

Janice Crotty Fraser

Taking a Content Inventory

There's something interesting happening this year in our industry.

Despite the downturn, major companies are investing in their Web sites.

Somewhere back in 1997, large companies decided that centralized Web development departments were too slow or too controlling to keep up with the rapid innovation that characterized the Web at that time. Soon thereafter, every department had a Web content group that operated more or less independently from others around the company, and had free reign to develop content that seemed right for its section.

Content groups proved both good and bad: On the plus side, lots of useful content was created quickly, sites grew and matured at an astounding pace, and the Web's value became widely understood within these companies. Unfortunately, the sites became sprawling structures with unconnected silos of content that provided little continuity. They failed to provide a cohesive experience for the site's visitors and were expensive to maintain.

The current economic downturn has made it more important to optimize than to innovate. Companies are recentralizing management of some Web functions, and creating a hybrid process for content creation. Departments across the company can still generate content, but the holistic user experience—architecture, design standards, and basic functionality—is managed centrally.

The push for centralization has two primary drivers: operational efficiency and user experience concerns. Content management systems (CMSs) improve efficiency by processing all content through a single storage and retrieval system. Instead of supporting an array of systems, the technical team can focus on maintaining and extending one plat-

form for the whole company. Companies are addressing user experience concerns by reworking their sites; overhauling everything from the navigation to a site's fundamental organization.

Despite the temptation to start making changes, proceed with caution. Before you even consider a CMS migration or a re-architecture project, you'll need to take a content inventory. These projects affect vast amounts of existing content, some of which may be redundant or outdated. They're different from the typical projects that architects and developers have faced in years past, and they require new tools.

The Post-Downturn Architect's Tool

After years of boom and sprawl, many Web sites resemble L.A. County more than an organized system of resources—you'd need a really good road map to find your way around. Before an information architect can hope to reorganize your site to improve the user experience, someone needs to understand it—the scope, nature, and context of all those piles of content. In most companies, no one person is familiar with everything that's there.

The basic task of re-architecture is answering the question "What goes where?" The content inventory answers the "what" part of the question, so that you can get to work arranging the "where" using other architectural techniques.

A content inventory is a methodical review of a Web site's content. It's essentially a research project, and the information you glean from conducting it is sometimes as important as the deliverable you create at the end. There are various kinds of inventories that you can use alone or in combination to reach different ends. Three basic types of inventories cover most cases:

online resources

Take Stock

More ideas on how to put your site in order.

Argus' Resources for Information Architects
argus-acia.com/ia_guide/index.html

Jesse James Garrett's Resources for Information Architects
www.jjg.net/ia

A survey is a high-level review of core site pages, usually taken at the beginning of a project. Surveys help you understand the scope and nature of the material—the type of content, what topics it covers, and so on. At the end of a survey, you should have a clear understanding of the major chunks of site content. You can use the survey on its own, or as a launching point for other inventories. I usually find it helpful to structure the survey as a miniature version of a detailed audit.

A detailed audit is a comprehensive, page-by-page site inventory. When complete, this audit lists every page by name and URL, assigns it a unique number to identify it, and lists major attributes of the page that will eventually form part of the important meta data. Often, architects find it easier to begin the detailed audit by doing a quick survey to flesh out a basic framework before beginning the page-by-page site review. The completed audit is useful during migration to content management systems.

A content map is a visualization, a simple illustration of the site's major content components. Resist the urge to arrange components by their current location within the architecture. Instead, group them to reflect the most important user and business objectives. Content maps are the most powerful of the three tools for understanding the big picture, and they can be derived either from surveys or from detailed audits.

Quality inventories must be accurate, consistent, and thorough. If you take inventory with attention to detail and completeness, the end result becomes a solid basis for future architecture and migration work. If sections are missing or mishandled, the entire inventory loses credibility—and this isn't the sort of task that you want to re-do.

Setting Up

Performing surveys and detailed inventories involves essentially two steps: Set up your file, and gather the data. The file templates for a survey and a detailed audit look virtually identical. They differ only in the amount of detail that you record for each page, and the number of pages that you review. In short, the survey records some information for a sampling of pages, while the detailed inventory records all information for all pages.

You can set up the file in any spreadsheet or database application: Excel, Access, FileMaker Pro. I usually use Excel because it's so widely known that I can feel comfortable handing the files off to clients or coworkers without worrying about whether they have the application or know how to use it.

In the Excel file, every row corresponds to a page on the site, and every column is a piece of information about that page. The data that you'll want to record for each page varies from project to project, but there are some good standards with which to start.

There are three general types of data for each page: identification data, such as page title and URL; content data, which describes the page type and subject matter; and management data, which may include the content owner or producer, and flags for calling attention to stale content that should be removed from the site.

While the pertinent information varies according to the needs of your project, the following is a basic set of data fields that you can use. (These have been adapted from a methodology I learned from my business partner, Jesse James Garrett, author of *jgg.net*.)

Link ID. In my audits, I give every page on the site a unique ID. It's a minor annoyance, but a major benefit. With the link ID, you can reference pages with confidence. Referring to pages by URLs, which can be quite long, becomes cumbersome. By saying "look at item number 5.3.6.1," everyone can flip to that page in the inventory and be certain that you're talking about the same piece of content.

To create the IDs, I start by giving every page on the site-wide navigation its own number. Home, for instance, stands alone at the top level of the site. Its number is 1.0. At the next level, you might find About the

Company, Products, and Customer Service. These would be numbered 1.1.0, 1.2.0, 1.3.0, respectively. Within the Products section, the Applications top page would be 1.1.1.0 and the Service Products top page would be 1.1.2.0. If there were five Service Products content pages below that, they would be 1.1.2.1, 1.1.2.2, 1.1.2.3, and so on.

Pages with subpages get the .0 suffix, while pages without children don't. This way, I know at a glance whether a given page has subpages. This also lets me use the Excel autofill feature to generate page IDs for the subpages in that section. To use these, I simply click on the parent page ID cell and drag down the column to fill in the sub-page ID values.

As you build the inventory, every time you step down a layer in the navigational hierarchy you add another dot and digit. Over time, this numbering scheme instantly reveals both the breadth and depth of a page's location within the site. In some sections you'll find that you have eight or ten dots (meaning that it's very deep) and in other sections you'll find digits as high as 15 or 16 (meaning that it's broad). For further graphic representation of the hierarchy, you can use Excel's indent feature to inset sub-page IDs.

Link Name. In most cases you can use either the HTML page title or the link text within the `<a href>` tag to give you the link name. I usually find that one is more reliable than the other, depending on the site. Some sites use the same page title on multiple pages, but provide meaningful names in the actual link tags. No matter where you glean the information, your goal is to collect the data in the same way for every page. So look it over, make a decision, and stick with it throughout the project.

URL. The URL and the link name can often be captured by a so-called spider or Web crawler program. These programs can give you a great head start on a detailed inventory, but they aren't a panacea. The goal of the inventory is to produce a document that's meaningful to humans and represents the perceived architecture. If you use a Web crawler, review and edit the results manually, as the Web crawler rarely captures URLs in a way that follows the architecture.

Content Type and Document Type. These two fields describe the content. Content type isn't the same as topic—it tells you what kind of information it is, not what the information is about. For instance, marketing information, data sheets, technical specifications, and customer stories are all content types. You must decide on a complete set of possible types before you begin a detailed audit. This gives you a controlled vocabulary—a fixed set of values from which you can choose to fill the field. The document type field is similar, telling you what kind of document you're dealing with: paragraphs, a list, a form, a white paper, and so on.

By using a controlled vocabulary, you can begin to identify all pages of the same type in your site.

Topic. This field describes what the content is about. This isn't a standard values field, but rather an open field that you can fill with any words that describe the content topic.

Management Fields. These are the most open fields, and you can use any that help you in your project. In past projects, I've used producer, content owner, user type (the intended audience), company type (customer, partner, and so forth), facets, frequency of update, and outdated flag.

Cell Format Conventions

Because consistency is paramount, and repeating work is painful when you're creating a detailed inventory, establish cell-formatting conventions before you begin. As before, this is one reason why it's best to start a detailed inventory by doing a survey. The survey gives you an opportunity to quickly review the issues you'll encounter down the line and decide on a workable strategy.

As I move through the inventory, I mark redundant content and cross-links by shading the link name and URL fields a light gray. I often use cell formatting tricks with color and indentation to illustrate the level under which information falls. For example, I indent the ID and title cells of child pages. In addition, I mark top-level links with yellow across the whole sheet; second-level with green across the first two cells only. Lower-level links I leave plain and indented to indicate their level; and I often include bracketed and italicized hierarchy notes in the URL field.

Filling in the Survey

So now you're done setting up. Making the decisions about which fields to include is half the battle. For surveys, you won't need to gather all of the information you'd probably want in the detailed audit. At least plan to capture link ID, page name, URL, content type, page type, and topic. Because this is your first review of the site, you won't have established a list of values for content type and gauge type yet. Don't worry, that's partly what the survey is for.

Browsing the site and filling in information can be tedious, but it never fails to inform. You want to follow a broad selection of links to capture information about the major site sections. Look at the top pages and a variety of content pages in each section. As you fill in the spreadsheet, you'll be sketching out the major features of the site. While it won't show every page on the site, the completed survey should show every major content component. For a large-scale Web site, expect to spend about 40 hours on a survey.

As you work on the survey, you can make a list of values for the fields that require controlled vocabulary, including content type and page type. When you've finished the survey, you'll have a draft list to circulate among the project's major stakeholders. Together, you can refine and edit the list until it's fairly complete before you begin the detailed audit. Most sites have fewer than 25 content types, and fewer than 15 page types, though these numbers can vary widely.

Mapping the Content

Once I've finished the survey, I take all of the site's major content components, put each one on a sticky note, and cluster them according to user and business goals. If you have a clear understanding of these goals, this activity is fairly straightforward. This is a good activity to do with a small group of clients or co-workers.

Your cluster groupings can be mapped using Visio, Photoshop, or any number of other visualization programs. I show redundancies across groups by stacking the boxes and coloring them differently. With a three-hour working session and five hours of independent work, you'll have a content map to use as a conceptual reference for architecture decisions. Often, this visualization provides a radically different perspective on the site than a traditional architecture diagram would provide. With a good map, information architects can build stronger relationships between content, identify and eliminate duplications, and re-envision architecture with a view toward breaking out of content silos.

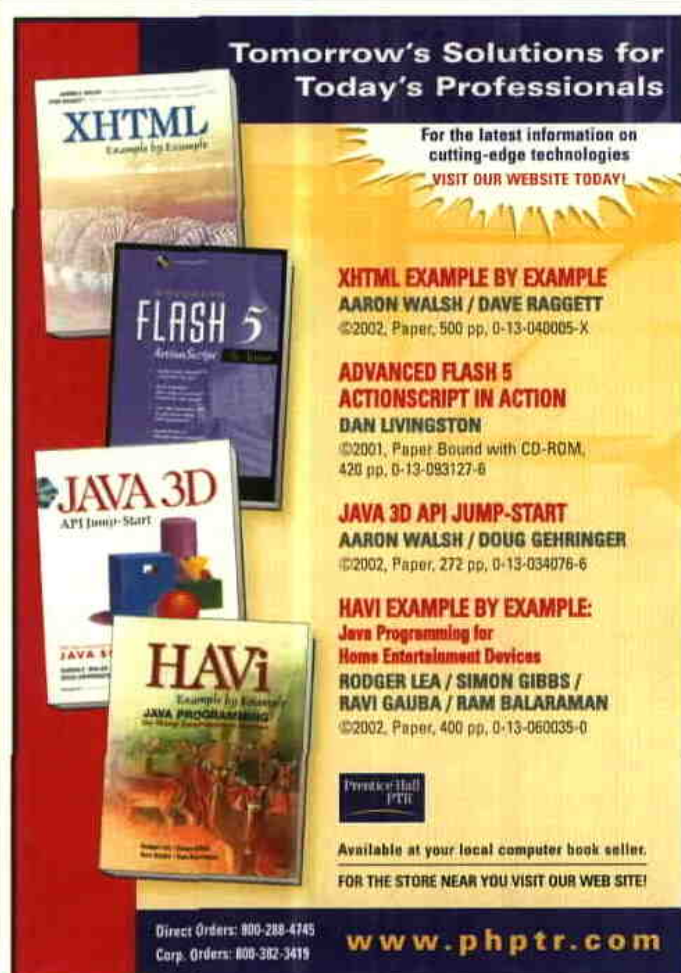
Full Detailed Audit

If you're preparing for migration to a content management system, you'll eventually need to take the framework from the survey and perform a detailed audit. Immediately prior to the migration, you should spend several weeks following every link on the site. Assembling a comprehensive listing of pages makes it possible to track those pages in the move to the new system. While this may feel like tedious work, it will give you a deep understanding of the site content. The greatest benefit of tracking pages this way is that you'll be able to identify and eliminate redundant, outdated, and otherwise ineffective content. The detailed audit is a deliverable with a fairly short life span. Once the migration is complete, it will no longer be useful, so don't be concerned about updating and maintaining the file in the long term.

Rethinking Content Structures

You need to know what you have to work with before you can organize it better. The inventory, above all else, helps you get to know the content deeply; this is as important to a re-architecture as understanding user goals and business goals. Make associations across groupings, identify redundancies, and slice it along a different grain. ><

Janice is a partner with Adaptive Path, a user experience consulting firm. She recently completed a 110-hour content audit with more than 8000 page records. The happy client has the Content Map (printed in glossy color and mounted on foam core) hanging outside the V.P.'s office. You can reach Janice at janice@adaptivepath.com.



Tomorrow's Solutions for Today's Professionals

For the latest information on cutting-edge technologies
VISIT OUR WEBSITE TODAY!

XHTML EXAMPLE BY EXAMPLE
AARON WALSH / DAVE RAGGETT
©2002, Paper, 500 pp, 0-13-040005-X

**ADVANCED FLASH 5
ACTIONSCRIPT IN ACTION**
DAN LIVINGSTON
©2001, Paper, Bound with CD-ROM,
420 pp, 0-13-093127-6

JAVA 3D API JUMP-START
AARON WALSH / DOUG GEHRINGER
©2002, Paper, 272 pp, 0-13-034076-6

HAVI EXAMPLE BY EXAMPLE:
Java Programming for
Home Entertainment Devices
RODGER LEA / SIMON GIBBS /
RAVI GAUBA / RAM BALARAMAN
©2002, Paper, 400 pp, 0-13-060035-0

Prentice-Hall
PTR

Available at your local computer book seller.
FOR THE STORE NEAR YOU VISIT OUR WEB SITE!

Direct Orders: 800-288-4745
Corp. Orders: 800-382-3419

www.phptr.com