

Variable Architecture in Web Analytics

Web Analytics Design

By Paul Legutko

“What do you want to measure?” asks the Web analytics implementation manager. “Everything,” replies the business manager. Great, exactly what a Web analyst wants to hear. But measuring everything and making that information analyzable, useful and practical involves more than passing every bit of information on Web site behavior to a Web analytics tool. It involves organizing that information into a variable architecture that retains the analytical detail but provides the practical means by which to digest it.

Variable architecture is not the same as data architecture. Designing a database and creating data architecture is largely independent of the delivery or reporting mechanism; a SQL query doesn't care how many values a variable has or in how many permutations its values occur. Nor is variable architecture the same as information architecture in Web site design: many kinds of hierarchies - not just informational - might be at play at the same page or action on a Web site. Furthermore, having variables capture each bit of information is not the same as having variable architecture that relates these bits of information to each other and makes it readily digestible to product or marketing managers. Rather, variable architecture describes the way that interesting information is stored in such a way as to preserve the detail of anything analytically interesting but also to allow reporting to be as easy and practical as possible.

Variable architecture and the process of designing it are applicable to any Web site. The model to be followed is simple. First, you identify facets of the Web site that are interesting from both an analytical and reporting perspective. Second, you assign different combinations and permutations of these facets to different destination variables, leaving room for ultimate rollups in each of these facets. Not every combination of facets will be interesting, but a variable architecture will emerge that will not only preserve the details but also give insight into the different ways these facets interact.

I say “facets” of the Web site to keep it one level above specific content or site functionality. Facets are the where, when, why and what of Web behavior. What are visitors doing, where and when are they doing it and why? “Page” is one such facet that most of the time is an out-of-the-box kind of variable. But on a Web 2.0 site, page might be less interesting than other kinds of facets, such as widgets, Flash screens, movies or other dynamic content. These facets are easily identifiable by the interested stakeholders - managers of marketing, content, merchandizing or Web site design - who want to know various kinds of information about Web site activity.

Each facet can have thousands of different values. It must be remembered that values are different from variables or metrics. A common mistake in Web analytics implementation is to assume that for every specific piece of information, one needs a separate variable. Not true: every specific piece of information needs to relate to a common facet of a Web site, which is then placed within the variable architecture of the Web measurement implementation.

Suppose, for example, that you have an e-commerce site that sells music CDs. One interesting measurable facet is genre: classical, country, pop and hip hop. The wrong thing to do is assume you need four custom variables: classical, country, pop and hip hop. These are not variables; they are values in a single variable - genre. Comparing each to each other analytically is apples to apples. But casting the net wider, there are other facets, like genre, that may be of interest: album title, display prominence (like a featured box on the home page), page (or “area/moment of interaction” in Web 2.0 sites without “pages”), the action itself (“buy now” versus “view reviews”), the price of the album and other e-commerce aspects (cost of goods sold, units, orders, etc.), global site navigation usage and internal search terms. Each facet can contain thousands or hundreds of thousands of values, but they should all be relevant to the same facet.

The moment of measurement occurs when a visitor does something - anything - on a site. At that moment, the interaction of these facets should be captured to preserve both the detail and the rollup. As in particle physics, not every action on a Web site involves every facet, but most involve several. The variable architecture now comes into play in how the values of the destination variables are populated:

Relevant facets at the moment of interaction: A, B, C

Variable 1: A_B_C

Variable 2: A_C

Variable 3: A_B

Variable 4: B_C

Variable 5: A

Variable 6: B

Variable 7: C

In the example I mentioned, suppose a visitor clicks “purchase” (as opposed to “view details” or “view reviews”) for the Beatles’ White Album when it is displayed in the home page “featured CDs” box. The three facets here are prominence, action and product. Thus, the values sent to destination variables would be:

Variable 1 (prominence, action, product): hpfeatured_purchase_beatleswhitealbum

Variable 2 (prominence, action): hpfeatured_purchase

Variable 3 (prominence, product): hpfeatured_beatleswhitealbum

Variable 4 (action, product): purchase_beatleswhitealbum

Variable 5 (prominence): hpfeatured

Variable 6 (action): purchase

Variable 7 (product): beatleswhitealbum

At first glance, this matrix seems like overkill, but now consider the reporting and analytical potential. Analytically, variable 1 can be crunched in a statistics program for multivariate analysis; Web site designers interested in placement can look at variable 5 or variable 2; e-commerce or marketing managers can easily report on variable 6 or use it for campaign optimization; merchandizing managers can use variable 4 or variable 7. To get these kinds of reports using only variable 1 would be time-consuming, involve extensive segmentation or

filtering, and, in the end, would not be practical (and thus not used). Measuring everything happens using only variable 1, but making everything practical happens with variables 2 through 7.

A Web site might constitute a dozen such facets - and, as I mentioned, Web 2.0 sites would have to consider as a facet what was once called a page. Obviously, such variable architecture requires many custom variables, but why else does a solution provide 100 custom variables out of the box? Custom metrics in enterprise-level Web analytics solutions are notoriously underutilized. But with a rigid variable architecture in place, measurement managers can take full advantage of the customization built into such solutions.

The most common facets in variable architecture are the where, what and why of visitor behavior. "Where" refers to a page or section of a page such as home page or left navigation, for example. "What" refers to a concrete item such as a product, article, video or offer. "Why" refers to what the visitor wants to do, perhaps "buy now," "play video" or "see larger picture." Any action on any Web site usually involves all three items, and applied variable architecture preserves these facets in a practical but analytically powerful manner.

One point to emphasize is that the base variable (variable 1 above) is actually necessary. Even though this variable might be unintelligible for reporting purposes and contain hundreds of thousands of permutations, at least the data is there. If necessary, it can be extracted and put into a statistical application or SQL database. The last thing a Web implementation or measurement manager wants to say is, "We don't have that detail." The brilliant thing about most Web analytic tools is that once the information is in the database, it can be extracted somehow. But it has to get there first. Using only rollup variables but leaving out the detail might be good for specific reporting, but not for follow-up analysis or investigation.

Technical teams might protest that passing so much information into a Web analytics tool might slow down site performance. These tech teams might also balk at the unanticipated time or resources needed to implement these measurement tags on a site, given that many of these facets occur outside of a page load where most traditional Web measurement tags get called; measuring many facets might require Flash integration or on-click handlers. Obviously, some compromise might be required, but a Web analytics implementation manager should also bear in mind the often considerable time one saves in reporting when a robust variable architecture is in place. Providing reports to marketing, merchandizing, executive or product management teams only using the detailed variable 1 in my example would require extensive filtering, number crunching and data extraction that would make reporting far more manual and time-consuming than a quick, automated solution available in the Web analytics interface or Excel integration.

Variable architecture is the intersection of data storage and data delivery. It satisfies analysts in preserving the detail, but it also gives managers a quick and seamless way to report site activity. It is easy to create, simply involving the identification of interesting facets of a Web site. By keeping the distinction between values and variables, it keeps all the apples in one bucket and the oranges in another, but provides the fruit basket too. It takes advantage of the flexibility of enterprise-level Web analytics solutions and is easily transferable to Web 2.0 sites. In short, it should be the central component of a Web analytics implementation design.