

# Building a Large, Successful Website Efficiently through Inquiry-Based Design and Content Management Tools

By Theodore Frick, Bude Su and Yun-Jo An

In May 1994, the first website for the School of Education at Indiana University Bloomington (IUB) was launched: <http://education.indiana.edu>. At that time, the home page had two links – one to our Instructional Systems Technology Department, and the other to the IUB home page.

A decade later, our site now consists of more than 6,000 web pages, and we have received more than 41.5 million page views in the last three years — approximately 38,000 page views per day. Our website is highly ranked in Google searches. Google ranks its search results based on popularity of web pages — a particular web page is ranked highly if other very popular web pages point to it. For example, when searching for the terms, “school of education,” Google ranked our home page in the *top two* out of about 268 million pages that contained these terms in July, 2005.

How did we make this kind of progress in the past decade with a relatively small expenditure of resources? (The “we” in this article refers to the authors. The first author is the School of Education web director and a faculty member with a reduced teaching load. He is supported currently by two doctoral students, the co-authors, who are half-time graduate assistants.)

## Overview

We use an iterative design process that is inquiry-based. We conduct needs assessment of stakeholders, create rapid prototypes and do usability evaluations of design prototypes

to fix problems before we publish our web pages for the world to see.

The web director and one of his assistants have also created an in-house content management system that makes web publishing and maintenance very efficient. The people who use the content management system (called the EdWeb

Page Maker Tools) are staff and graduate students who are already employed in various units within the School of Education, and whose main jobs do *not* involve web maintenance. These web content managers require about two hours of training on the use of our EdWeb Tools. They spend typically no more than one or two hours per week updating the web content for their unit, and are supervised by the same person who supervises them for their non-web duties. Each unit is responsible for keeping their own web content up-to-date. A typical unit is an academic department or program area

(such as language education) or an administrative office (such as external relations or the job placement office).

Extensible Markup Language (XML) provides the means to store the content of our website *independently* from the visual appearance of web pages. Thus, we can change the visual design without modifying the content, and the content managers can change the content without modifying the design. Whenever we make major design changes, we can then republish our website through use of our in-house EdWeb Tools *without editing the content*.

Cascading Style Sheets (CSS) further affect the visual appearance of content in a similar manner. Using CSS



we can change the styles text formatting (such as fonts for headings and paragraphs), and we can modify location and appearance of blocks in our design (such as the page header, sidebar and footer, background, etc.) by editing our style sheets. These changes are rendered immediately throughout the website. No web pages need to be republished. No web content is changed, only its stylistic appearance. We discuss these strategies in more detail below.

## Practical web design: An inquiry-based process

Boling and Frick (2002) have developed an iterative design process that results in useful and usable websites (see <http://education.indiana.edu/~pedagogy/preview>). This design process is inquiry-based and was created and improved through formative research methodology (Reigeluth & Frick, 1999). The major steps in the process are heuristic design activities:

- needs assessment of stakeholders in the website
- rapid prototyping on paper and usability evaluation
- rapid prototyping on computer and further usability evaluation
- building the website technically
- website maintenance

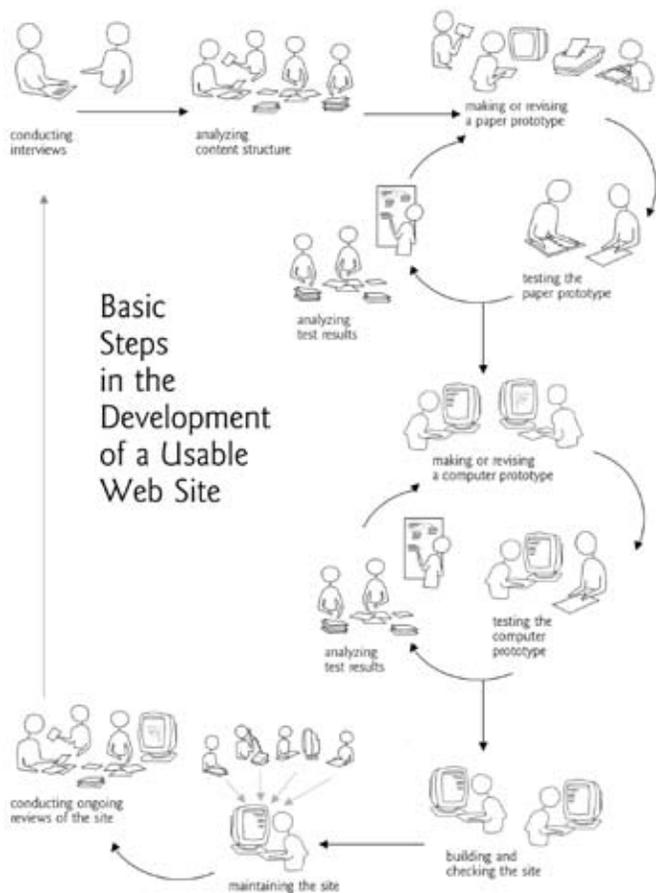


Figure 1. Schematic of the web design process we use

## Assess stakeholder needs, including those of the users

We normally begin the design process by conducting a needs assessment. In the past decade, we have done needs assessment about every three years when undertaking a major redesign of our whole site. We also conduct needs assessment when developing new sub-sites and when redesigning very large and mission-critical sites such as our Office of Teacher Education site.

During needs assessment we often seek multiple sources of information. Typically we interview key stakeholders in the School of Education; these include faculty, staff and administrators. Equally important, we seek information from primary target audiences for our website: prospective students; current students; faculty, staff and associate instructors; and our alumni. Our secondary target audience is K-12 teachers. These audiences are also important stakeholders whom our website attempts to serve. Results from these interviews help us to form goals for our website.

We also interview “gatekeepers” in the School — the people who interact often with prospective and current students, their parents, and other members of the public. We ask the gatekeepers to list the most frequently asked questions they have received over the past year, who asks the questions, when and how they ask (phone, e-mail, walk-in) and what the answers are. Examples of frequently asked questions are:

- “When can I take the Praxis I exam?” (from students who want to get admitted into our teacher education program)
- “What programs do you have?” (from prospective students)
- “Can you graduate in four years?” (from parents of undergraduate students)
- “What are the prospects of getting a job when finished with a degree?” (from students and parents)

Finally, we look at web statistics to see which of our existing web pages are receiving heavy traffic. For example, our home page is visited often, with more than 1.6 million page views in the past three years; and the most frequently accessed EdWeb page link on our home page is “Academic Programs and Departments.” Our search box is used second most frequently (based on statistics tracking how often the IU search engine accesses our customized search results page headers and footers). We also get a very large number of page views on our K-12 resources pages — e.g., lesson plans for teachers, classroom management styles, for teens only, and resources for teaching, reading and social studies.

Next we sort through the data we have gathered. We create a set of index cards onto which we put individual FAQs, goal statements, existing hyperlinks and other statements of need (one item per card). We often end up with several hundred index cards. Then we do a card sort, typically by inviting students or staff who are interested and who have some free

*“This process effectively creates the information architecture for the entire website. The information architecture is not a matter of personal opinion ... but is based on empirical data gained through disciplined inquiry.”*

time to group cards with common themes together rapidly. This rapid-fire card association task is somewhat akin to the matching game, Concentration™. As the piles start to grow, the sorters label them, using common words that appear on cards in each pile. Sometimes this can result in 40-50 piles of cards with handwritten labels, which are each wrapped with rubber bands or clipped together. When the sort is finished, we start grouping the piles into piles of piles with even further commonalities, with the goal of reducing the number of big piles to 10-15 or so, or as few as possible which make obvious sense. For example, some of the big pile labels for the main School of Education site are: “academic programs,” “prospective students,” “current students,” “faculty/staff,” “alumni,” “finding people and places,” “jobs/employment,” “professional resources,” “support services” and “current news and events.” Within a big pile, e.g., “prospective students,” some examples include: “teacher education,” “Why teach?,” “Is the School of Education right for me?,” “programs,” “steps to becoming a teacher,” “how to apply” and “costs and financial aid.”

Furthermore, the labels of cards on the sub-piles become the *elaborators* for the main labels at the top level of the information hierarchy. For example, the elaborators for “current students” include: “advising,” “registering for classes,” and “courses.” The elaborators for “finding people and places” include: “faculty and staff directory,” “contact info” and “maps.” In effect, the labels of the sub-stacks become the second level in the information architecture. And the labels in the sub-sub-stacks become the third level in the hierarchical order.

This process effectively creates the *information architecture* for the entire website, and for a sub-site within the overall site, such as an academic program area or department. If you visit our website at <http://education.indiana.edu>, you can see all labels from the card sort by looking at the main hyperlinks on our home page, and on secondary pages such as the one for prospective students at: <http://education.indiana.edu/prosp.html>. The results of the card sort are directly translated into the names of the hyperlinks for navigating the site. It is very important to note that the specific wording and organization is further refined through usability testing described below, and so the results of the card sort should be viewed as tentative until we have observed how well they work with representative users of the website.

By generating the top-level structure this way, our website is organized to be compatible with the needs of the primary target audiences, which is often different from how people within the institution may think of our organization. This is also a different structure than the administrative organization of the School of Education, which is headed by our Dean, and consists of subordinate departments, each of which houses academic programs and faculty.

The card sorting process typically results in an information architecture that is relatively compact. Without that information architecture we would have approximately 6,000 hyperlinks on the home page, making it highly impractical to use. With the card sort approach, the number of levels in the hierarchy is kept to a minimum: breadth is greater than depth (cf., Shneiderman & Plaisant, 2004). This means that links to home pages for all major units within the School of Education are just one click away from the home page — and that most web pages are not more than two or three clicks down the hierarchy. For example, if we were to put 15 links on the home page, 20 links on each second-level page, and 20 links on each third-level page, the hierarchical table of contents could point to 6,000 unique web pages. Of course, the card sort and resulting information architecture are based on commonality of user needs, not some predetermined number of links per page. But the example illustrates that if top-level navigation pages are well-organized, a fairly large number of web pages down the hierarchy can be accessed with just a few clicks.

An information architecture that results from research can be helpful when justifying the site design to administrators and faculty. The information

architecture is not a matter of personal opinion or preference of any one particular person, but is based on empirical data gained through disciplined inquiry. Moreover, administrators and faculty are stakeholders too, and their needs and goals *are* represented in the information architecture, as well as those of other target audiences such as students, alumni and K-12 professionals.

## **Create rapid paper prototypes and conduct usability evaluations**

Next, the design team creates a set of paper pages, called a rapid paper prototype, containing a sample of the content and structure that is being proposed for the site (cf., Snyder, 2003). The content structure is based on the information architecture we derived from the needs assessment and the card sort. The labels on the cards become the hyperlink names, and the cards in each stack become the basis for naming hyperlinks the next level down, and the next and so on, as described above. The paper prototype is typically put into a 3-ring notebook. We write numbers or letters next to the hyperlinks (underlined text), and then we create tabbed pages with corresponding numbers or letters on the tabs, so that we can simulate web browsing.

We conduct usability tests of the paper prototype by selecting members of the target audience. Usually we need to select only 4-6 members of each appropriate group (cf., Dumas & Redish, 1999; Krug, 2000; Nielsen & Landauer, 1993; Nielsen, 2000). We then observe how these people use the paper prototype to answer frequently asked questions that were identified in the needs assessment. We ask them to think aloud, record the paths they take, determine whether they find the information and find out where they would look for it in the prototype. (A very practical reference on basic principles of web design and usability testing is Krug (2000), *Don't Make Me Think!*)

Occasionally we alter the prototype — sometimes on the spot — and continue to test it until we have identified major problems with the design of the information architecture. If the problems are severe, we attempt to redesign the paper prototype and conduct another round of usability tests. Otherwise, we fix the problems and incorporate the design solutions in our computer prototype, which is the next phase.

## **Create rapid computer prototypes and conduct usability evaluations**

We do rapid computer prototyping next and conduct further usability tests. In composing the computer prototype we attend to some web elements (how users will navigate the site, approximate banner graphics, page layouts, occasional images, etc.). At this point, we are not trying to make the design look completely finished, but to get enough of it working on the web so that we can try it with users. In the past five years we have made this rapid prototyping process fairly easy for ourselves by creating approximate design templates and by using our EdWeb tools to build or re-build an existing website to create the prototype of a new or revised one.

During the usability evaluation, it is our experience that participants will be more likely to notice problems and comment on them if the website *looks* like a prototype, not a finished product. Indeed, some intentional typographical errors, occasional missing items and crude graphics in the design prototype can help encourage users to make comments during formative evaluation (cf., Thiagarajan, Semmel & Semmel, 1974). We also seek feedback from other key stakeholders on the computer prototype at this time. Administrators and faculty tend to be more candid about design issues when they can literally see that the website is still a prototype.

*“Participants will be more likely to notice problems and comment on them if the website looks like a prototype, not a finished product.”*

After modifying the initial computer prototype based on feedback from administrators and faculty, we select a new group of four to six users who are representative of each relevant target audience (cf., Krug, 2000; Nielsen, 2000), and conduct usability tests as described above for paper prototype testing (see Figure 2).

At this time users try to find information under more authentic conditions than with a paper prototype. We specifically choose representative conditions, so that users are observed with PCs and Macintosh computers, typical web browsers and both broadband and dial-up connections. Users are asked to think aloud, and we record browsing paths and also use of the search engine. We use these data to identify further problems with the design, including user navigation difficulties and frustration with any web pages that take too long to display. If the problems are still severe or numerous, we will conduct new usability testing with new users after making changes in the computer prototype.

When satisfied that we have fixed the big problems with the design — based on our usability findings — then we move on to the final production of the site.

## Build the website technically and completely

At this point we need to pay attention to numerous details for web publishing — i.e., getting final versions of graphics produced so that they look good and load quickly, creating and debugging cascading style sheets (CSS), making sure HTML or XHTML is valid, making each web page look good in terms of layout, checking the use of white space and the inclusion of graphics, and so on. We also need to test our hyperlinks to assure correct linkage. Normally we do these tasks in a web server folder that is hidden from the public web (i.e., nothing links to it) so that the public is unaware of the new site, but we can view it during bug testing.

As the production test site nears completion, we ask key School of Education stakeholders (such as our Dean, department chairs, faculty and staff) to preview it. Based on their feedback and comments we make further cosmetic changes. We do not make major substantive changes at this time, since the information architecture and overall design have already been modified based on data from usability evaluations. As mentioned above, the reviewers at this stage know that this design process is inquiry based, and that empirical data are used to make design decisions.

In fact, during the final production of the 2004 redesign, our Dean and several faculty members questioned the

redundancy of audience hyperlinks on our home page. These links occur in both the horizontal navigation bar below the page banner, and again vertically in the left-most column of hyperlinks in the page body. The Dean suggested that we should consider removing those audience links in the left-most column, since it would make the appearance of the home page “cleaner” and less “busy” (and less redundant). Our response to the Dean was based on usability results. We reported the facts: *none of our users during the usability tests actually used the links in the navigation bar across the top of the home page, and these links were seldom used on the second-level pages.* Instead, users frequently selected the audience links on the home page in the left-hand column during our usability evaluations. Moreover, during think aloud, it was very clear why. The elaborators that are immediately after each hyperlink were frequently used to decide which category to choose. These were the *empirical* findings during usability tests.



Figure 2. Testing a computer prototype.

Why, then, did we not remove the links in the navigation bar across the top of the screen? University administrators were encouraging standardization of audience links at the top of web pages as a consistent navigation device throughout IU (e.g., see <http://www.indiana.edu> and <http://www.iub.edu>) and the campus web manager, speaking on their behalf, had strongly encouraged us to remain consistent with this design goal. The compromise was to leave the links in both places on the School of Education home page, even though they are redundant. If we had not had empirical data from usability testing, the links in the left column would likely have been removed, and the usability of our home page would have suffered as a consequence.

Finally, when we are ready to go “live” and the Dean’s Office has approved the final design, we publish the website for the world to see (using our content management tools described below). If the website is completely new, then we need to add hyperlinks on other existing web pages on the School’s site and also notify other webmasters of the new site. If the site is a revision, usually little external change is required since we make every attempt to keep file names the same after a revision so that we do not break external hyperlinks to the site or parts of it. We also ask one of our university webmasters to make the university search engine index (or re-index) the new site immediately. After that, each page is indexed by the “spider” that follows millions of web links within the 750,000 web pages at our institution — which ordinarily occurs about once a month, and about every two weeks for pages that are updated frequently. Other search engine “spiders” (e.g., Googlebots) will also soon find and index our new site by following hyperlinks as they normally do.

Then we announce the new website in ways that are appropriate for the occasion. If the site is brand new, it also needs to be registered in places like Yahoo and Google.

## **Maintain the website**

Every website is like a city or town. Someone needs to manage it — to maintain it and to make needed improvements as the city grows and changes over time. This is usually where we bring in the content managers in the School of Education who will be responsible for site maintenance. Basically, we train the content managers to use our EdWeb Tools so that they can update the content as it changes in their areas of responsibility. Our philosophy is to have people closest to the content handle its management. They know when something changes and which part of the website needs to be fixed to reflect those changes. These people do not need to have special web design skills; they are normally staff members or graduate assistants in the individual units of the School of Education. They focus on the content, since we have provided them with a working information architecture derived from our needs assessment and a design and navigation system that has been evaluated with members of the target audience through usability testing. Usually someone who is experienced with e-mail and word processing can learn to use our EdWeb Tools and a WYSIWYG editor like FrontPage or Dreamweaver.

## **The overall design process is not step-by-step, but heuristic, and should be used intelligently**

The design process we use is not strictly step-by-step, but instead should be conceived of as a set of heuristics or principles to be followed. We do not always do all of the steps in Figure 1. For example, if we already have relevant data, then there is no need to conduct another needs assessment. Or if we are testing dynamic elements such as web forms and their results, we may skip the paper prototyping since these can often be tested better with a computer prototype. We normally do not skip usability evaluations however, since there is no substitute

*“We do not always follow this web design process step by step, but instead use it intelligently.”*

for good usability data to help decide what needs to be fixed in a web design. An exception to this is when we are making minor changes to a website that has previously undergone considerable usability evaluation. In that case, we just make the changes as part of regular ongoing site maintenance. However, if we are making major changes to a high-traffic navigation page (e.g., home page), then we will mock up a quick computer prototype and do a round of usability testing to make sure we have not created new problems for users.

In short, we do not always follow this web design process step-by-step, but instead use it intelligently. The basic premise is that we use inquiry as a method to make design decisions when we do not know how something will work or what needs to be there in the first place. Basing design decisions on empirical data about how well the site is working for the primary target audience helps make our website more effective than it would be if we had not done usability testing. It is more effective because big problems with the design have been identified and fixed as best we can and subsequently verified through further usability testing.

The reader may wonder: How long does this design process typically take? Unfortunately, there is no simple answer to this question, since it depends on the size and complexity of the website, the number of people who are working on the project, their skill levels and the amount of time they have available. We have found that with new and relatively small websites, such as “Enhancing Diversity” at <http://education.indiana.edu/~ediverse>, that it took a design team of 4 persons working part time about 12 weeks from needs assessment to final production. Most projects have taken between three and six months with part-time personnel. A very big and complex project, the 2004 redesign of the entire School of Education website and the Office of Teacher Education, took the authors more than a year to complete. We have never had staff who could work full time on our projects.

Finally, this design process itself is not enough to manage large numbers of web pages in an efficient manner. For cost-effective website management, we have created our own custom in-house EdWeb tools.

## **EdWeb tools for content management**

The EdWeb Page Maker tools were written by Frick and Su in PHP in 2001 and revised in 2004. PHP is a server-side scripting language that works within HTML documents to enable dynamic web pages generated on demand. Although there are several other options for this, such as ColdFusion, Java and Perl, we chose to use PHP for three reasons.

- PHP is free. It is open source and we have found examples on the web and in books readily that we can adapt into own programs (see <http://php.net>, Castagnetto, et al., 1999).
- PHP is a combination of functionality, power and robustness. It has elements of many popular high-level languages (Argerich, et al., 2002; Choi, et al. 2003). For example, programmers only need to write a few lines of PHP code to retrieve or store data in a database or a file.
- PHP and HTML are intermixable in the same file, so this facilitates development and testing of dynamic web pages. Any text editor (e.g., Pico, Homesite) or WYSIWYG tool like Dreamweaver can be used for editing.

## **How the EdWeb tools work with XML, HTML and CSS**

In the School of Education, the content of a web page is stored in XML format on a secure web server. We have approximately 6,000 XML content files which are used in combination with our custom design templates to create the 6,000 corresponding HTML files. To illustrate, excerpts are listed below from one of our XML content pages (the ‘...’ means content is omitted in examples below, to save space):

```

<?xml version="1.0" standalone="yes"?>
  <page>
  <docauth>
  Y. J. An
  </docauth>
  <doctitle>
  Welcome Visitors!
  </doctitle>
  ...
  <docbody>
  
  <ul>
  <li><b><a href="index.html">School of Education</a></b>: home
page</li>
  ...
</docbody>
</page>

```

This text is stored in an ASCII text file named `welc.xml` on our web server just as other kinds of files are (such as JPG and HTML files). XML allows us to create our own tags which are used to surround pieces of content. In the example above, the `<doctitle>...</doctitle>` pair of tags marks off the text “Welcome Visitors!” The `<docbody>...</docbody>` tags mark off the HTML content which will be inserted into the main part of the web page when published.

The EdWeb Page Maker tools combine the XML content with the appropriate design template so that a static HTML web page is published for the world to see. Here is a snippet from a *design template* for the header for our top-level pages:

```

<head>
  <meta http-equiv="Keywords" content="<?php echo ($keywd);?>, School of
Education,
  Indiana University Bloomington" />
  <meta http-equiv="Description" content="<?php echo ($desc);?>, School of
Education,
  Indiana University Bloomington" />
  <title><?php echo ($title);?>: School of Education, Indiana University at
Bloomington</title>...</head>

```

The design template is a combination of HTML and PHP instructions. We have bolded the PHP instructions in the example above. For example, the `<?php echo ($title);?>` will cause the value of the PHP variable, `$title`, to be inserted between the HTML `<title>...</title>` tags in the *design template*. Thus, the whole HTML title for a web page might be, “Welcome Visitors!: School of Education, Indiana University at Bloomington.” (Note that the HTML title does not appear on the web page itself but at the top of the browser’s window frame, and on web pages containing search results.)

Our EdWeb publishing tool, written in PHP, will read the contents of the `welc.xml` file and it will also read the contents of the design template. For example, here is a snippet of PHP code that reads the XML file:

```

chdir($subdir);
if (!$fd=fopen("$filename.xml", "r")) {
  echo ("Cannot open the file: $file!");
} else {
  $source = fread($fd, filesize("$filename.xml"));
  $auth=grab("<docauth>";</docauth>");

```

```
$title=grab("<doctype>";</doctype>");
...
$body=grab("<docbody>";</docbody>");
fclose($fd); }
```

Notice that the variable, \$title, will contain the text that is literally grabbed from between the <doctype> and </doctype> XML tags in the welc.xml file. Thus, in the *design template* above, the text, "Welcome Visitors!" will be inserted where \$title occurs in the PHP instruction: <?php echo (\$title);?>.

This is how XML content is added to the design template for the header, which is stored in a temporary PHP variable, \$shheader. In a similar manner, XML content is added to the design templates for the web page body and footer sections, and temporarily stored in PHP variables \$sbody and \$sfooter. The contents of these three variables are written to an HTML file that is stored on the web server as welc.html. The PHP code snippet that does this:

```
chdir($htmdir);
opendir($htmdir);
if (!$file=fopen("$filename.html", "w")) {
echo ("Cannot open the file");
exit;
} else {
fwrite($file, $shheader);
fwrite($file, $sbody);
fwrite($file, $sfooter);
fclose($file); }
```

The resulting HTML web page is what the world sees through their web browsers. The XML files, HTML/PHP design templates, and the EdWeb tools

are what we and our content managers use to build and maintain the website. Although we could have all of our web pages generated dynamically via PHP by reading the XML and combining with design templates whenever a page is requested, we store static HTML pages instead. We do this for efficiency—to reduce the load on our web servers, and to maximize delivery speed of web pages to users.

We use a general design template for all departments, programs and offices in the School of Education, and then we customize page headers with the unit name, sidebar navigation and footer information for each of these units. The School of Education is a complex organization with many program areas and offices. Each unit needs to have its own identity while remaining consistent with the design of the overall School of Education website (see Figure 3). The content providers in each unit edit the main content sections of their web pages. They can create a new web page or modify the content of an existing web page.

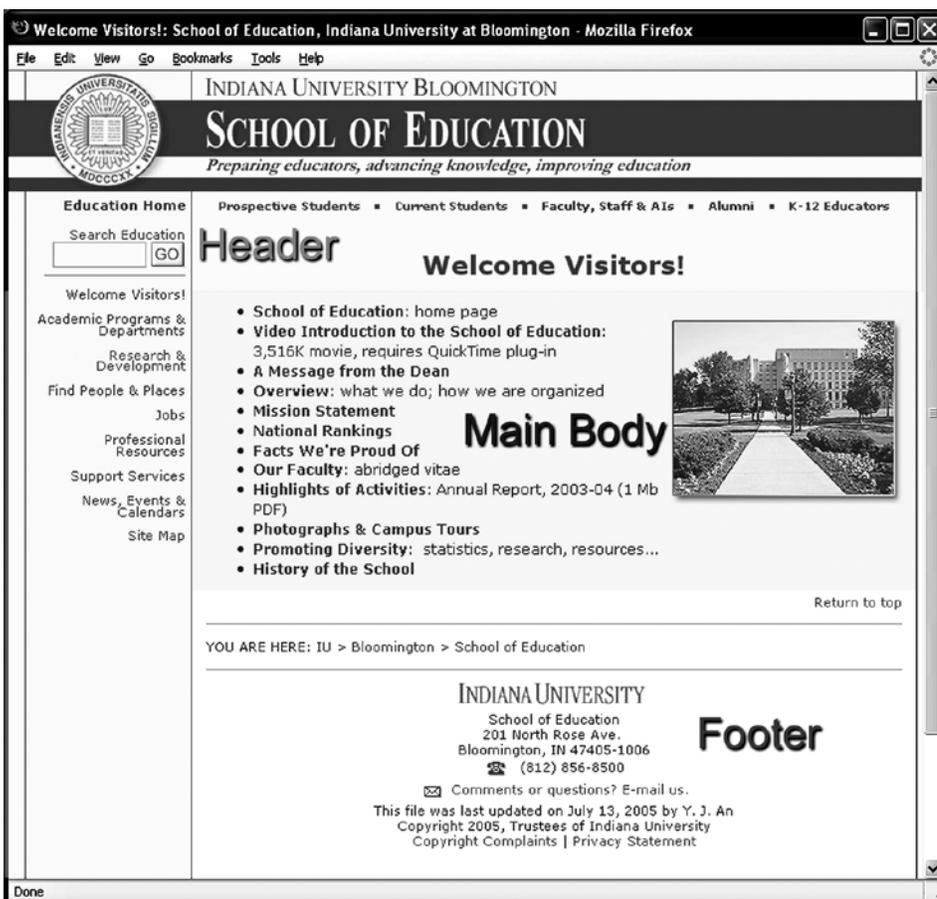


Figure 3. Three main sections of a web page.

Whenever there is a change in a particular template, sidebar or footer, the web director at the School level will make the change and use the EdWeb PageMaker tool to re-publish all the web pages with the new appearance in a given sub-site. Once a template is changed, republishing takes only a few seconds on the live website.

Aside from efficiency and controlling the consistency of appearance of our web pages, another very important reason for using design templates is to help ensure that our web pages follow web standards and will be rendered consistently by all major web browsers. We have found the work of Jeffrey Zeldman (2003), *Designing with Web Standards*, and Eric Meyer (2003) to be very helpful for technical strategies in creating our design templates. The technical details regarding XHTML and CSS are beyond the scope of this article. However, these strategies are extremely important for the long-term maintenance of our website and for reducing costs and time for making changes, and for minimizing problems that particular web browsers may encounter when rendering any of our web pages.

In short, we made every effort with our 2004 design to be compliant with XHTML and CSS standards. XHTML is the standard for eXtended HyperText Markup Language, and CSS is for Cascading Style Sheets. While this required about two months of development, testing and debugging initially, that effort has subsequently improved efficiency whenever we want to modify the design. For example, if we want to change the appearance of our hyperlinks (e.g., color, font, style), we just modify the CSS file. If we want to change the font size of all normal paragraphs, we just modify the CSS file. If we were to move the sidebar from the left side to the right side, all we would need to do is modify our CSS file. Once we save the modified CSS, then the entire website instantly takes on this new look. We do *not* have to republish any web pages, or change any content. We just change the style sheet. Interested readers can view our CSS file at: <http://www.indiana.edu/~educ/css/soe.css>. To see the XHTML for any of our 6,000 web pages, e.g., <http://education.indiana.edu/welc.html>, just go to that web page in your browser, and then use your browser's "view source" on the menu bar. Note especially that the DIV identifiers correspond to those in the CSS file, and control the appearance of blocks associated with the banner, sidebar, main body and footer divisions.

It is also important to note that we can create different styles for different kinds of use of our website. For example, we have a style defined for *printing* our web pages that helps save paper by hiding the unnecessary web navigation (e.g., sidebar, top navbar), increasing margins, reducing font size, and so on. We have a style defined to accommodate older web browsers such as Netscape 4.x, which pre-date the subsequent formation of XHTML and CSS standards, to keep them from crashing or garbling how our pages appear. We have embedded style elements to improve accessibility of our web pages by those with visual impairments who use screen readers. Eventually, we may create styles for hand-held devices such as PDAs.

Finally, we have located all of the graphics for our templates in a central location on our web server, so that all of our design templates (and hence all 6,000 web pages) point to the same web addresses for those images. This greatly simplifies maintenance. If we want to change one of these images, we simply replace the file in one location. For example, if we wanted to change the appearance of our banner, we would need to replace the graphics files associated with the banner in just one location, and if necessary change the CSS specifications for dimensions or the precise coordinates where the banner is rendered on a web page. We do not need to republish any web pages. The whole site is changed immediately.

The other very important reason for keeping the graphics centrally located and consistently referred to by all our web design templates is to minimize the time users will wait for web pages to be downloaded and rendered. Once a user has viewed one of our web pages, his or her browser will have cached all of the main graphics. Thus, when this user views another page, the browser does not need to download those graphics again from our web server. This helps make our

*“Once we save the modified CSS, then the entire website instantly takes on this new look. We do not have to republish any web pages, or change any content. We just change the style sheet.”*

web pages load very quickly, typically no more than a few seconds on a dial-up connection. Thus, users seldom need to wait after clicking on hyperlinks when they navigate our site; this helps to reduce frustration.

## How School of Education staff use the EdWeb tools

The EdWeb Page Maker tool is easy to use. The process of how this tool works is described on the web at: <http://education.indiana.edu/guide/guide.html>. Only trained content managers are given access to the EdWeb Page Maker.

Note that for security reasons, we have smudged the specific URL information in the screen displays illustrated at the URL above. There are no public hyperlinks to our EdWeb PageMaker tools. Our content managers make bookmarks in their web browsers to their EdWeb starting pages, and then they must supply the proper credentials to gain access. An .htaccess file limits which users can gain access to folders that contain our EdWeb tools. This method is common to Apache and IIS web servers. Only users who pass the authentication required by the .htaccess file can run programs and access files located in this folder. At Indiana University, each user has his or her own username and password. We just put the username in our .htaccess file or .htgroup file. See: [http://webmaster.iu.edu/security\\_info/index.shtml](http://webmaster.iu.edu/security_info/index.shtml). Content managers only need to know their personal passwords to gain access. If we did not take these precautions, then potentially anyone in the world could change our web pages if they knew where the EdWeb tools were located and the specific URL for that unit's starting page. As it is, even if they did know the URL, they still could not get in without proper credentials. This security system has worked well for us for many years now.

These EdWeb tools have three advantages. First, they allow us to control centrally the overall appearance of the web pages so that all the web pages within the School of Education have a fairly consistent look.

Second, the EdWeb tools allow us to decentralize the content management and move this function to appropriate individuals in their respective programs, departments and offices. We conduct an initial, one-time training session of 1-2 hours with each content manager. Each unit typically has one or two persons to take care of its website, which we consider to be a sub-site of the overall School of Education website. Since the website management in the local unit level normally does not require more than one or two hours per week, the local web content manager usually has many other duties, (e.g., as a secretary or graduate assistant). However, there are a few content managers who are responsible for a number of websites within large departments.

Third, the EdWeb tools require content managers to enter page titles, main headings, keywords and content descriptions in addition to the web page content. Our tools further append additional text to page titles and meta-tags for keywords and

content descriptions automatically. The appended text includes the specific unit name (office or department), followed by "School of Education, Indiana University Bloomington." This information helps search engines to index our web pages accurately, since words that occur in titles, main headings, keyword lists and content descriptions are given more weight in the indexing process that search engine spiders use (the spiders have no semantic means of determining what words are most important, so the web page, in effect, tells them!). In turn, this makes it easier for users to find the web pages they are looking for by using a search engine, since it can work better with this additional information. And finally, the titles of links in search results are very descriptive, since the titles of those links include unit and institution names. For example, a link to one of our web pages on a Google search results page does not say just "Master's Degree Program," but instead it says "Master's Degree Program, Instructional Systems Technology, School of Education, Indiana University Bloomington."

In addition to tools for our content managers, we have developed similar PHP-based tools in the School of Education for faculty and staff to update their own profiles, for outsiders to send e-mail to our faculty and staff without revealing faculty and staff e-mail addresses on our website (to minimize potential harvesting and the resulting flood of spam messages), and for anyone to submit job announcements to various departments and program areas to post on their websites.

## General strategy for School of Education web development and management

In summary, our general strategy of maintaining over 6,000 web pages is to:

- Use an inquiry-based approach to design — user needs assessment, rapid prototyping and usability testing
- Keep content in XML format, separate from its appearance on the web
- Have web designers at the school level create HTML/PHP templates and EdWeb Tools in PHP, and have content managers focus on content
- Let those closest to the content maintain and update it

The web director is a faculty member in the School of Education who is appointed for 12 months, half-time for web duties and half-time for normal faculty activities that include teaching, research and service. He normally teaches graduate courses in web design and computer-mediated learning, so there is a good alignment between his areas of expertise and web demands. He is a highly experienced computer programmer and has considerable technical expertise in how client computers, servers and the internet work together. He also does research related to web design and usability evaluation. Finally, he has often directed teams of students in his classes to work on design projects that have contributed to the School of Education website during the past decade.

The two web assistants are doctoral students in Instructional Systems Technology who have complementary

skills. One is a computer programmer with a computer science undergraduate degree. The other is more skilled at visual design, photography and image editing. They typically work about 15-20 hours each per week and assist with general website maintenance of top-level pages, refine and expand our EdWeb tools, and help our content managers with technical problems as needed. At the time of this writing, the School of Education is planning to hire a full-time, highly qualified webmaster responsible for duties of the two graduate assistants and much of the day-to-day work of the web director. The current web director plans to remain in an advisory capacity, while resuming full-time teaching and research activities.

In 2004, 28 content managers handled website maintenance for their units as part of their duties. Web content managers typically spend *no more than an hour or two a week* on web matters. Most of the time they are working in maintenance mode, keeping existing content accurate and up-to-date and adding new content as needed. The content managers are directly supervised by the department chair, faculty member or office manager to whom they normally report for their other duties. Content managers typically require about two hours of training on use of our EdWeb tools. The web director and assistants help the content managers with technical problems and do individualized follow-up training as needed.

Everyone is highly motivated to keep their web content up-to-date and accurate, since they know that large numbers of people are visiting our website. They, or anyone else, can see statistics on our web traffic at: <http://education.indiana.edu/webstats.html>. If the web director spots a problem with a particular web page, he knows whom to contact, since the page author's name and the date the page was last updated are displayed in the page footer.

This cost-effective system of web management has been in place for the past six years and has worked well for us. We have been able to build a large, successful website very efficiently by use of a content management system that is facilitated by our EdWeb tools and by the inquiry-based web design process we have created and refined.

## Acknowledgement

Professor Elizabeth Boling has made major contributions to the research on the overall web design process as illustrated in Figure 1. She also contributed the artwork for Figures 1 and 2. She is a faculty member and currently chairperson of the Department of Instructional Systems Technology, School of Education, Indiana University Bloomington.

*Theodore Frick is an associate professor and web director for the School of Education, Indiana University Bloomington. He teaches courses in computer-mediated learning, advanced instructional development and production,*

*and research methods in instructional technology. His research interests include web design praxiology and educational systems theory (SimEd). For further information or to contact, see <http://education.indiana.edu/~frick>.*

*Bude Su is a Ph.D. candidate in instructional systems technology and works as assistant of the web director at School of Education in Indiana University Bloomington. Previously she was the national director of International Education and Resource Network in China (<http://www.learn.org>). Her major research interests include online teaching and learning, technology integration in education, organizational behavior and knowledge management.*

*Yun-Jo An is a Ph.D. candidate in instructional systems technology and works as an graduate assistant for the School of Education web director at Indiana University Bloomington. Her major research interests include online learning, collaborative problem-based learning (PBL, and organizational learning.*

## References

- Argerich, L., Choi, W., Coggeshall, J., Egervari, K., Geisler, M., Greant, Z., et al. (2002). *Professional PHP 4*. Birmingham, England: Wrox.
- Boling, E., & Frick, T. W. (2002). *Practical web development: A systematic process*. Unpublished manuscript. Bloomington, IN: Instructional Systems Technology.
- Choi, W., Kent, A., Lea, C., Prasad, G., & Ullman, C. (2003). *Beginning PHP 4*. Birmingham, England: Wrox.
- Dumas, J., & Redish, J. (1999). *A practical guide to usability testing* (revised ed.) Exter, England: Intellect.
- Krug, S. (2000). *Don't make me think! A common sense approach to web usability*. Indianapolis, IN: Que.
- Nielsen, J. & Landauer, T. (1993). A mathematical model of the finding of usability problems. *Proceedings of the INTERCHI'93* (pp. 206-213). Amsterdam, the Netherlands: ACM Press.
- Nielsen, J. (2000). Why you only need to test with 5 users. Retrieved May 21, 2005, from <http://www.useit.com/alertbox/20000319.html>
- Meyer, E. (2003). *Eric Meyer on CSS: Mastering the language of web design*. Indianapolis, IN: New Riders.
- Reigeluth, C. M., & Frick, T. W. (1999). Formative research: A methodology for improving design theories. In C. M. Reigeluth (Ed.), *Instructional-design theories and models: A new paradigm of instructional theories*, Vol. II (633-651). Mahwah, NJ: Lawrence Erlbaum.
- Shneiderman, B., & Plaisant (2004). *Designing the user interface* (4<sup>th</sup> ed.). Boston, MA: Addison-Wesley.
- Snyder, C. (2003). *Paper prototyping: The fast and easy way to design and refine user interfaces*. San Francisco, CA: Morgan Kaufmann.
- Thiagarajan, S., Semmel, D., & Semmel, M. (1974). *Instructional development for training teachers of exceptional children: A sourcebook*. Minneapolis, MN: Leadership Training Institute/Special Education.
- Zeldman, J. (2003). *Designing with web standards*. Indianapolis, IN: New Riders