Ten Problems with Single Sourcing

Dave Yeats and Heather Hull

Though there have been numerous conference papers, articles, and books devoted to the topic of single sourcing, there have been fewer works about potential problems that should be identified before adopting a single-source documentation strategy. This study looks at ten specific problems (including issues of training, productivity, and morale) that can arise during the implementation of a mature single-sourcing model of documentation management. This list of problems, while not comprehensive, does provide some points of reference and a framework within which technical communicators can consider the implications of adopting a single-sourcing documentation model.

INTRODUCTION

Throughout the past several years, single sourcing has been featured in talks at technical communication conferences, reported in technical communication journals, and described in many books. Single sourcing and knowledge management have spawned technical communication consulting firms that specialize in introducing new techniques and technologies to technical communicators. At the past several STC conferences, there has been an abundance of sessions about single sourcing: how to start and manage the project, which tools to use, and how it will solve problems and make our jobs easier.

Overall, the information about single sourcing has been overwhelmingly positive. Single sourcing can reduce costs, improve efficiency, save time, and increase quality. Many technical communicators adopt a single-source solution based on the favorable reviews about single sourcing. They accept the information that they receive from their peers and the industry as the whole truth.

However, a part of the discussion around single sourcing has gone largely unaddressed. There has been little talk about what might be wrong with a single-source approach to documentation development and management. Those studies that do address potential problems are written with the assumption that the decision to shift to a single-source documentation method has already been made.

Our aim is to encourage technical communicators and managers to weigh the negative aspects of the strategy before making a decision. (A few noteworthy exceptions to the lack of critical talk about single sourcing include the special issue of *Technical Communication* from August 2003 and the “Implications of Single-Sourcing” session at the 2003 STC Conference.) If they ignore the negative aspects of single sourcing, technical communicators and managers may encounter unexpected effects that hinder their documentation effort more than help it.

PURPOSE

Any talk such as this one is capable of producing a heated debate, if not an outright argument. Both the advocates and opponents of single sourcing are impassioned about the issue, precisely because it affects the working lives of technical communicators to a surprising degree. Therefore, we think it is important to point out that the aim of this paper is not to make a grand claim that a single-sourcing documentation strategy is somehow flawed or worse than more traditional methods. We agree that single sourcing can provide real benefits. The aim of this paper, then, is simply to raise issues that can be overlooked in this area and hopefully spark a discussion that would lead to technical communicators who are better equipped to understand the implications of a decision to change their documentation strategy, and, perhaps, provide a framework around which communicators on both sides of the single-sourcing issue can form a discussion.

For the purposes of this paper, we consider several different single-sourcing techniques. In most instances, we have defined single sourcing as its most highly evolved form. However, we occasionally broaden our definition to a more general definition that includes any technique that re-uses content. For example, one could argue that a simple letter composed in Word that is used as a default template for customer correspondence can be considered a single-source strategy.

For the most part, however, we are considering problems that can occur in a comprehensive, database-driven knowledge management system in which a large set of data may be used to produce many different documents in many different formats. In this more highly evolved form of single sourcing, text may be stored in SGML or XML source files only to be parsed and rendered in a PostScript, PDF, HTML, or other output format. We have attempted to clarify cases where the problems outlined below are specific to a particular level of single sourcing, by providing a specific example or description of each problem.

THE PROBLEMS

As we pointed out before, this paper attempts to do more than just identify the problems that can arise during or after the implementation of a single-source documentation strategy. Many books and consultants already exist to help documentation organizations make the change to a new paradigm. Instead, this paper identifies issues that should be examined *before* a single-source ideology is accepted in the first place. In our estimation, then, a single-source strategy to documentation development can:

- Add a new level of information management.
- Emphasize specialization over generalization.
- Introduce unnecessary overhead.
- Involve tools and technologies unrelated to writing.
- Use small, strictly defined units of text.
- Further separate appearance from content.
- Require a migration effort.
- Introduce stressful change.
- Require data management skills.
- Cost significant time to correct small mistakes.

Add a new level of information management

While single sourcing enables increased efficiency in the writing process because of content reuse, that efficiency comes at the price of increased information management. Instead of just writing and publishing content for the sake of the content itself, technical communicators who single source now have the overhead of analyzing how content must be chunked and designed most efficiently for reuse. The amount of management overhead depends on the nature of the project.

Projects that reuse large sections of text, such as entire chapters, require the least amount of information management. There is less concern about how the text will fit in with the text around it because it is a module in and of itself, although it is important to check text changes in every context.

Projects that reuse smaller bits of text, such as individual steps in a procedure or heading titles, require more thought and design and may be more difficult to manage. These smaller sections of text can easily be taken out of context. A change to the text that makes sense in one book may not make sense in another, although the original text was correct in both contexts.

In addition, managing all the text itself takes work, as stated by Robert Kramer in an article in *Technical Communication*. “There is extensive ‘mechanical’ or busy work in the management of files, conditional expressions, error tracking, and version control” (1).

Emphasize specialization over generalization

Many technical communicators are accustomed to a work day that may include writing, editing, page layout, graphic design, web development, project management, and other tasks. Single sourcing, however, separates many of these functions from the writing aspect of the document creation and encourages technical communicators to become specialists. In his introduction to the August 2003 issue of *Technical Communication*, Locke Carter writes that some technical communicators believe, “The build-up of skills that writers possess may have come to a head; writing may be de-coupled from information design, from layout, from grammar” (2).

In his article “Single Sourcing and the Technical Communication Career Path,” Michael Albers comments on this same problem and juxtaposes what he calls the “craftsman” model of document production with the “specialist” model of document production (3). He, too, takes issue with the shift that single sourcing is imposing on technical communicators, and outlines a solution that takes into account writers’ craftsmanship. Albers’ article is essential reading for any manager considering a shift to a single-source paradigm.

Many types of single sourcing focus on writing bits of reusable text rather than writing the text most appropriate for a particular context. As a result, technical writers become content developers following a formula for writing documentation to fit the data into pre-defined blocks. Writing for context and flow may give way to the importance of content reuse and structure. Technical communicators may resist giving up the relative freedom and variety that many non-single-sourced workplaces offer.

Involve unnecessary overhead

Technical communicators should analyze their document libraries thoroughly before contemplating any type of single-sourcing project. Content reuse is one of the strongest arguments for single sourcing. Single sourcing should probably be considered if, for example, a company has:

- Training materials, user’s guides, and help systems that use the same content.
- Multiple products with the same basic functionality that need similar documentation.
- Customized document sets for each customer.

If each piece in the company’s document library is unique with very little content sharing, the overhead of introducing single sourcing may outweigh the benefits. Single sourcing may require a high level of information management, the expense of new tool sets, and the learning curve associated with new, more technical skill sets. If single sourcing does not make sense for a simple document set, this overhead is unnecessary.

Involve tools and technologies unrelated to writing

In many cases, single sourcing requires that technical communicators learn tools and technologies that are not directly related to their writing. This is not a new concept; desktop publishing has forced writers to become familiar with formatting issues that are not directly related to writing. Single sourcing, however, brings these new tools and technologies to a more technical level.

When single sourcing with XML files, for example, technical communicators must not only learn how to read and write XML, but they often also have to learn how to use an XML editor, a content management system, how to create or use a document-type definition (DTD), as well as a myriad of other applications and technologies that have traditionally resided outside the expertise of a technical writer. The issue becomes, then, whether it is wise to discard the kinds of skills and abilities that technical communicators possess (like writing and design skills) to favor strictly technical abilities. Although we do not question the ability of technical communicators to do technical work, we do want to question the wisdom of removing any more of the writers' focus from the content.

Use small, strictly defined units of text

Technical communicators have grown accustomed to writing in terms of chapters or sections (in the case of manuals) or in terms of topics or HTML pages (in the case of online help systems). In a single-sourcing paradigm, however, writers are required to think of their writing in terms of small, strictly-defined information units that can be separated into XML. Single sourcing forces writers to consider that, in some cases, the sentence they write in one location may be used in another context in a different location. In an article in *Technical Communication*, Ann Rockley writes that "Technical writers need to understand how information can be used in multiple ways as they write to ensure their content is reusable" (4). Each sentence (or paragraph or section, depending on the level of single sourcing) must be completely self-contained.

As a result, technical communicators must always keep in mind the XML hierarchy and its rules, which can be complex for these small information units. This can hinder the writing process, which can be more free-flowing when it is not inhibited by such great attention to structure.

Further separate appearance from content

Many technical communicators see their work as a marriage between text and the appearance of the text. With desktop publishing tools, technical writers have become experts and capable masters of page layout and design. The design and presentation of information is arguably as important as the content itself. Technical writers who are accustomed to being in control of the presentation of the data are limited by a single-source environment. Specifically, they must divorce themselves from the idea that they are in control of every widowed or orphaned line of text or every awkward line break.

Proponents of a single-source approach to documentation contend that the separation of text from the context of its visual layout actually creates a more efficient, preferable division of labor. This presumes that technical communicators would prefer to write without the burden of designing the page or creating a visual representation of text. However, it is just as likely that technical writers prefer and even enjoy the act of using their design skills to present information both textually and visually. Typically, a single-source solution does not allow the amount of freedom that gives communicators complete control over the appearance of their documents.

Although many documentation groups do, in fact, use FrameMaker templates or CSS to control the visual layout of the HTML, the use of these tools does not remove the writer from the visual design to the degree that a single-source documentation paradigm does. It is one thing for a writer to actively interact with a template and see (in real time) how their words appear in a document and another to write text separate from its appearance. Although there has been no research in the area, it would be interesting to observe whether writers regularly edit text in a rigid template-governed writing situation to control the way that text appears in a document.

Require a migration effort.

While there are some processes that minimize the effort required for the migration of documents to a single-sourcing environment, a shift in documentation paradigms typically means that legacy documentation must be migrated to a new system. Even if a documentation group plans to begin using WebWorks Publisher with their existing FrameMaker documentation, hunting down poor paragraph style consistency and other idiosyncratic issues within documents can be time consuming. In addition, the paragraph formats applied to a FrameMaker document may not readily map to the chunks of information that a group plans to use in its single-source definitions.

This re-architecture and reorganization of information requires a large effort. This effort is time spent away from content development and may result in the neglect of urgent writing projects. In a workplace environment in which writers often do not have time to complete all of the projects in their queue in a timely manner, writing groups could be forced to postpone pressing documentation projects in order to complete the initial work required to translate an entire library to a new writing paradigm.

Introduce stressful change

As with any organizational change, the adoption of a single-sourcing strategy for documentation can put additional stress on members of a documentation team. Aversion to change is not unique to technical communicators, but a shift to single sourcing does represent a large-scale change in the way writers work. This dramatic shift may introduce an unexpected level of dissatisfaction in a writing group and, along with it, a dip in morale and productivity.

Locke Carter's introduction to the August 2003 issue of *Technical Communication* refers to the research in the area of change management in business journals that may offer more insight than is typically available in the field of technical communication (5). Any manager interested in rolling out a single-source strategy should understand the implications of a large-scale change in work strategies so that they might anticipate and minimize the effects of a negative response.

Require data management skills

In addition to single-source authoring and management technologies, technical communicators must become experts in data management. Data management typically has been the domain of programmers and other IT professionals, and technical communicators usually have less experience in the area. What little experience writers do have might come from source management systems, but even those systems pale in comparison to the complex architecture of knowledge management systems. One change in a piece of content can have a ripple effect throughout the system, and technical writers must be aware of how their work affects the system as a whole. This means that communicators must understand how text in one file may be used in multiple documents.

While some knowledge management systems seek to automate or alleviate the burden of data management, technical communicators must still understand the (perhaps unfamiliar) concept of automating document creation or "building" generated documents from disparate units of data.

In his article in the August 2003 issue of *Technical Communication*, Robert Kramer describes the complexity of text reuse. "The complexity this adds to the writer's task comes in the form of troubleshooting why certain conditional statements fail, how to organize extensive texts that must be valid for multiple, unique versions; how to control versions of texts where a new feature may only be true for the current version of a product, but only on a few platforms" (6).

Cost significant time to correct small mistakes

On one occasion, a technical writer excited by the new single-source documentation paradigm she was working under reported that single sourcing is great because it only takes an hour or two to update many deliverables if a large section of content changes. When asked how long it takes to correct a comma error, the same writer reported that it takes about the same amount of time. In other words, an error that could be completed in a short amount of time in a more traditional documentation environment actually takes longer in a single-source environment.

The reason this unusual situation can occur in some single-sourcing environments is that it takes some amount of time to build deliverables from the source of data. That means that while a source document may only need to be updated once to make a large change, a small change still requires the same amount of effort to generate deliverables.

CONCLUSION

While the ten problems listed above may not offer a comprehensive list of issues that can arise in a single-source environment, it does, we hope, help technical communicators consider the implications and results of a decision to change the way documentation is developed. If nothing else, we hope to encourage a cautionary stance toward adopting a new technique. Documentation teams should question, for instance, whether delivering multiple formats is better than delivering the single best format for a particular audience. In addition, communicators should weigh the benefit of maintaining information that resides in several deliverables against the cost of managing the data in a different way. Only after technical communicators understand the costs associated with single sourcing can they fully experience the benefits.

REFERENCES

(1) Kramer, Robert. “Single Source in Practice: IBM’s SGML Toolset and the Writer as Technologist, Problem Solver, and Editor.” *Technical Communication* 50.3 (2003): 333.

(2) Carter, Locke. “The Implications of Single Sourcing for Writers and Writing.” *Technical Communication* 50.3 (2003): 319.

(3) Albers, Michael J. “Single Sourcing and the Technical Communication Career Path.” *Technical Communication* 50.3 (2003): 335.

(4) Rockley, Ann. “The Impact of Single Sourcing and Technology.” *Technical Communication* 48.2 (2001): 191.

(5) Carter. “Implications.” 319.

(6) Kramer. “Single Sourcing.” 331.

Dave Yeats
Instructor
Department of English, Texas Tech University
P.O. Box 43091
Lubbock, TX 79409
806-742-2501

Dave Yeats is a PhD student in the Technical Communication and Rhetoric program at Texas Tech University. In addition to issues in documentation management, his research interests include usability, open source software development and documentation, and technical communication pedagogy.

Heather Hull
Senior Technical Writer
Motive, Inc.
12515-5 Research Boulevard
Austin, TX 78759
512-531-2437

Heather Hull is a senior technical writer at Motive, a leader in the service management software industry. She has been in the technical communication industry for nine years and has experience with writing, editing, training, and project management.