. Essentially, this adds a value to the end of each record in the Web server access log, producing a result like this:

```
64.233.187.104 - - [21/Nov/2005:13:00:10 -0500] "GET /excavation/shovels.html
HTTP/1.1" 200 909 "http://www.tldm.com/excavation/index.html" "Mozilla/4.0
(compatible; MSIE 6.0; Windows NT 5.0)" "wa_id=968071547711.4673"
```

Although there is good evidence that people do periodically remove cookies, the use of cookies to anonymously identify visitors is still the best method for linking a visitor's activity on your Web site. The number of visitors who reject cookies issued directly from the site they are visiting (first-party cookies) is still small. I recommend that you use a brief, easy-to-understand explanation of cookie use on your site in your privacy statement. If appropriate, assure the reader that the cookie provides no identifying information *about them as an individual*. If there are advantages *to them* from the use of a cookie, mention that. For example, if you present any kind of customized view based on preferences they have selected on your site, let them know they are gaining value through use of a cookie.

As long as the cookie is not rejected for the visit, the Web analysis identifier will serve well to track the activity for that visit. Typically, this identifier is also used to distinguish new visitors from returning visitors. As the length of time since the last visit and the current visit increases, so does the likelihood that the original cookie has been removed. Therefore, the reliability of data for identification of new and returning visitors for long term (a year) declines.

To use the identifier that you have defined in your cookie (`wa_id`), you need to add this variable name to the Web data mart configuration in two places:

- First, define it as a field to be kept as one of your Web log fields. You can do this using the processing interface accessible via your Web data mart properties or by updating the "wbfields" data set.
- Second, identify the name of your field in the "wbconfig" data set as the value for the `ubiquitous_identifiers` parameter name.

If you already have a cookie-based identifier in use on your site, you don't have to add a second identifier just for Web analytics. As long as that identifier is used on all pages and the value is recorded in the Web logs, specify that particular identifier in the Web data mart configuration. Use of the cookie meets the challenge of assigning acceptable and anonymous identification to the visitor.

SAS Web Analytics addresses the challenge of re-linking requests from the visitor that occur prior to the assignment of that identification with requests that contain the identification. When new visitors arrive at your site, they have not yet received a cookie from your Web server. It is only during the serving of the first page that the cookie is delivered. That means that there is no cookie identifier associated with the first request of a visit. Once the cookie is delivered from the server, all subsequent requests for that visit contain the cookie identifier value. It's obvious that analyzing data about your Web site that doesn't reflect the entry point for the visit connected with the activity for the remainder of the visit will prevent accurate understanding of behavior on your Web site. Through patented technology, algorithms specifically designed to re-match the first request of a visit with the remainder of requests for the visit, SAS Web Analytics connects the data into a complete record for each visit.

DATA CLEANING

Standardizing the values of your data will improve the usability of the data for analysis and reporting. The data written to a Web server access log is a record of what was requested of the Web server. Ultimately, your goal is data that has been cleaned to represent human activity on your Web site. Yes, you are modifying your data. The limit you should place around this modification is to refrain from any change that will have the appearance of enhancing activity and behavior on your site.

Within the first step ETL, there are multiple points at which you have the opportunity to operate on your data. Two primary times for intervention into the processing are:

- just after the raw data is read in
- just before the data is written to a detail (weblog_detail) SAS data set (still prior to summarization)

The first time (after input) is a great time to apply any supplemental filtering (beyond the automated robots and special clients filtering done automatically for you) to exclude data from analysis. It also gives you a chance to do any variable creation or data manipulation based on the initial values of fields as they appeared in your Web logs.

The second time (before output) is after all the processing and filtering based on your Web data mart configuration, but before output to your detailed data set. This is a good time for additional data cleaning, content categorization, and variable creation based on cleaned data.

PAGE NAMING

There can be variability in representing the values for the requested file that ultimately mean the same thing. Depending on the configuration of the Web server, most clients and servers arrive at the same page (for example, the "home" page for The Lost Dutchman company) for all values for the requested file listed here:

- www.tldm.com/index.html
- www.tldm.com/
- www.tldm.com
- tldm/index.html
- tldm.com/
- tldm.com

In your reporting of traffic for these pages, you could end up with separate entries for the number of views for each of these pages. Using only one of the numbers would not accurately reflect the use of The Lost Dutchman home page.

Standardize the values for the “home page” for any directory level within your site. With the cooperation of your content production staff and applications developers, limit the names used for the home page and superior level pages for Web sections on your site. For example, besides the home page, you have 10 main content sections on your Web site. Use the same page name (index.html, default.htm, default.jsp or whatever is appropriate) for the top level page for each of those sections (such as /excavation/index.html, /panning/index.html and /maps/index.html). In addition, ask your content production staff to be consistent in coding the source for linking among pages. Ask them to always specify the page name in the link. For example, ask them to code

```
<a href=”/excavation/index.html”> not <a href=”/excavation/”>
```

However, even with excellent compliance from your content production and application developer staff, visitors can still enter inconsistent variations. Though it would be cleaner from a data perspective not to recognize all the variations using various Web server configurations, it would be detrimental to the user experience and lose visitors. So you want your Web server to continue to find the home page even if the visitor doesn’t supply that part of the page name.

Using the intervention point, before output, examine the value for your requested file variable. If it ends with a slash (/), append the appropriate standard home page name used at your site. You can expand this to examine for just a domain name with no slash and also append the appropriate home page name there. If your site does not maintain a standard home page name between sections and directories, create a nightly pre-process that traverses your content file structure (or content management data base) to produce a format that tries to identify the home page for each directory based on a series of possible home page names defined by you. Use that format to append the appropriate home page name for that specific directory. The source for the format would be similar to this.

```
proc format library=control;
value $hmepage
'excavation/'='index.html'
'excavation/shovels/'='index.html'
'excavation/explosives/'='default.htm'
'treasuremaps/'='default.htm'
'treasuremaps/sunken/'='index.html'
'treasuremaps/buried/'='index.html'
other=’ ‘;
run;
```

The `control` libname stores the format in one of the configuration areas of the Web data mart. You should only change the requested file name for successful requests. If you change it for pages requested in error, then you will mask any systematic errors occurring on your site.

```
/* only requests considered page views are candidates for changes          */
/* because we don’t want to alter page names containing errors             */
if page_count = 1 then do;
   /* get the filename part */
   filename = reverse(substr(reverse(trim(requested_file)),
                      1,indexc(reverse(trim(requested_file)),"/")));
   /* get the directory part */
   dirname = reverse(substr(reverse(trim(requested_file)),
                     indexc(reverse(trim(requested_file)),"/")));

   /* when no filename part exists, use the format to determine */
   /* what the appropriate home page name is and tack it on     */
   if filename = '/' then filename=trim(filename)||put(dirname,$hmepage.);

   /* solve issues with remaining filenames of only /           */
   if filename ne '/' then filename=substr(filename,2);
   if filename='/' then filename='';
   requested_file=trim(dirname)||trim(filename);
end;
```

SPIDERS, ROBOTS, AND CRAWLERS

Unlike referrer spam that is sometimes automated (see the "Referrer Spam" section), you do not want to deny access to these automated entities because their access to your site results in the inclusion of your content in search engines, blog mentions, and other automated inventories of Web content. You might want to limit access by these automated visitors to certain sections of your site or for only those automated visitors that behave properly (don't overly tax your server resources). That decision-making and the resulting implementation rest with your Web server systems administration team. Given that it is a necessary evil to allow these visitors into your site, what can you do?

There is a tremendous difference in the way human visitors and automated visitors behave on your site. First of all, automated visitors **never** buy anything! You can't convince them of the value of your offerings, and you can't help them solve their customer service problems. So you don't want to make decisions about your site based on any data representing the activity of automated visitors. If you believe the average time on your site is one minute and that visitors are consuming, on average, 50 pages of your content, you will be making different decisions about content and navigation than if you find average time on your site is five minutes and that visitors are consuming, on average, five pages of your content. However, there are at least two legitimate uses for data representing the behavior of automated visitors.

- Having a total picture of the load request on the Web server. This is of more interest to the systems administration team than your marketing or content staff. While providing "total-use" reporting is a valid use for SAS Web Analytics, it is typically secondary to more market-driven and customer-focused analysis available from SAS Web Analytics. Total-use reporting should be addressed separately.

- Learning more about the behavior of search engines on your site as a possible aid to understanding ways to optimize your site. Analyzing this activity could be helpful as an addition to reading the literature about site optimization for search engines to be able to verify changes in behavior as a result of optimization efforts. This endeavor is separate from analysis to enhance your Web site for human visitors and should be addressed separately.

Many well-behaved robots use an easily identifiable user agent string. However, there are robots that can be difficult to recognize. Use a well-maintained robot identification list, such as the database on The Web Robots Pages (www.robotstxt.org/wc/active.html). Supplement that list with spiders that are detected during your daily processing. Periodically reevaluate the entries in your spider list to remove any that start to be used by legitimate, human visitors.

You can maintain a list of clients (robots, spiders, crawlers, creepers, etc.) using the Web data mart configuration data set or via source code. Use the Filtering tab in the Web Data Mart Properties window to edit the spider configuration data set or edit the "wbspider" data set directly. Alternatively, you can easily maintain your spider list in a macro that is called "after input" intervention point. This gives you the opportunity to remove these records at the onset of processing. Here is a sample.

```
%macro wa_remove_agents (action=%nrstr(delete;));
  /* set value of user agent to lowercase for ease of handling */
   user_agent_wa=lowcase(User_Agent);

   /* list user agents here */
   /* Google's Web crawler */
   if index(user_agent_wa,"googlebot") > 0 then &action;
   /* Inktomi's Web crawler */
   if index(user_agent_wa,"slurp") > 0 then &action;

   /* continue with long list of individual entries here */

%mend wa_remove_agents;
```

With this macro, you can perform a series of statements as the action. For example, the macro call could be:

```
%wa_remove_agents (action=%str(do; page_count=0; spider=1; end;));
```

The default action with the macro call, `%wa_remove_agents`, is to delete the record.

You can also take advantage of the use of SAS®9 Perl regular expression functionality. Like the macro call (not the macro code), this code is placed directly into the “after input” usermod.

```
/* Deletion of spiders through SAS 9 Regular Expressions                    */
/* spiders = the regex pattern to match on.  Update this field with new spiders. */
/* pattern = compiled regular expression for performance                    */
/* position if pattern match found position will be > 0                     */
/* delete record if position is greater then 0                              */
/* will capture a majority of spiders                                       */
/* update list as needed                                                    */

/* retain compiled regex variables for each pass */
/* increase pattern numbers as list increases    */
retain pattern_1 pattern_2 pattern_3;

if _n_ = 1 then do;
/* Generate list of spiders                                                 */
/* Currently prxparse has a limit on length of characters so create a separate   */
/* variable after about 8 regEx expressions                                 */
    spiders_1 = '/slurp|bot|spider|archiver|seek|crawl|scooter|answerbus/i';
    spiders_2 = '/teoma|link|infoseek|spy|gulliver|harvest|iltrovator/i';
    spiders_3 = '/inktomi|mercator|search|go\!|incywincy|larbin|look/i';

/* Use prxparse to compile the perl RegEx       */
    pattern_1 = prxparse(spiders_1);
    pattern_2 = prxparse(spiders_2);
    pattern_3 = prxparse(spiders_3);
end;

/* Actual spider matching and deletion          */
    position_1 = prxmatch(pattern_1, user_agent);
      if position_1 > 0 then delete;
    position_2 = prxmatch(pattern_2, user_agent);
      if position_2 > 0 then delete;
    position_3 = prxmatch(pattern_3, user_agent);
      if position_3 > 0 then delete;
```

There are two important notes about this code:

1. Be sure to set only the spider and pattern values once (when _n_ is 1).
2. Remember the retain statements.

New automated visitors will continue to appear on your site. The SAS log of your ETL processing will contain warning messages identifying visitors that have exhibited characteristics (too many views, for example) that create the suspicion that the visitor is a spider. After the initial implementation, I recommend setting up an ancillary process to filter the SAS log of your ETL processing to produce a “suspected spider” list. Investigate the entries on the list by resolving the IP address to examine the domain name of the visitor (www.DNSstuff.com). If you judge the visitor to be a spider, you can update the spider configuration data set, the special client configuration data set, or your filtering source code to exclude future requests from this visitor.

OTHER SPECIAL CLIENTS

There might be other identifiable sets of visitors that you want to filter from inclusion in your Web data mart, but—like search engine spiders—you do not want to exclude access from the Web site. For an Internet site, you might be interested in excluding the activity of your own employees since, even though we assume they are human, they might have extremely different behavior on your Web site than the visitors you are attempting to understand and influence.

The most straightforward approach is to add all of the IP addresses (or range) used by your organization to the special clients configuration data set described in the section “Referrer Spam.” This would address remote connectivity via VPN, but would not exclude employee access to your Internet site from public locations such as airport kiosks.

CONFIGURATION CONSIDERATIONS

AUTOMATED PROCESSING

Your Web analytics implementation begins with the transfer of daily (or periodic) Web server access logs to a processing location. At a minimum, ETL will be run against those logs. At successful completion of the two ETL steps, the Web access logs will be removed from the processing location. In a fully automated environment for daily/nightly processing, several other processes also are scheduled. This list is a skeleton for processing steps that you should consider automating.

- Access log pick up. Depending on the environment, copy or FTP the closed access logs to the location from which they will be processed.
- Archive a copy of the access logs (optional). If you plan to maintain a copy of the raw access logs, place a copy in your archive location. This should not be the same as your processing location. I recommend compressing the logs with the appropriate compression tools for your operating system.
- Check to see if the expected number of logs are in your processing location (optional). Do you have a set number of logs expected daily? Is there a set naming convention expected for those logs? If an incorrect number or name for logs means there has been a processing error, stop the processing.
- Gather any external data sources (optional). Are there formats you use for categorizing or cleaning the data? Are there other data sources you use to identify additional classification of your data (such as departmental affiliation for intranet sites, IP resolution for geographical locations)? If so, run any jobs to prepare/update that data. Again, if failure on these jobs will impact the successful processing or cleaning of the log data, check for errors, and stop processing.
- Run the first step of ETL to produce the interim SAS data set with detailed records (weblog detail in Figure 2). The sample job supplied with your SAS Web Analytics software (`daily.sas`) contains three main steps. The first two (both `%edataetl` calls) perform the first step of ETL to read the Web server logs, perform standard filtering, and custom filtering and mapping. Although there is an error check in the code between the second and third step (`%if &wbrc = 0`), I recommend breaking this job apart and creating more robust error-checking that occurs prior to the analytical ETL (`%waetl`).
- Check for anomalies in the SAS log from `%edataetl` (optional). Search for ERRORS. If found, stop processing.
- Run the second step of ETL to further transform, augment and load your Web data mart (`%waetl`). This process uses the detailed SAS data set created by `%edataetl` to produce dated and detailed data sets per day. One contains an observation for each retained request, and the other contains an observation representing each visit on the site. Those data sets are then extensively summarized to produce a variety of data sets used for further analysis, reporting, and prediction.
- Check for successful completion and remove Web logs from processing location. Because SAS Web Analytics allows the flexibility to process logs in any date order or to add more logs for a day for which logs have already been processed, it is important that processed Web logs be removed from the processing location to prevent inadvertent repeated processing.

WEB DATA MART CONFIGURATION

In addition to the special client-and-spider configuration data sets, you can supplement configuration data sets that identify browsers and operating systems used by visitors to your site. These data sets are populated initially with the most commonly encountered browsers and platforms, but can be customized to further match the visitors for your site. Other default configuration settings address such issues as a timeout period by which the end of a visit is identified and characteristics of the URLs on your site. If your site experiences unusually short or long visits, you can modify the timeout setting. The industry standard is 30 minutes. If your URLs have unusual characteristics (exceptionally long URLs, use non-standard query string or cookie delimiters, for example), you have options for handling a variety of these issues.

Two categories of configuration settings pertinent to the retention of data and summaries are:

- The number of date-based data sets for views and visits (configuration settings start with wab_num)
- The length of time to maintain historical data in summary data sets (configuration settings end with _history)

Date-based data sets: For each day, a date-based data set is created that contains an observation that represents each request (that you have not chosen to exclude through filtering) made to the server. These requests include both the successful requests that resulted in a view to a page via the browser and requests that failed and resulted in an error message in the browser (such as file not found). This data set is referred to as the *detail data set.* The naming convention for these data sets is detail_yyyymmnn. Don't be confused about the existence of weblog_detail (the interim data set produced from the first ETL step) and detail_yyyymmnn. Weblog_detail (actually called "weblog_detail_1") is produced with each ETL run with the same name and it is just the interim result of the initial extraction. Only one detail_yyyymmnn is produced per day and it is the permanent data set in your Web data mart with individual records for each Web request. If you run multiple ETLs that process web logs for the same day (or past days), the records will be placed in the appropriate detail_yyyymmnn based on their timestamp date. A second date-based data set contains an observation that represents each visit to the site. It is an initial summarization of the detail data set in that a record reflects data values that show characteristics describing the entire visit (which browser was used, the first page viewed, the last page viewed, the length of the entire visit, the number of pages viewed for the entire visit, and so on). This data set is referred to as the *session data set*. By default, there are 30 (detail) and seven (session) data sets retained. It is unusual to need access to these data sets. So there is rarely any need to increase this default setting. If disk space is not an issue, I recommend setting up an ancillary process to periodically (monthly, for example) copy and compress these data sets to an archive area prior to their removal from the Web data mart.

Summary data sets: Data sets representing activity summarized for days, weeks, months, quarters, and years are created. Parameters that specify the number of units for each of these time periods that data will be retained can be adjusted (called `WAB_timeperiod_IN_HISTORY`). The default settings are generous (185 days, 52 weeks, 36 months, 12 quarters, 3 years). Note that unless adjusted, the default for individual days does not cover an entire year. If your intention is to keep data summarized for each day for an entire year, you need to adjust the `WAB_DAYS_IN_HISTORY` parameter.

In addition, data sets specific for providing pathing analysis (behavior to, from, and between Web pages) and dashboard and scorecard reporting (viewing metrics and trends, analyzing and forecasting performance) have configuration settings to control data retention.

Setting the first day of the week is a noteworthy configuration. If your site maintains a weekly publishing schedule, this parameter is especially effective. For example, if new content is published to your site on Tuesdays, then it would be common to want to report on activity for a Tuesday-through-Monday time period. While you can always select arbitrary date ranges for reporting, by defining Tuesday as the first day of your week, reports based on weekly summaries will be presented covering Tuesday through Monday.

REPORTING AND PRESENTATION

There's a reason that the sections about configuration are so short; there is little required for you to do in order to produce reports, including predictive analysis, using SAS Web Analytics. Out of the box, SAS Web Analytics can help you create a wide variety of reports about activity (traffic) on your site, as well as a dashboard and scorecard.

There are more reports and more numbers available through SAS Web Analytics than can be manageably digested. There is a learning curve associated with understanding the meaning of the data both in terminology and relevance to your site. The SAS Web Analytics Report Viewer is an interactive interface that lets you create and explore complex reports. While this is powerful, it creates an opportunity for the uninitiated to produce reports that will require effort to absorb.

The challenge at the beginning of implementation is to choose the information most critical to your decision-making and focus on that information. Return to the business issue "**GET A GOAL**." If you don't have at least one identifiable business objective for your Web site, you won't have a guide for the most important focus point and the associated metric to evaluate. In the absence of good direction from your marketing strategist teammates, reflect on what you understand to be the mission of your organization or any corporate announcements about the important initiatives for your organization.

For our example, let's return to our business objective of increasing the number of visitors who request further information about the treasure maps you sell. Since you don't do direct sales of the maps from your site, you are interested in collecting information about potential customers so that you can contact them. Your site is set up with introductory information about the various treasure maps available, including the levels of danger or difficulty that map followers might face and the potential rewards. Depending on the level of danger that potential clients indicate they are willing to endure, the clients are guided to a "contact me" form that requires different amounts of contact information.

Here's an index of the relevant pages on the site.

ID	URL	Page Name	Links to
1	www.tldm.com/index.html	The Lost Dutchman home page	2, 3, 4, 5
2	www.tldm.com/moreinfo.html	Greater details about risk and rewards	1, 3, 4, 5
3	www.tldm.com/adventurersguide.html	Special offer for the adventurer's guide	1, 2, 4, 5
4	www.tldm.com/contact_timid.html	Contact me for the Timid	1, 2, 3, 5, 6
5	www.tldm.com/contact_brave.html	Contact me for the Brave	1, 2, 3, 7
6	www.tldm.com/thankyou_timid.html	Thank you for the Timid	1
7	www.tldm.com/thankyou_brave.html	Thank you for the Brave	1

Remember, you are limiting your focus to the most critical information related to increasing requests for information. In this case, reaching a "thank you" page means that the visitor has requested information. Start by looking at traffic trends for the thank-you pages. This information can be found in the page frequency report (Figure 3). But to spotlight the traffic for the thank-you page, create a page-frequency report that shows the traffic for just the pages of interest over time. There is other data relevant to traffic to these pages, but this sample has been modified to include only views and visits, for purposes of discussion (Figure 4).

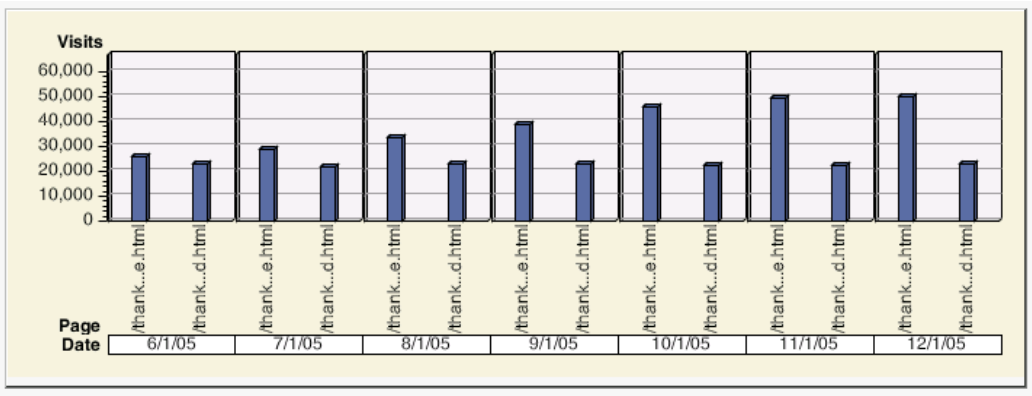

Figure 3. Page Frequency Report for Number of Visits over Time

Date	Page	Visits	Views
6/1/05 (1 to 2 of 2)	/thankyou_brave.html	25,902	25,902
	/thankyou_timid.html	22,989	22,989
7/1/05 (1 to 2 of 2)	/thankyou_brave.html	28,534	28,534
	/thankyou_timid.html	21,777	21,777
8/1/05 (1 to 2 of 2)	/thankyou_brave.html	33,456	33,456
	/thankyou_timid.html	22,809	22,809
9/1/05 (1 to 2 of 2)	/thankyou_brave.html	39,011	39,011
	/thankyou_timid.html	23,001	23,001
10/1/05 (1 to 2 of 2)	/thankyou_brave.html	45,987	45,987
	/thankyou_timid.html	22,546	22,546
11/1/05 (1 to 2 of 2)	/thankyou_brave.html	49,387	49,387
	/thankyou_timid.html	22,314	22,314

Figure 4. Page Frequency Report for Visits and Views over Time

Your report shows you that the use of the “brave” contact-me form is increasing steadily, while the use of the “timid” “contact me” form is only holding constant. Be sure to consider any campaigns or advertising that might affect your traffic. This is the report you initially put in front of your marketing strategist, content producers, creative designers, and executives. But be ready for the questions such as “Why aren’t we seeing increases from the timid miners?” and “Can we increase use even more from the brave miners?”

So you might want to investigate more about the path visitors are taking to reach the “contact me” form and, ultimately, the “thank you” page to find out if there is anything on the Web site that might influence differently the behavior of the groups of miners. This information can be found in the pathing analysis report (Figure 5). There are several paths that can be followed to get to the “contact me” pages. You can use the interactive pathing to present a picture of the paths used to reach each of the pages.

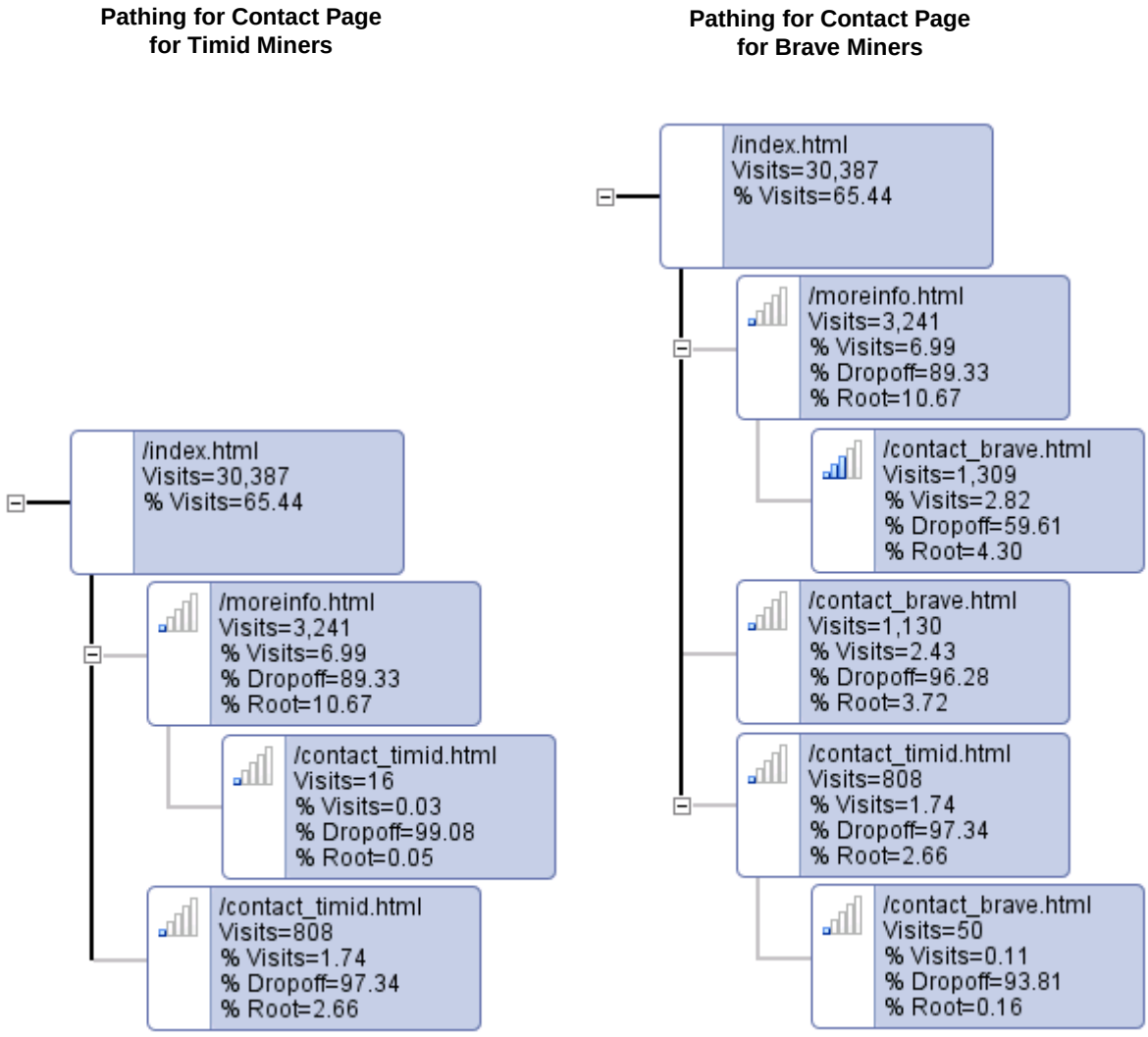

Figure 5. Pathing Analysis Report

You can see from this that nearly all visitors to the “timid” contact page come directly from the “home” page; few come from the “more information” page. A roughly equal number of the visitors to the “brave” contact page came from the “more information” and the “home” page. No one came from the “special offer” page. Interestingly, several visitors went to the “brave” contact me form *from* the “timid” contact me form, but no visitors traveled the other direction. Is there something about the “more information” page that encourages brave miners to proceed to the “contact me” form, but doesn’t sufficiently encourage the timid miners? Is there anything different on the two “contact me” forms that might affect the pathing between the two pages?

Looking at the pages reveals two issues. There is no link on the “brave” contact me form to the “timid” contact me form, but there is one in the reverse direction. Evaluating the content of the “more information” page shows a much greater focus on the excitement of the risks than the assurance of ultimate rewards, possibly holding much more appeal to the brave miners than the timid miners. Making those changes might address the issue of no increases from timid miners. But what about overall increases from brave miners?

You noticed there were no visitors that traveled from the “special offer for adventurers” page to either “contact me” page, even though the “special offer” page was designed to entice brave miners. You check for errors (status code reports) and find the “special offer” page has been producing the client error *404 – file not found* because there is a typographical error in the link to the “brave” contact me page.

This example is meant to convey that the details from your analysis will help you locate land mines within your Web site. These details can guide you to the sources of problems and places on your Web site that are not producing the activity you intend. You have to provide logic, creative design, usability, and compelling content to advance the continuous improvement of your site.

The best way to combat the learning curve about Web analytics related to terminology and relevance to your site is to introduce use of the data gained from Web analysis in manageable amounts and in real-life scenarios. Pulling data out of the SAS Web Analytics Report Viewer that helps identify the source of problems on your Web site to use in alternative presentation styles is an effective way to begin using the Report Viewer. Some colleagues who see these nuggets of insight will ask for direct access to the Report Viewer to continue to mine for additional value.

Although there are others, two features of the SAS Web Analytics Report Viewer that are particularly important for making the analytics results easily available to everyone are:

- The ability to directly access the report viewer with the specifications to reproduce a particular report for any desired time period
- The ability to export Web queries from the Report Viewer to Microsoft Excel.

Reproducing particular reports: The objective is to provide a method for colleagues to redisplay a report, on demand and alter the time period to their specification. Using the Report Viewer, you select the appropriate report. The relevant technique is to use the calendar navigation to select a time period, such as the previous 30 days. Save the URL currently displayed in the browser to use as a link in an email, document, or Web page. When this link is used, the report is displayed with the data refreshed for the current time period. This same technique can be used to produce reports for an entire year (or specific duration), summarized by a period of time (week, month). As time passes and data continues to be added to the Web data mart, each access to the Report Viewer with the saved link will present a refreshed and updated view to the reports.

The advantage of this technique is that it immediately produces the initial report desired, but it also provides direct access to the Report Viewer. This is valuable for the user who needs guidance zeroing in on the specific report needed, but wants to be able to use the Report Viewer for further exploration.

Exporting Web queries: The objective is to provide a familiar interface for colleagues but still have the ability to have data updated as it is refreshed in the Report Viewer. Using the Report Viewer, you select the appropriate report and, as above, you use the calendar navigation to select the desired duration and period for summarization. Instead of saving the URL, use the “Generate URL for Export” icon in the left column of the report. This provides a URL that can be imported as a Web query in Excel. Copy the URL. Invoke Excel. Use the Data menu to import external data by defining a new Web query. Paste the URL into the Address field of the new Web query and import. To maintain the formatting of the original Report Viewer display, use the Web query formatting option for “full html formatting.” You can use any of the features of Excel to enhance the appearance of this report.

The advantage of this technique is that you can present a compact and well-defined picture about a particular issue on your Web site by importing Web queries that represent multiple reports from the SAS Web Analytics Report Viewer. It maintains the power of the Report Viewer because the Web queries can be refreshed either on demand by the user or programmatically on a set schedule. Although this technique and the accompanying process to update the spreadsheets programmatically is somewhat supplemental to direct use of SAS Web Analytics, it is a powerful alternative. A few more details about the process follow.

The steps necessary to programmatically refresh the data produced by the Web query in an Excel spreadsheet involve setting up a SAS job that is scheduled to run once a day (or at an appropriate interval for your data) that opens the spreadsheet, refreshes the data, and closes the spreadsheet. Sample code follows. It will need to be customized for your environment.

1. Create an Excel macro to perform the data refresh. These instructions are specific to Microsoft Excel. They are not part of SAS Web Analytics functionality. If you are familiar with using Excel to create macros and know other ways to accomplish these tasks, follow the methods known to you or use the Excel help as a supplement.
   a. Open your spreadsheet.
   b. Make sure the External Data Toolbar is displayed. If not, use the Excel Standard Toolbar → View → Toolbars → External Data.
   c. From the Excel Standard Toolbar → Tools → Macro → Record New Macro.
   d. Enter a macro name, such as Refresh.
   e. Store the macro in "This Workbook".
   f. Click the **OK** button.
   g. Click the **Refresh All** icon on the External Data Toolbar.
   h. From the Excel Standard Toolbar → Tools → Macro → Stop Recording.
   i. Check the contents of your macro. From the Excel Standard Toolbar → Tools → Macro → Macros.
   j. For the macro name you just created, click the **Edit** button.
   k. The contents of the macro should be similar to the sample below. Differences might include a different macro name (if you didn't use Refresh) and different comments regarding creation date.

```
Sub Refresh()
'
' Refresh Macro
' Macro recorded 9/16/2005 by Jerry Hosking
'

'
    ActiveWorkbook.RefreshAll
End Sub
```

   l. Exit Microsoft Visual Basic.
   m. Exit Excel.

2. Create a SAS job that will run the commands to do the following. Sample code follows. You need to customize the code for your environment.
   a. Start Excel.
   b. Open the spreadsheet.
   c. Refresh the data (using the Excel macro).
   d. Save the spreadsheet.
   e. Close the spreadsheet.
   f. Quit Excel.

```
/* set options needed to issue system commands */
/* and control process scheduling */
options noxwait noxsync;
/* invoke Excel */
/* customize to specify the location of your */
/* Excel executable */
x '"C:\Program Files\Microsoft Office\OFFICE11\EXCEL.EXE"';

/* let SAS wait until Excel is started */
data _null_;
x = sleep(5);
run;
/* set up communication to issue commands to Excel */
```

```
filename commands dde 'EXCEL|SYSTEM';
/* issue the command to open your spreadsheet */
/* customize to specify the location of your spreadsheet */
data _null_;
file commands;
put
'[OPEN("H:\web_analytics_reports_folder\web_analytics_report_1.xls")]';
run;
/* let SAS wait until spreadsheet is opened */
data _null_;
x = sleep(5);
run;
/* issue the command to run the Excel refresh macro */
/* you previously created */
/* customize to specify your spreadsheet name and macro name */
data _null_;
file commands;
put '[RUN("web_analytics_report_1.xls!Refresh",False)]';
run;
/* let SAS wait until refresh is accomplished */
data _null_;
x = sleep(15);
run;
/* issue commands to save and close active spreadsheet */
/* then quit Excel */
data _null_;
file commands;
put '[SAVE]';
put '[CLOSE]';
put '[QUIT]';
run;
```

3. Schedule the SAS job to run at the appropriate time interval.

CONCLUSION

The objective of this paper is to identify and defuse land mines that present barriers to a successful Web analytics implementation. While there are many technical issues and configuration considerations that are important, the business issues and the ability to examine the content of your Web site in light of the insights provided by the analytics are equally important. Your work to make the data portray an accurate picture of the behavior of the visitors that are significant to you will translate into meaningful revelations from your Web analysis and reporting. Acting on those revelations to continuously improve your Web site will produce improved results.

REFERENCES

Alberta Government. “Definitions”. *Environment*. October 25, 2004. Available http://www3.gov.ab.ca/env/air/Info/definitions.html.

Apache. “Log files. “ *Apache HTTP Server Version 1.3 Log Files*. (accessed January 13, 2006). Available http://httpd.apache.org/docs/1.3/logs.html.

Korpela, Jukka. “Methods GET and POST in HTML forms - what's the difference?” *IT and communication*: Yucca’s fee information site. September 28, 2003. Available http://www.cs.tut.fi/~jkorpela/forms/methods.html.

Sullivan, Craig. “RE: [webanalytics] Re: How is your web metrics program structured?” *Web Analytics Forum*. May 18, 2005. Available http://groups.yahoo.com/group/webanalytics/message/2492.

Value Based Management.net. “The Deming Cycle.” September 18, 2005. Available http://www.valuebasedmanagement.net/methods_demingcycle.html.

Wikipedia. “Continuous improvement.” *Wikipedia: The Free Encyclopedia*. October 3, 2005. Available http://en.wikipedia.org/wiki/Continuous_improvement.

Wikipedia. “Referer Spam.” *Wikipedia: The Free Encyclopedia*. December 19, 2005. Available http://en.wikipedia.org/wiki/Referrer_spam.

www.DNSstuff.com. Accessed January 13, 2006. Available http://www.dnsstuff.com/.
(Note: For example, http://www.dnsstuff.com/tools/whois.ch?ip=149.173.5.120 shows SAS Institute Inc. Before using a Web site for IP resolution information, read the terms of use.)

RECOMMENDED READING

SAS Institute Inc. 2005. *SAS Web Analytics 5.2: Administrators Guide.* Cary, NC: SAS Institute Inc.

SAS Institute Inc. 2005. *SAS Web Analytics 5.2: Users Guide.* Cary, NC: SAS Institute Inc.

Web Analytics Forum: http://groups.yahoo.com/group/webanalytics/.

ACKNOWLEDGMENTS

The developers of SAS Web Analytics and SAS IntelliVisor were critical to the implementation and my use of SAS Web Analytics for SAS Institute. I thank Caroline Bahler, Robert Levey, Jonathan Polito, and Frank Roediger for their assistance and passion for producing a Web analytics solution.

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the author:

Jerry J. Hosking
SAS Institute Inc.
SAS Campus Drive, U2100
Cary, NC 27513
Work Phone: 919-531-7056
Fax: 919-677-4444
E-mail: jerry.hosking@sas.com
Web: support.sas.com

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.