

Light at the End of the Tunnel: Proven ROI by Using a DITA-Based Content Management System



Keith Schengili-Roberts, AMD

Photo not available at time of print

Keith Schengili-Roberts
AMD
keith.schengili-roberts@amd.com

Keith Schengili-Roberts is the Manager of Documentation and Localization at AMD for the Graphics Products Group. Prior to gaining this role, he worked as the Information Architect, redesigning the documentation delivery and the processes behind them. Keith is an award-winning lecturer on Information Architecture and Information Management at the University of Toronto's Faculty of Information Studies Professional Learning Centre. He is also the author of four professional technical titles, the most recent being *Core CSS, 2nd Edition*.

INTRODUCTION

Several years ago, the AMD Graphics Products Group (then ATI Technologies Inc.) decided to move toward using the XML-based Darwin Information Typing Architecture (DITA) for our documentation and localization needs. The goal was to reduce costs through content reuse and the creation of a more efficient localization process.

We were no strangers to the concept of reuse: our Information Architect had already successfully implemented a single-sourcing toolchain. However, we reasoned that developing and maintaining our content in XML—and managing it within the framework of a content management system (CMS)—would produce further cost savings and a significant long-term return on investment.

As it turned out, we underestimated both the cost savings and the benefits. Not only have localization costs been dramatically reduced after just over a year of full implementation, but we've also logged sizable productivity gains. We're now delivering more documents in less time and at a lower unit cost, giving us more time to focus on improved quality.

BACKGROUND AND BEGINNINGS

Our Documentation and Localization team is made up of nine people: four senior technical writers, two localization project coordinators (who also provide QA services), two process engineers, and a manager (who often doubles as a writer, depending on workload). The team works closely with a number of other AMD departments, including a group that is responsible for delivering technical training

and engineering information to AMD partners around the world. We produce a wide range of documentation, including partner training materials (40 percent), partner engineering materials (40 percent), and end-user documentation (20 percent). Our materials are delivered mainly in PDF format (much of it also printed), with most of the remainder provided as XHTML-based online Help.

Our localization requirements have always been imposing. We deliver documentation in as many as 26 languages, most of which ship at the same time. We localize the online Help and GUI for our ATI Catalyst Control Center software, which is provided in 22 languages and updated monthly. End-user manuals for our OEMs are typically shipped in nine to 18 languages per release, often with a dozen major product releases and variants over the course of a year. Our engineering and training materials are typically translated only into Traditional or Simplified Chinese but are

often lengthy and present unique localization challenges for our vendors and localization coordinators.

To cope with these and other documentation challenges, we took steps in 2004 to investigate ways in which we could streamline our processes and, while we were at it, improve documentation quality and consistency. At that point, we were using a FrameMaker-based toolchain, with Perforce as our version-control system. The first step was to perform a comprehensive information architecture-based audit, with the aim of implementing more effective single-sourcing methods.

“Not only have localization costs been dramatically reduced after just over a year of full implementation, but we've also logged sizable productivity gains.”

The solutions we crafted helped us produce higher quality, more consistent documentation, and served to reduce overall localization costs by year's end. But it was also clear that there were limits to this approach, and we would soon need something more flexible and extensible.

That same year, the OASIS international standards consortium was bringing together DITA users and tool vendors in an effort to develop a common specification.

Among the vendors were companies developing CMS solutions, and we quickly realized that the DITA/CMS combination was exactly what we were looking for.

The first hurdle was to choose a CMS. Our initial survey of the market revealed over 70 potential vendors with a bewildering array of features and functions. To whittle this list down to something more manageable, we decided on three primary requirements:

DITA support, Unicode enforcement, and the ability to generate printable documents.

The Unicode requirement was particularly important to our goal of improving the existing localization process. FrameMaker output often required the use of multiple code pages, which made it difficult to produce consistently error-free localized documentation. For instance, Japanese documentation might require the use of multiple code pages within a single page, making conversion from one format to another a chore. Unicode would not only address the code page problem but would allow us to easily accommodate right-to-left (RTL) languages such as Arabic and Hebrew, something our old toolchain simply couldn't handle.

A Unicode-enforcing environment also benefits writers since characters remain constant no matter what font is used for output. Writers no longer need to worry about font-dependent glyphs, where, for example, "use 15 W resistor" (correct) becomes "use 15 W resistor" (incorrect).

We discovered that only a few CMS vendors fulfilled our primary requirements.

In the end, we chose IXIASOFT, largely on the strength of its TEXTML Server document repository. Working under contract to AMD, IXIASOFT created a custom DITA-based CMS client that uses TEXTML Server as its back end. This project grew into the DITA CMS Framework which is now offered by IXIASOFT to customers worldwide.

The design work, vendor research, and formal RFQ phases were complete in the third quarter of 2005, and the final technical evaluation and vendor selection occurred in the following quarter. IXIASOFT completed a preliminary version of the CMS in December 2006. Though this "alpha" phase of the project allowed our writers to gain experience with the tools and learn DITA basics, all had been well versed in modular, topic-based approaches to content

development long before the final toolchain selections were made. Thus, within weeks of the Phase 2 delivery in January 2007, we were using what was to have been a beta version of the product to generate and localize complete deliverables.

"...writers were spending only 50 percent of their time writing and the other half of their time formatting content."

REAPING THE BENEFITS

In late 2006, we found that during a month-long test of our old tools, writers were spending only 50 percent of their time writing and the other half of their time formatting content. The average applied to writers of varying experience and material of various levels of complexity. That test convinced us that WYSIWYG XML editors would be equally distracting and cause writers to lose sight of the goal of focusing on content rather than presentation. Thus, we opted to use oXygen, a non-style-based XML editor from SyncRO Soft Ltd.

The shifting of our entire current documentation base to the DITA CMS has resulted in a number of other major advantages. The system's efficient search engine has led to higher topic reuse rates, and the CMS has

become a central repository from which the writers can mine information any time. In addition, image files are handled within our CMS using a specialized topic type, which means, in effect, that we simply reference another topic when specifying an image. When published, the “image topic” is replaced by the referenced image. This approach is particularly useful when it comes to localization, since we no longer need to reference language-specific directories. We’ve also learned to avoid using English text in images as much as possible, reaping further localization cost benefits.

The CMS allows us to produce localization kits using XLIFF (XML Localization Interchange File Format) against the source DITA XML. This capability allows us to send to localization vendors only those topics that contain content changes over what’s held in Translation Memory and eliminate the “100% match” charges we used to incur. In addition, since formatting is handled by a separate process, we need only localize the raw XML for any target language. This process eliminates the often high DTP costs we regularly incurred using the old toolchain.

The localization cost reductions have accounted for much of our planned ROI. The conservative estimate is a 10 to 13 percent reduction in annual localization costs. The new system has also dramatically shortened our

localization cycle from weeks to days as less material requires processing.

An unexpected benefit of employing the new open source standards is that it helps us quickly and easily identify weaknesses in the material returned from localization vendors, including non-canonical Unicode and malformed XLIFF. The filters we’ve put into place ensure that none of this material makes it back into the CMS. The tests have prompted us to review our localization vendor arrangements, and, where necessary, switch to more technically savvy vendors and boost the quality and consistency of our localization output.

We’ve also discovered that we can use the metadata in CMS files to measure individual and group productivity and status, and, in conjunction with the powerful built-in search capabilities, provide management and contributors with an array of invaluable metrics.

Figure 1 is an example of how we use these metrics. In addition to a precise monthly topic count, we’re able to track the total cumulative number of topics contained within the CMS. When the former values are plotted on a logarithmic scale in Figure 2, we find that the writers typically produce an average of more than 1,000 new or modified topics a month (a value which has been steadily increasing over the past quarter).

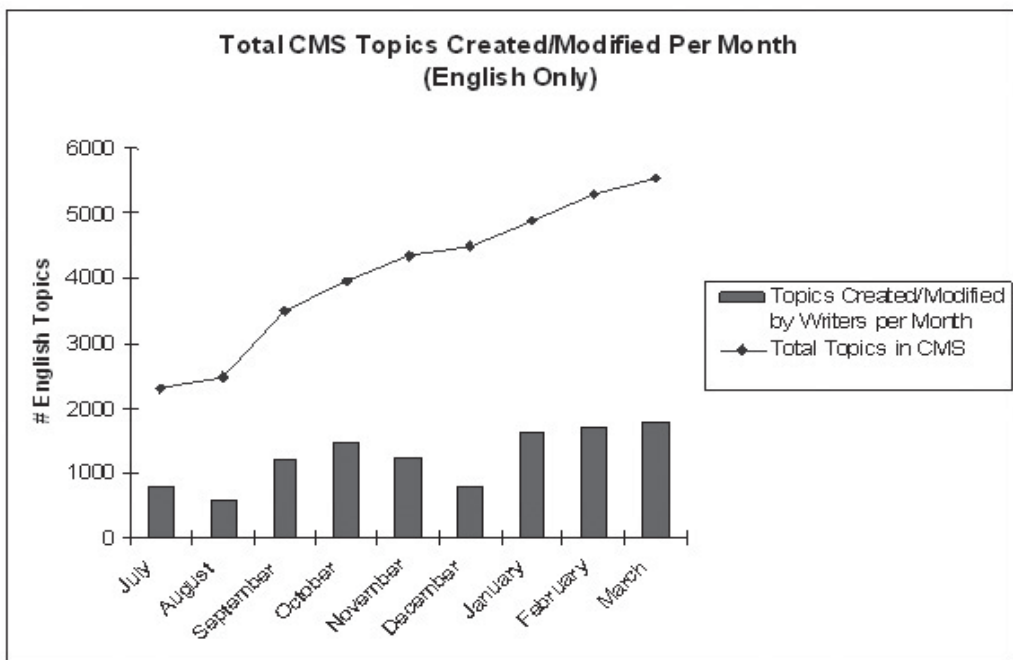


FIGURE 1: TOPICS CREATED OR MODIFIED BY WRITERS PER MONTH

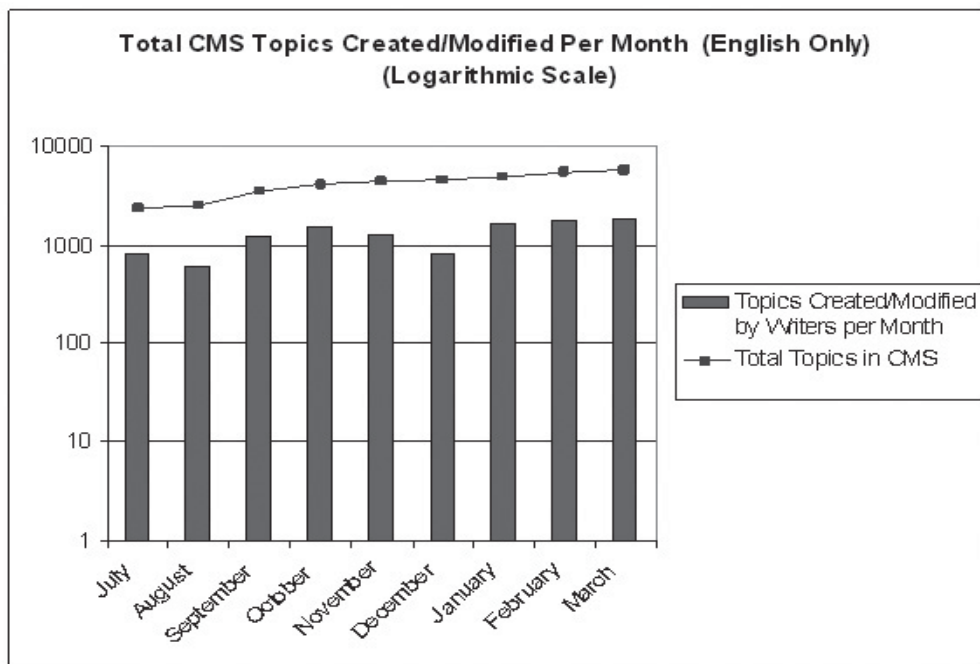


FIGURE 2: TOPICS CREATED OR MODIFIED PER MONTH (LOGARITHMIC SCALE)

As shown in Figure 3 (on page 66), we have, as of April 2008, produced just under 450 documents using IXIASOFT’s DITA CMS Framework, a figure which includes localized versions. What was interesting was not just the steep climb in the number of total publications, but the fact that the numbers of documents published in English were growing at a higher rate than expected. While one would expect that the number of localized documents ought to be higher than the number produced in English (given that each English document produces multiple localized versions of itself), the number of documents produced in English was roughly half of the total.

It’s now long been apparent that the new system has allowed us to meet an increasing documentation load with no need for additional headcount. Wanting to know whether we could quantify this increase in productivity prompted a review of our data of work completed using the old toolchain versus work completed using the CMS. We had detailed information on the total number of published documents from a job tracking mechanism that had been in place from the old toolchain, and we searched through current data from the CMS to compare documents of a similar type. Reducing the amount of time spent on formatting should

translate into an increase in content production, of course, but the 200 percent increase in output surpassed all of our expectations.

Table 1 (on page 66) indicates just how successful the new implementation has been compared to the old toolchain.

The data is drawn only from the three quarters in which we have comparable data, and it includes results from similar topic sets that were largely immune to the variance in product schedules. The table clearly shows that we’ve produced significantly more documents of the same type and size using the DITA CMS Framework. We’ve also reduced the median time it takes for document creation to publication by more than half. In fact, we reduced the mean time by almost two-thirds. Not only have we more than doubled output based with the same headcount, we’ve become faster and more consistent in delivering that output.

The factors that I believe have led to this increase in productivity relate in part to the unique nature of using DITA within a CMS. We get significantly better reuse of individual topics than with the old toolchain thanks to the search capabilities in the CMS. It is feasible to construct new documents solely by recycling existing topics from several maps into a new map. The time not spent working on formatting

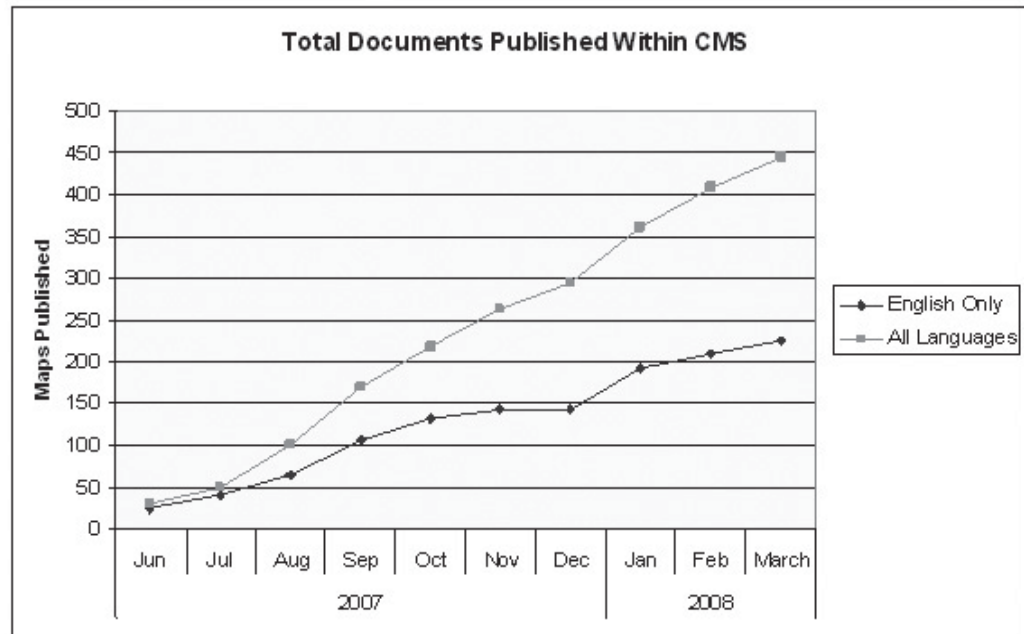


FIGURE 3: NUMBER OF DOCUMENTS PUBLISHED

undoubtedly plays a role as well, and I suspect that DITA's focus on topics as stand-alone entities has forced the writers to produce more internally consistent information than before, which not only improves the speed of delivery but the quality of the information provided. Given the fact that we have been authoring in a topic-based manner prior to implementing a CMS, it is likely that another organization working in this manner would see greater productivity increases that we encountered. If people are looking for ROI measures prior to implementing their own DITA-based CMS, it is hard to beat measurably greater productivity and faster document throughput on top of lower localization costs.

Of course, establishing a CMS to manage DITA content may not be required in all circumstances. If, for example, documents aren't localized into more than a couple of languages, if content reuse is not a significant concern, or if project timelines are lengthier, ROI may be less significant. For our organization, however, the cost savings and other benefits have been substantial, and we fully expect those benefits to continue and to expand.

The CMS system combines the efficiencies inherent in topic-based authoring with industrial-strength processing tools. I can't imagine any other way of using DITA more effectively. □

	Old Toolchain			DITA XML CMS			Ratios
	Q3 2006	Q4 2006	Q1 2007	Q3 2007	Q4 2007	Q1 2008	
Document Count	8	5	8	16	14	18	2.28
Median time (days)	52.5	40	32.38	19	14.5	25.88	0.48
Mean Time (days)	79.5	70.8	35.5	23.87	37.86	8	0.38

TABLE 1: PRODUCTIVITY INCREASES BETWEEN THE OLD AND THE NEW TOOLCHAINS