Tackling the Problem of Repurposing Web Content

An XML-based content management system helps Utah Valley State College manage the growing challenge of website maintenance

By **Nathan Gerber** and **Lance Merker**

Like many institutions across the country, Utah Valley State College (UVSC) found itself struggling to keep its website current. The IT department, which was responsible for keeping site content up-to-date, faced a growing burden in maintaining and updating the site. It was a familiar story—never enough time, money, or resources to do things right, and the process increasingly bogged down as the workload grew.

In the spring of 2003 we began looking at alternatives that would simplify and streamline the process of updating the university's web pages. Specifically, we wanted a straightforward way to repurpose content across pages. At the same time, we needed a solution that would allow stakeholders outside the IT department to contribute and maintain website content without having to become technical experts. Because the IT department would support the technology chosen, it was decided that IT would make the decisions. Several key staff from marketing, which is responsible for site design and usability, were also involved in the process.

Separating Content from Design

From the outset, we knew that the key to effective repurposing of content is separating content from design. Historically, web pages included content and design in static files, requiring considerable manual effort to reuse common content in different places or to make changes to that content or its presentation.

We looked at many different content management systems (CMSs) from a broad range of vendors, quickly narrowing the options to a short list. We implemented test instances for a few of these systems, some of which were very static in nature and some of which were very dynamic and robust. Most of the systems dealt with a proprietary technology or implementation, which caused some concern on our part. Then we hit on XML (Extensible Markup Language) technology and began looking at the systems that use it.

XML offers a way to create content components or "chunks" of data that can be easily reused and reformatted for different purposes. With XML, content is separated from structure, look and feel, and design, providing a universal format for organizing, structuring, and storing site content. XSL (Extensible Stylesheet Language) controls the selection and formatting of XML data, using a language called XSLT (XSL Transformation). XSL provides simple tools to control the design, format, and even file output type of selected content. Using XSLT within XSL files permits filtering, sorting, and processing of content and data before generating and outputting a file. Basically, it allows your XML data to be formatted the way you want it for publishing on a web page. The most important characteristic of this technology is that it allows us to separate content from design and structure using standard technology that is widely recognized in the web industry.

We knew that an XML/XSL approach would provide the freedom we needed to separate content from design. This would allow a design team to create a web layout, design, and information architecture to fit today's needs, and then later change it by simply modifying the XSL style sheet. We could, in fact, have several style sheets working simultaneously, with one to control the look and feel of top-level pages, another to control second-level pages, and perhaps another for faculty pages.

XML data can originate from files stored anywhere on the web. For example, an XML file containing course or faculty information could be output by a database at the institutional level, allowing subsets to be displayed on web pages at each individual college or department. This power to quickly and seamlessly manipulate data into different file formats

(web page, Excel spreadsheet, raw text, and so forth) in a cost-effective manner attracted us to using XML as the basis for handling campus web content.

A Tool Anyone Can Use

The content contributors (departmental staff or faculty assigned to maintain a department's web presence) to UVSC's website were too busy with their primary job responsibilities to adopt a content solution with a long or steep learning curve. For this reason, we knew we needed a CMS to oversee development and maintenance of the website. More than just a tool for editing web pages, a CMS simplifies the entire site-management process. A CMS makes it possible to keep up with the day-to-day demands of maintaining hundreds of web pages, and it helps manage complex content architectures and multimedia mixes. A CMS pulls website tasks into one process, in a controlled environment, where page creation, updating, and publishing can be tightly managed. If implemented correctly, a CMS can do this cost-effectively while leveraging existing staff.

Many of today's CMS vendors have their roots in the Internet business boom, which was driven by e-commerce and publishing companies that needed systems to support transaction-based online businesses. Highly customized multimillion dollar CMS solutions were all the rage in the late 1990s, but none of them fit the specifications or the budgets of higher education.

For higher education, a CMS should provide simple tools that make editing a web page as easy as editing a document. This simplicity is important to staff members contributing content to the website, because their time is better spent addressing the needs of students than mastering technology. An appropriate CMS allows subject-matter experts to become fully engaged in the process of keeping the website fresh with timely and responsive content, without burdening them with technical details and a steep learning curve.

Narrowing the List

We concluded that using XML/XSL within a CMS would be the best solution for UVSC because it would reduce our workload without imposing a significant training requirement for users. The creators of transaction-oriented websites have long understood the benefits of using XML in a CMS; thus, many systems catering to these markets were designed around the XML/XSL model. These systems often require the use of XML across an entire website, however, necessitating a site-wide redesign and implementation effort.

For UVSC, this type of expense and resource drain seemed unnecessary and counterproductive. We wanted to keep the UVSC solution within budget and simple for users. As a result, we sought an on-demand (or software-as-a-service) CMS, which would allow us to manage, within a single environment, multiple content types including structured XML and the many thousands of pages of legacy HTML content that UVSC must maintain.

We wanted a system that had proven itself in higher education, so steered away from products developed for e-commerce, news, media, or similar industries. We spent a fair amount of time researching the issues with our peers on other campuses, keeping in mind the differing CMS needs of different institutions. We looked specifically at what worked for institutions that had already tackled the web content problems UVSC faced. We communicated with several schools having similar issues to solve with their CMS implementations, comparing their needs with ours through e-mail, in-person meetings, and telephone conversations.

Although we could have selected a complex system and hired programmers to customize it for our needs, as some of the institutions we consulted had done, we wanted a modular and scalable CMS that allowed plug-and-play options whenever possible. For example, just because a CMS provides an API for building a blogging component doesn't mean that building your own blogging application is an appropriate use of staff or budgetary resources.

The "must have" requirements boiled down to an on-demand (therefore cost-effective) standards-based solution simple enough for any staff, faculty, or student to use and to leverage other existing applications that most content contributors were already familiar with, such as Microsoft Word, Adobe Photoshop, and PDF creators like Adobe Acrobat. Because these applications were already familiar, there was no learning curve for most content contributors, which saved time and resources. We also wanted our new system to allow our programmers to continue using their toolset (at the time consisting of Dreamweaver, Fireworks, and PageMaker) to develop any needed back-end templates or structures for the new system and to leverage data in existing systems. In the end, we selected OU Campus, from OmniUpdate, Inc.

Getting Started

We started our work in the new CMS by using Adobe's Dreamweaver web design tool for coding and creating templates in XHTML and setting up the XML structures. Contributors could then use these templates and structures to create their web pages of content. Because the application was already familiar to us, there was no learning curve, which saved time and resources. We also use Adobe's Fireworks for graphic manipulation and creation. Fireworks lets us create vector and bitmap images that we later optimize for website use by saving them as other formats, such as JPEG. Our development team also used Acrobat for converting documents to PDF for web posting and PageMaker (now InDesign) for exporting existing document data in XML format. The Dreamweaver, Fireworks, Acrobat, and InDesign applications are now Adobe products, which was not the case when we started. These applications could be swapped out with other tools that accomplish the same functions; we chose them because they were standard or use standardized technologies and saved us time and resources.

With our templates and structures created, it was time to start tackling projects. In cooperation with the UVSC user community, we identified a few projects needing immediate attention that, if implemented correctly, could be repurposed with great value across the campus for a variety of groups. These projects included the course catalog, tuition tables, and campus policies.

Repurposing Course Catalog Content

In the past, the catalog section of our Curriculum Office would build the course catalog and prepare it for print. IT would receive an electronic copy and spend days converting it into a web-ready format. With our new system, we could quickly generate an XML document from the electronic file using PageMaker, since this is where the catalog data were available. This document fell right into the site's course catalog template. The template had been designed by marketing staff, but in order to repurpose the content, we created a new version of the template in Dreamweaver. Campus departments could grab their specific content from that XML document and drop it into their own site templates using the CMS and their login information. The result was a great step forward for UVSC in both content management and design, and a huge time-saver for everyone involved.

Updating Tuition Tables

Keeping tuition tables current has always posed a problem. We now have an XML document that is formatted and published using an XSL style sheet. This allows the Tuition Department to maintain the actual table data, changing them when necessary in only one place. Those changes automatically update the information in all other content locations, providing a level of consistency that is vital for various audiences, eliminating a burden on the users, and nipping a potential problem in the bud.

Converting Campus Policies

Unlike some institutions, UVSC's campus policies reside in The Online Policy System (TOPS), which manages the process of routing and approving policies. Once approved, a policy is posted to our live web server for public reference and in XML format for entities across the campus to extract for their own use. Many departmental websites refer to policies but, for a variety of reasons, do not want end users (students, parents, or alumni) to be redirected away from the departmental site. The CMS structure allows each department to use the policies within its location, keeping end users on that particular site. Again, distributed content is automatically updated, minimizing potential problems and duplicative work.

Moving Forward

With these projects successfully completed, we have moved on to UVSC's home pages. We are working our way through these pages by audience type (prospective students, visitors, students, faculty and staff, alumni, and so forth) and will hand them to marketing to maintain through the CMS. The first area we plan to migrate serves prospects and visitors because we expect to benefit the most from a more uniform look and feel and consistent information in this area. We expect to change our design template for these external audience home pages every 18 months in an effort to keep the pages interesting and current.

As we transition pages and sites into the new system, we train the associated users to maintain their content, which can take as little as 30 minutes. Eventually, we expect to have IT staff out of the business of content maintenance and focused entirely on supporting the systems that run the site. We hope to have completed the transition of the UVSC site by July 2008.

Final costs of implementation are not yet known, but as of this writing we have stayed within the estimated annual expenditure in our budget. On average, we have eight people at a time working on the implementation, depending on the specific tasks as we work our way through the process. Although we have moved past the planning and research stages and on to implementation, we have only scratched the surface of implementing XML on the UVSC site. Still, we have already achieved noteworthy success in our early projects.

Recommendations

The best advice we can offer is to plan well. This includes planning for the technologies to be used and the people who will be involved. Get the key players on your team, and target a small number of projects that will demonstrate the "wow" factor to everyone before you undertake wide-scale implementation. This strategy will help win over the naysayers.

For any project of this kind and scope, usability is the key—or the killer. Acceptance and use are driven primarily by ease of use and comfort with the tools. In our situation, given how complex XML can be, using a CMS that effectively masks this complexity is an important part of ensuring simplicity and encouraging use. Remember that the return on investment of new technology is zero if no one uses it.

Another important consideration for us was to choose a framework that did not force us to migrate legacy content into a proprietary database schema. Forcing such a migration was unnecessary and might have placed such severe time and technical burdens on our staff that the project would have stalled. It was more important that our framework be straightforward and allow content reuse and the separation of page design so that staff could easily and quickly start repurposing content.

Selection of appropriate technologies and tools is vital to a successful content management project. For UVSC, the choice of an XML-based CMS was key to improving the flexibility of our website, simplifying the maintenance of content, and managing the workload for IT staff. We cannot emphasize enough the value of using standard technologies to support content repurposing as well as continuous development and growth. As we've often heard, nothing stays the same, and this is certainly true when it comes to website content.

In our case, the participation of marketing has been essential. The marketing team is charged with website research and design. They gather usability and audience-experience data through focus groups, which aids all of us in making good choices concerning the website. Such practices might prove invaluable in your migration efforts as well.

It's easy to get caught up in the implementation of a new technology and lose sight of its intent. Remember that users are the reason for tackling such a project. Content management is a framework that includes people as much as it provides technological solutions. It is important to employ strategies that are simple for end users and administrators to adopt. If your people cannot, or will not, participate, then the framework is worthless. Address the diverse needs of your most valuable resource—the people who will use the system. *e*

Nathan Gerber (gerberna@uvsc.edu) is Director of Web Development Services at Utah Valley State College. Lance Merker is President and CEO of OmniUpdate, Inc.