Content Management and the Separation of Presentation and Content

Dave Clark
University of Wisconsin-Milwaukee

The importance of separating presentation from content is taken as a given in many kinds of publishing, despite the fact that the notion of separation has received little critical scrutiny. I provide a closer look at the separation, first by providing contemporary and historical context, then by laying out key distinctions in the ways the separation argument is used in Web design versus Web content management versus full-featured content management systems (CMSs). I suggest that these distinctions are critical in how we should view the separation and the implications of the separation for the work of technical communicators.

The importance of separating presentation from content (also discussed as "form from content" or "design from content" or "semantics from presentational markup") is a baseline assumption of many kinds of publishing, including Web design and development, desktop publishing, and content management; a Google search for "content from presentation" yields tens of thousands of hits, as does "presentation from content." Advocates of the separation argue that writers and designers should store the text, images, videos, etc., that make up a document (e.g., the writing, charts, and tables of an academic article) separately from the larger aesthetics of that document (e.g., *TCQ*'s page size, font selections, and spacing standards). Creating a separation of this kind allows content and presentation to be linked only loosely and altered independently of one another. For example, in Web design most developers now use a combination of Cascading Style Sheets (CSS) and templating that allows them to rapidly change a site's look and feel without impacting the site's writing and multimedia files. Where changing the style of a site's level 1 headings (say, from 14 pt, green, Helvetica to 16 point, blue, Verdana) once meant manually editing each of a site's HTML files, now developers can simply change the heading's entry in the style sheet and the change is automatically propagated across the site.

The advantages of a separation, then, can be compelling. In content management, whether we are discussing website content management, small, in-house single sourcing systems, or enterprise-wide content management (ECM), separating content from presentation can not only save time but allow for rapid reuse and repurposing of content. A single piece of content, properly marked and stored, can automatically and simultaneously appear in user manuals, help files, and press releases that can in turn be automatically altered to appear in print, on the Web, or on mobile devices. Once initial designs are created, fonts, colors, and layout are added on the fly for the specifics of each genre and/or medium, and with, for example, a simple change to a style sheet, aesthetic changes can easily be applied to past as well as future documents, making it easy to maintain organizational consistency.

The separation is foundational to content management, a fact that can create philosophical and cognitive dissonance for technical communicators trained to think of information as content that is inherently linked to presentation. Design scholars both within and outside of our field have long argued for a whole document approach, in which content and presentation must be designed simultaneously to be rhetorically effective. Design should be thought of not as a look and feel imposed on top of content but as a rhetorical melding of form and function (cf. Buchanan, 1989; Kostelnick & Roberts, 1997; Schriver, 1996). In making this argument, these scholars are tapping into literally thousands of years of similar debate: Rhetoricians, theorists, designers, artists, and architects have long debated the relationship between content and aesthetics, rhetoric and ornamentation, form and function, and medium and message.

Of course, in any vision of the separation, content and presentation may be stored separately but are ultimately reunited. Arguably a well-designed system could produce outputs that are no less rhetorically effective than individually crafted documents, and at the same time could provide cost and time savings as well as fascinating rhetorical challenges for technical communicators. But designing such a system is no easy task, and I suggest that it is at least strange that a presentation/content separation is being so widely implemented with so little talk of the rhetorical assumptions that underlie it and the implications of those assumptions for our work. There are a few exceptions: Web designers like Stein (2000), Meyer (2003), and Cohen (2004) have argued for a more nuanced understanding of the separation, suggesting, for example, that presentation should be conceptually separated from structure in these discussions, that structure and content must be linked, and that there are cases in design when content, structure, and presentation are truly inseparable:

> My aversion to a rule-of-thumb separating style from content comes partly from a belief that some style is content. Not style to attract attention, but style to inform. Now, there is much art in attracting attention and I don't mean to belittle that power, but there is also art to communicating well in visual media. In Tufte's Envisioning Information, quite the opposite of separation occurs. Presentation makes sweet love to

structure when conceiving an informative vision: "To envision information—and what bright and splendid visions can result—is to work at the intersection of image, word, number, art." This is design inextricable from authorship. It's style that cannot be dissected from content without bleeding away informative power. (Stein, 2000)

But in general what discussion exists is devoted to advocating the separation and to the pragmatics of achieving the separation if, for example, a communicator must update legacy documents or struggles with certain types of content that are not easily separated from their presentation (e.g., displaying complex math equations on systems without the capability).

In what follows, therefore, I have worked to scrutinize the separation. I have begun by narrowing my focus: *Content management* can be a slippery category, so I have defined what I mean by content management and described the types of content management that are the focus of this article. Next, I have suggested that historically, different scholars, fields, and practices have drawn lines between presentation and content in different ways and with different implications. The question then became, How do content management systems separate presentation from content? I have provided answers to this question and followed those answers with a discussion of the affordances of those answers, exploring the real impacts of content management's separation on the work of technical communicators.

In this article I have not argued from a Luddite or substantivist viewpoint (i.e., the separation is evil and bad), and not only because doing so would seem futile; there are hundreds of blog postings, articles, books, and vendor white papers devoted to either touting the separation or accepting it and moving on to discuss the problems of implementation. I am not, after all, convinced that attempts to separate content from presentation are wrong. I simply believe that we should investigate the separation on its own terms, interrogate what it is that designers and vendors mean when they talk of the separation, and consider the separation's implications for technical communication.

DEFINING CONTENT MANAGEMENT

I must start by defining what I mean by *content management*, because doing so is not an easy task. The term is often used sloppily, covering enough ground that it can be easily dismissed as another buzz phrase (as is sometimes true with *knowledge management*) rather than a serious or specific enterprise:

It seems that a number of different technology disciplines are currently converging under the generic term of content management (CM), blurring the boundaries between them. Vendors do not help the picture by offering complementary and often overlapping functionality all branded as "content management." …

The term content management has been a constant source of ambiguity. It is being used to describe both the management of the content within these intelligent objects (e.g., the management of the authoring process of a report) as well as describing the management of the objects themselves in the context of a larger system (e.g., the composition of a Website). (Parapadakis, 2000)

For the sake of clarity, I draw a distinction between *content management* and *content management system*. Certainly, some definitions of content management incorporate the tools as well as the process, but I find it helpful to think of *content management* as referring to process, as we can see in this definition from Parapadakis, who defines content management as

The process of managing electronic content through its lifecycle—from creation, review, storage and dissemination to destruction. (Parapadakis, 2000)

JoAnn Hackos also focuses on the process in her definition, seeing content management as, at heart, an ongoing process of planning, evaluating, and revising an organization's information models, a process that should be undertaken in any organization "large enough to have more than one file cabinet":

If you find yourself asking: Where is the right information? How was it categorized and filed? Is it in the right form for reuse in a different context? How do we get it from a proprietary authoring system into other media? then, you need content management. (Hackos, 2002, p. 1)

Hackos further sees content management as an ongoing process of continually improving an organization's management of its content:

Content management provides a process for answering your questions by encouraging you to categorize and organize information for future retrieval and development. Grounded on a vision of the user experience, content management focuses not only on keeping track of information assets and making them accessible to users, it also focuses on finding new ways to deliver information to customers, employees, and business partners who need it. (p. 2)

In a time when most of the content in organizations is produced electronically, accomplishing these feats means developing processes, standards, and tools for the electronic use, storage, and retrieval of content. Hackos advocates supporting an organization's information model with XML (eXtensible Markup Language), a free, standardized, customizable markup language that allows content to be "interpreted and rendered by any presentation (publishing) system or by any business process" (Hackos, 2002, p. 68):

XML means using a non-proprietary language to label information modules; it means separating content from presentation; it ensures that modules are consistently structured. Most important, it provides a tool for the future, allowing businesses to present information in new ways, through new channels—many not yet developed. (Hackos, 2002, p. 2)

In any case, the technological means enabling content management is most often a "content management system," a digital technology (software and sometimes hardware) that assists in content management by simplifying and automating the processes of the "creation, approval, storage, retrieval, versioning, re-use and delivery of content objects" (Intentional Design, 2006). A little digging turns up dozens of overlapping and competing taxonomies describing the types of content management systems, including:

- **Document Management System (DMS).** To purists, a document management system is not really content management, because its focus is at the level of the whole document. A DMS manages the storage, tagging, browsing, and versioning of discrete documents in multiple formats, associating each with metadata (data linked to a document that describes that document; e.g., "Finance," "3rd Draft," "October 13, 2005"). Arguably, Flickr (an online photo-sharing site) is a type of DMS, in that it exists for and excels at providing tools for managing, browsing, and searching sets of files, and for providing unique ways of linking files in unexpected ways.
- **Web Content Management System (Web CMS).** Web content management is devoted specifically and solely to providing tools for the creation, presentation, and maintenance of website content. Web CMS includes everything from the back-end text and display handler of a blog to full-site creation and management tools like Drupal to the enormous applications that back user-customizable portal sites like Yahoo!
- **Content Management System (CMS).** Following Hackos's advocacy of XML-based content management, I would define a CMS as a system that approaches the problem of content management by using markup, metadata, and tools to break documents into component parts, to a level of granularity (e.g., paragraph level, sentence level, word level) set by organizationally defined information models, and labeling each part with metadata that describe its meaning and relationships to other content. The same content can then be automatically assembled in different genres, with different presentations, and in different media.
- **Enterprise Content Management (ECM).** Enterprise Content Management is an often-massive attempt to combine the functionalities of Web, document, and content management and to systematically incorporate not only

traditional publishing activities but e-mail, financial records, human resource documents, etc., for an entire organization.

The present discussion will be most clear if I use a bird's-eye taxonomy that allows me to highlight the types of CMS that are the focus of this article: Web CMS and CMS, the two varieties of content management most inclined to prompt claims of the glory of separating presentation from content. For the sake of sanity and brevity I would exclude varieties of content management specific to certain types of activity (e.g., learning management systems) and will not attempt to tackle ECM. The key distinction I wish to make here is about DMS versus Web CMS and CMS. A document management system by definition preserves documents intact (the DMS at my last job stored all documents in presentation-friendly PDF). A DMS is at base an electronic filing cabinet that replaces tabbed folders with sophisticated browsing and searching capabilities; it still sees documents as stand-alone entities with content that is locked to a particular presentation. Web CMS and CMS, on the other hand, share a devotion to the separation of presentation from content. Content is a thing to be created, stored, and managed, and presentation is a thing to be added just in time for the content to appear in a form suitable for human use.

As a result, Web CMS, CMS, and increasingly Web design in general sometimes require us to embrace a new relationship between content and presentation, and as writers we have to learn a host of new tools, techniques, and technologies in order to negotiate this relationship. On the one hand, it can be difficult to wrap your head around this new way of thinking. As writers, we have been trained for generations that content and presentation are tightly linked; while we may wait until we finish writing to clean up our format, the end result is a relatively static document, and changes to presentation affect content and vice versa. On the other, from the perspective of a writer working within a well-designed infrastructure, the experience could be similar to working in an organization with a strongly enforced style guide; the Word and Dreamweaver templates we were accustomed to using are simply replaced by different tools with different kinds of style enforcement.

And there are good reasons for embracing the separation. Separating content from presentation can not only expedite creation, revision, and reuse, but, as Hackos and others suggest, is a look to the future (Broberg, 2004). For example, Web authoring is subject not only to site-specific decisions and organizational style sheets but to the decisions of browser developers and standards organizations such as the World Wide Web Consortium (W3C), a prominent international organization dedicated to developing standards, protocols, and guidelines (including the official specifications for HTML and XML). Their goal is the promotion of Web "interoperability," making the presentation of Web documents independent of hardware platform, software, culture, disability, etc. (W3C, 2007a), and that goal requires that content and presentation not be hardwired to one another. They seek

to make Web content broadly available, and view interoperability as critical to the Web's reaching its "full potential" and ensuring "long-term growth" (W3C, 2007b).

There is also a common sense to the separation. As I suggested above, technical communicators, particularly scholars of information design like Kostelnick and Schriver, have a long history of arguing for the integration of verbal and visual rhetoric. But at the same time, our scholarship, textbooks, curricula, and everyday speech frequently oppose content or writing or information to aesthetics, layout, or design, and the result is obviously not always documents that have a disconnect between the rhetorical function of their content and the rhetorical character of their design. Still, I suggest that it matters how we talk about such things, and on its face the call for a fundamental, technological separation of presentation from content is in tension with our historical interest in creating documents that unify content and presentation, and increasingly we are hard coding that tension into CMS and Web CMS. It is worth exploring this tension, considering what it means to separate presentation from content. Can it be done? How is the separation achieved, and what are its implications? How will the separation make us think and work differently?

CONTEXTUALIZING THE SEPARATION

To begin considering the separation, we must first think of the language that we employ when we talk about it. As Bowker and Star (2000) suggest, "[t]o classify is human"; "a 'classification system' is a set of boxes (metaphorical or literal) into which things can be put to then do some kind of work—bureaucratic or knowledge production" (p. 10). We tend to reify a commonsense distinction between presentation and content in our work and speech, a distinction that structures and is structured by the educational and organizational practice of separating, say, graphic design from writing. In technical communication, we create courses and texts that, while they may argue for the integration of content and presentation, also reinforce the distinction by calling out document design as a specialty, continuing our tendency to talk about content as the core of what we do and the presentation of that content as secondary.

But what does it mean to separate the two, and where do we draw the line? Content versus presentation is a blurry classification, hardly a standard that is agreed upon by mutual fields or authorities, and what it means to present content can vary depending on what we are talking about, who is doing the talking, and what they mean by presentation and content. For example, in political discourse, there is a tendency to think of presentation as rhetoric, separating the content of what is said (the truth) from how it is presented (the rhetoric); thus, the Straight Talk Express and the No-Spin Zone. This separation, of course, is ancient; as Gideon Burton (2007) notes in *Silva Rhetoricae*, rhetoricians have long created and used a distinc-

tion "between *what* is communicated through language and *how* this is communicated":

> Aristotle phrased this as the difference between *logos* (the logical content of a speech) and *lexis* (the style and delivery of a speech). Roman authors such as Quintilian would make the same distinction by dividing consideration of things or substance, *res*, from consideration of verbal expression, *verba*.

The result of this "always … artificial and conditional" separation, Burton argues, was the *highlighting* of the interconnectedness of *res* and *verba*:

> Thus, rhetoricians divided form and content not to place content above form, but to highlight the interdependence of language and meaning, argument and ornament, thought and its expression. It means that linguistic forms are not merely instrumental, but fundamental—not only to persuasion, but to thought itself.
> This division is highly problematic, since thought and ideas (*res*) have been prioritized over language (*verba*) since at least the time of Plato in the west. Indeed, language is a fundamentally social and contingent creature, subject to change and development in ways that metaphysical absolutes are not.

This same discussion, of course, is at the root of the ongoing debate between the views of Socrates and those of the sophists over the place of rhetoric: Is rhetoric foundational or is it merely reality's window-dressing? The sophists saw truth as relative and as available to us only through language, so it made little sense to talk about rhetoric as external to a foundational truth; rhetoric itself was the only foundation. Protagoras, for one, seemed to deny that "anything truer than conflicting opinions existed" (Billig, 1987, p. 43), a point of view that has historically been characterized cynically as moral and ethical relativism, as opposed to radical contextualism (Porter, 1998, p. 99). Socrates, for his part, refers to oratory rhetoric as "pandering to the feelings of the mob" (Billig, 1987, p. 32), and refers to rhetoric in general as "an occupation which masquerades as an art but in my opinion is no more than a knack acquired by routine" (p. 33). Rhetoric was mere "jingle-making" (p. 33), grandstanding that emphasized presentation over content and was concerned not with discovering eternal truths, but with winning arguments at any cost. In the *Protagoras*, Socrates' denial of the importance of rhetoric creates a space outside of rhetoric and language, an external, fundamental truth for which rhetoric is the presentation. It is not the presentation imposed on top that matters, but exploring the truths underneath.

As I have suggested below, however, the separation between truth and its presentation envisioned by Socrates only weakly parallels the type of separation enacted by content management systems, and more contemporary separation discussions are more directly useful. Marshall McLuhan, for example, thought that the

medium used for communication was of critical importance to learning and culture. His famous, counterintuitive argument that "the medium is the message" seems to suggest that the means of presentation is actually more important than the content being delivered, but ultimately his point is that they are fully integrated (McLuhan, 1994). Neil Postman, another medium theorist, spins this assumption into a grim assessment of contemporary culture, which he argued was in a state of decline due to the growing emphasis on the unidirectional, passive viewership of television over the complexities of the written word (Postman, 1985). In *Technopoly*, a staple of graduate programs in the mid-90s, he further suggests that our widespread, uncritical adoption of computers led us to see all human problems in terms of information and data. Because computers primarily offer enormous efficiency and calculating power, they lead us to develop solutions that rely on the production of overwhelming amounts of information and statistics, creating an "information glut" that swamps formerly coherent and humane cultural narratives (Postman, 1992).

In architecture, Louis Sullivan's edict that "form follows function," that the design of a building should link tightly with its purpose (as opposed to following convention or classical style), is often cited as a milestone in the Modernist quest to unify form and content. Similarly, in the fine arts, Modernist art critic Clement Greenberg championed Jackson Pollock and "medium specificity," an artistic approach that reacted to realist attempts at accurate representation by instead embracing abstraction and producing paintings that concerned themselves primarily with their own media. Painting, to Greenberg, was about the fusion of content and presentation in ways "unique to the nature" of the canvas and paints being used (Greenberg, 1982). To reverse the point, one might argue that representational painting "would be a place where form is most separated from content. (This is how Greenberg saw it.)" (Lane Hall, personal communication, 2007)

The notion of form following function is a useful parallel with the design practice in technical communication. Despite the common practice of creating curricular separations between writing courses and courses like document design, we have typically argued that presentation and content are by definition rhetorically integrated. The work of scholars like Charles Kostelnick, Mary Lay, Karen Schriver, and hosts of others is founded on the assumption (sometimes explicitly argued) that the conventions, strategies, and practices of design are rhetorical in the sense meant by the sophists, not Socrates. As practiced by technical communicators, document design is not a sliding scale with ornamentation on one end and the truth on the other. Document design is an explicitly rhetorical practice that argues that both content and presentation are inherently linked, even as it emphasizes design through its very name and existence.

Finally, however, even in practical terms it is difficult to maintain the distinction between presentation and content. As designer Robin Kinross (1989) argues in

"The Rhetoric of Neutrality," even the most purely "aesthetic" designs, on the one hand, and the most "neutral," on the other, are rhetorical constructions that do not allow us to draw clean lines between what is being communicated and how it is being communicated. Kinross takes as his starting point Gui Bonsiepe's assertion that information designers need to remember that everything is rhetorical, which, he says, he closely follows with the assertion that rhetoric-free constructions are possible—his example is train schedules, which are an "example of information innocent of all taint of rhetoric." Kinross then demonstrates the rhetoric of train schedules, albeit as a narrowly defined rhetoric-as-eloquence rhetoric. He argues that the train schedules are designed, finally, to say something about the organizations that produced them. They are arguments for organizational ethos, in the same way as a corporate brochure or annual report. He concludes by suggesting that his article

> is intended to serve as a simple reminder that nothing is free of rhetoric, that visual manifestations emerge from particular historical circumstances, that ideological vacuums do not exist. . . . among information designers, there has been a tendency to escape from the assaults of the wider world, to deny any idea of rhetorical persuasion, and to take refuge in immaculate black machine casings. (p. 142)

He suggests, then, that no presentation is free of rhetorical content. I have suggested as a corollary that no content is free of presentation, even in practical terms. Content and presentation are never separated, because even the most poorly formatted Notepad document has a presentation: layout, fonts, paragraph breaks, capitalization, headings. Authors writing for content management work with interfaces that offer authoring-specific presentations of their content. And even on opening the database of a content management system and examining the raw text contained in a content management system, one would still be examining a rhetorical presentation of an organization's data, albeit one that was not very helpful to most human users.

HOW DOES CMS SEPARATE PRESENTATION FROM CONTENT?

If there is, then, no *true* separation of presentation from content, and a long history of arguments about how they should be defined and related to one another, the obvious question to ask about content management systems is what they mean when they talk of a separation. To start, content management systems classify both the "truth" and "rhetoric" in Socrates' formulation as content, not being concerned with drawing distinctions between truth and rhetoric or *res* and *verba.* A piece of

writing, loaded with elements of both what Socrates would classify as "truth" and the persuasive elements he would consider "pandering," would all be classified as content, and as a result if Socrates were alive and literate, he could find plenty of CMS-delivered content that contained elements he would classify as presentational "jingle-making." The look and feel of documents does not even make an appearance in his discussion, if only because of his deep distrust of the technology of writing. Content management systems make a separation argument that is more similar to that of architects, artists, designers and technical communicators, who argue about what it means to effectively integrate meaning with media. In content management, however, the argument is made technologically, and shows little knowledge of the past debate; as a result, the separation takes on a unique character that can have real effects on the work of technical communicators.

In computer science, the origin of the separation of presentation and content is arguably the concept of the *separation of concerns*, which advocates the separation of program elements into distinct, independent features that can then be separately coded and maintained, more easily understood by outsiders, and allow for easier troubleshooting, as failure of one segment will not automatically lead to failure of another. Most programming languages use a variety of this separation, whether through object orientation or by encouraging the structuring of code into distinct processes and functions. Although *separation of presentation and content*, like *content management*, is a loosely defined phrase, I suggest that in talking about the separation of presentation and content it is useful to think of the separation in two distinct ways:

- **Content is complete texts. Presentation is output structure, navigation, and visual style.** In this formulation, most common in Web design and in more rudimentary CMS and single sourcing, presentation is a content wrapper that includes navigation (e.g., links, table of contents, indices), output layout, and the visual style for the content. Content, the authored text and images of a genre (blog entry, press release, home page, etc.), is stored as a unit. Because it is stored as a unit, the text has a predefined structural presentation (headings, paragraphing), and visual style (font choices, color, etc.) is linked to content via that structure.
- **Content is content modules. Presentation is output structure, navigation, visual style, and genre definition.** Rather than being stored as complete documents, content is broken into component parts, at a level of granularity defined by information models (e.g., at the section, paragraph, sentence, or even word level). Content is then divorced from a document-centric structure and can be assembled in myriad ways. Presentation, then, includes the process of building custom documents, creating genres of documents on the fly based on organizational definitions.

Content Is Complete Texts. Presentation Is Output
Structure, Navigation, and Visual Style.

Not surprisingly, when computer scientists encountered the many problems of Hypertext Markup Language (HTML), a language that historically has allowed (or even required) designers to meld presentation and content through the use of font-, color-, and layout-specific tags, they sought to create a clear separation that would free HTML-tagged content from a particular page structure and visual style. In developing Cascading Style Sheets (CSS), they chose to draw the line between the unique, structured text of a page and the structure, navigation, and visual style of the page on which it appeared.

Consider a brief example from Web design. While true geeks have always insisted on writing their own code using only a text editor, WYSIWYG applications like Microsoft FrontPage often produce HTML code that binds a page's content to a particular presentation. A warning on a Web page might be coded like this, using a <font> tag:

```
<font face="arial" color="red" size="3">Do not stick a fork in the toaster.</font>
```

This warning would, then, be displayed in roughly the same way in all contexts, taking into account the properties of the browser, printer, or mobile device on which it was presented. By contrast, a developer using a Cascading Style Sheet might code the same line like this:

```
<div class="warning">Do not stick a fork in the toaster.</div>
```

The class "warning" can then be linked to one or more style sheets that define what text labeled "warning" should look like. For example, a simple style sheet might define the presentation of a "warning" like this:

```
warning {
font-family: Verdana;
font-size: 16px;
font-weight: bold;
font-color: red;
}
```

The separation of presentation from content here, then, means the logical, textual separation of visual style from the content to which is it applied. Previously, each piece of text was wrapped in a specific set of style commands, but with CSS the style information for a warning need not appear in the same place, or even the same document, as the text of the warning. This new flexibility allowed for differ-

ent text labeled "warning," whether on the same page or elsewhere on the site, to have the same presentation, and changes to the style information (say, from red to blue) change all of the warnings automatically.

As a result of developments like these, the <font> tag is rapidly becoming a thing of the past. The standards organizations and browser developers, who collaboratively structure the use of HTML, CSS, and other Web languages, have long since decided that presentation-specific content tags were a step backward in markup practice. The newest version of HTML deprecates the <font> tag; sites using it will work for the time being, but eventually browsers may stop supporting it altogether. This functionality, of course, closely resembles the capabilities of basic word processing, which also allows for the creation of styles that can be applied within and across documents. Version 2.0 of CSS, however, adds the possibility of conditional presentation, meaning that a designer could make a warning appear small, red, and Arial when printed, and huge, blue, and Times on a cell phone.

Still, so long as a website's pages are relatively static, meaning that the text unique to the page is delivered in a large chunk rather than through the automated assembly of component content modules, the separation of presentation from content remains the separation of text and images from the formatting to be imposed on top. This is the model of the most ubiquitous types of Web CMS (e.g., blogging software). Authors enter content into a Web-based form and click a Submit button, after which the content is stored in a database as a piece, without regard for the kind of content it represents. All content is treated identically, and presentation varies only when specific pieces of the text are specifically styled (as with a warning).

Web content management systems tout their ability to separate content from presentation, so they naturally avoid presentation-specific tags such as the <font> tag. But they still have their limitations in terms of creating a clean separation between presentation and content. By definition, a Web CMS generates HTML and CSS code for display in Web browsers, so its separation of presentation and content is necessarily limited by the need for both presentation and content to be linked to structural HTML tags. To avoid having content that is presented as a long, unbroken gray line, that content must be tagged with tags (e.g., <h1>, <div>, <span>), whether by including those tags in the database with the content, or by automatically inserting the content into a preexisting tag-filled page. Because CSS works by applying presentational preferences to individual tags, presentation, too, is inherently linked to structure.

As a result, the changes that can be made to a Web page's presentation are limited, because while a single, content-filled page can easily have multiple presentation styles imposed on top of it, changing the appearance of headings, page layout, and the positioning of page elements (e.g., navigation bars and text blocks), the underlying structure remains as a mediator between presentation and content. Consider, for example, the fantastic examples at http://www.csszengarden.com/, a site

devoted to demonstrating the "complete and total control over the style of a hypertext document" that CSS can provide; dozens of designers have provided radically different presentations for the same content page, but the granularity of their changes is limited by the decisions that were made in producing the underlying HTML; the paragraph relating to "Benefits," for example, can be presented only as a complete unit.

Content Is Content Modules. Presentation Is Output Structure, Navigation, Visual Style, and Genre Definition.

The act of dividing and chunking content for separate storage and labeling, however, redefines the relationship between presentation and content. While vendors of Web design and content management products speak in very similar terms about the separation of presentation and content, the reduction of documents to component parts means that presentation is no longer conceived of as being about aesthetics, but about genre formation and production, as the production of an organization's documents becomes a matter of assembling content modules. Content is stored in individual chunks and associated with data that describes not necessarily how it is to be structured, but the kind of content it is. Most content management systems use tagging languages like SGML or XML to mark up content for storage. These tags define only the content of what they tag; in this way, traditional authoring, which included designing or at least managing presentation, is turned into structured authoring, which focuses on "using a tagging language to identify elements of the document based on their content rather than on their appearance" (Hackos, 2002, p. 68).

Once an organization's designers have created an information model and standard rules for tagging, writing becomes a process of creating information that fits into large-scale, carefully designed semantic structures that clarify the relationship between the content module and the others produced by the organization. For most organizational information models, a tag like "warning," which might be adequate for formatting a small website, would be hopelessly inadequate and replaced by something more specific and meaningful to authors. For a company that manufactured appliances, this warning might be relevant to all toaster materials that included warnings.

    <legal_warning_toasters_all>Do not stick a fork in the toaster.</legal_warning_toasters_all>

Other types of warnings might apply only to specific models:

    <legal_caution_toasters_X10>The X10 is not designed for bagels.</legal_caution_toasters_X10>

Using tools that can check the XML against rule sets and then use style sheets to output the XML in a variety of genres and formats, this legal content can be automatically presented whenever it is relevant to the particular materials being examined. The same content module could appear on every page of a website and in the small print of the manual. Should the legal department require a wording change, the content would only need to be changed once, in the content management system, to update all the documents that use it.

In this simple example, the separation between presentation and content remains much the same as with other content management systems. A key distinction arises, however, if we consider a more extensive example of how content might be reused in different genres. Consider this example from the Rockley Group's *Managing Enterprise Content*, where content is tagged generically, under the assumption that different pieces of content can be applied to different genres with little or no alteration. In this table, we see portions of three documents that serve different purposes but in places share similar rhetorical functions, such as providing a basic description of a product.

To me, this is a great example that makes a separation of content from any single genre compelling. Product descriptions should be consistent, and the differences here are so minor that they could easily be cleaned up and regularized; here Rockley, Kostur, and Manning use "nested reuse," which provides an identical description to all three documents but adds additional detail only when needed. All three documents get the first chunk of content:

<The B-Brother model 1984 is a device that connects directly to a consumer's television to track their television watching habits.> [Brochure, Operations Guide, E-Commerce]

But only the brochure and operations guide get the second:

TABLE 1
Example from Rockley Group's *Managing Enterprise Content*

Brochure	Operations Guide	E-commerce Site
The B-Brother model 1984 is a programmable device that connects directly to consumers' televisions to track the channels they flip to, what programs they record, and what commercials they skip. The information is instantly transmitted to the cable or satellite provider.	The B-Brother connects directly to consumers' televisions. It can be programmed to track what channels they flip to, what programs they record, and what commercials they skip. The information is transmitted to the cable or satellite provider.	The B-Brother model 1984 is a device that connects directly to consumers' televisions to track their television viewing habits.

Adapted from Rockley, Kostur, & Manning, 2002, p. 36

<It can be programmed to track the channels they flip to, what programs they record and what commercials they skip. The information is instantly transmitted to the cable or satellite provider.> [Brochure, Operations Guide]

We could also argue about whether even something as simple as a product description should really be written in the same way in marketing materials as opposed to technical documents. More to the point, one could argue that in this case, the content management system is essentially automating the cutting-and-pasting process, although in practice cutting and pasting is usually followed by editing and revising, and CMS content must be written to be automatically appropriate for several kinds of documents, regardless of genre or medium. As a result, after implementing this type of separation, a significant part of the writing process becomes the negotiation of content across genres rather than simply within them, and writing is structured by the design and use of information models, rule sets, style sheets, and the technical infrastructure that maintains and enforces those models and rules and presents the content whenever it is requested.

In this section, I demonstrated the different ways that presentation is separated from content by content management systems, and I attempted to demonstrate the nature, and to some extent the inherent usefulness, of creating a separation. But before I delve into what I suggest are the implications of the very specific separation enacted by content management systems, I also wish to suggest that there are at least two good reasons to think that separating presentation from content is unproblematic:

- **Content and presentation have almost always worked this way.** In an early document on SGML (the markup predecessor to HTML and XML), Tim Berners-Lee noted that some authors are reluctant to give up control of their designs:

  some authors feel that the act of communication includes the entire design of the document, and if this is done correctly the formatting is an essential part of authoring. They resist any attempts to change the representation used for their documents. (Berners-Lee, 1992)

  But despite the theoretical and rhetorical debate about the relationship between content and presentation, since the time of illuminated manuscripts most publications have been produced using a model that separates content from presentation and requires authors to relinquish control of presentation. As I write this article, I am trying to follow the formatting guidelines required by *TCQ*, and if the piece is accepted the fonts, layout, etc., will be changed to match the journal's aesthetic rather than my own. Obviously, this is not unusual; most journals, newspapers, and magazines are designed with a consistent overall look and feel that creates and reflects the rhetorical positioning of

the publication as a whole rather than of individual articles. Technical communicators tend to work in the same way, and it would be very unusual (and likely annoying to bosses) if each new manual or specification used different fonts, layouts, and trim sizes. Many communicators use templates and departmental style sheets precisely for standardizing these decisions. Why should a content management system handle design any differently? If anything, technologies that separate content from presentation may allow for better rhetorical design, because they will expedite the production of multiple effective designs of the same content for different audiences and purposes, a move that is otherwise too impractical and costly.

- **Content and presentation always get reacquainted.** Even when content and presentation are separated, they do not remain separated, and arguably were never separated to begin with. Lane Hall, an art professor who helped me expand my above discussion of the Formalists, asked, very sensibly, "What would something fully separated look like? Could it be done?" The answer, of course, is no. Text in a content management system is by definition tagged and sorted into machine-parseable structures that, when presented to humans, are displayed in different kinds of rhetorical packaging. The fact that content and presentation are remelded on the fly does not necessarily mean that the end result is not just as rhetorically effective as a painstakingly designed three-fold brochure.

There are certainly reasons to doubt the ability of content management systems and designers to deliver documents of the same quality to which users are accustomed, as has been discussed by Albers, among others:

> Document management and knowledge management systems represent the technical solution to the information problem. They provide a method of delivery, but without well-designed information presentation, the user may not realize the benefits of any improvements to the content. Many of the design decisions get made by systems analysts who, while understanding effective information retrieval, lack a strong background in understanding how to actually convert the retrieved data into information suitable for the user. (Albers, 2003, p. 338)

But if we assume for the sake of argument a well-designed system, there is little reason to worry about a separation. The consumer of a brochure, operations manual, or e-commerce site should not even notice any difference.

THE AFFORDANCES OF SEPARATION

The primary implications of content management's separation of presentation and content are for technical communicators themselves. I believe that we are only be-

ginning to see and understand the long-term implications of the separation of presentation from content, and we would do well to consider its future implications in terms of what we understand to be its affordances. I am borrowing the term *affordances* from cognitive scientist and designer Donald Norman, who in turn borrowed it from psychologist James J. Gibson. Joanna McGrenere and Wayne Ho summarize Gibson's stance, describing an affordance as "an action possibility available in the environment to an individual, independent of the individual's ability to perceive this possibility" (Soegaard, 2003). Norman (2002) used the concept to illuminate user-centered design in *The Design of Everyday Things,* although he separates himself from Gibson in that he regrets not having used the term *perceived affordances* in his book: "What the designer cares about is whether the user perceives that some action is possible (or in the case of perceived non-affordances, not possible)." The distinction, he feels, is important in clearing up the many misunderstandings of affordances he has seen in use by designers:

> because I can click anytime I want, it is wrong to argue whether a graphical object on the screen "affords clicking." It does. The real question is about the perceived affordance: Does the user perceive that clicking on that location is a meaningful, useful action to perform? (Norman, 2004b)

For an example of "perceived affordances," consider Edward Tufte's rants against PowerPoint. He argues that PowerPoint is "evil" and that it "elevates format over content, betraying an attitude of commercialism that turns everything into a sales pitch" (Tufte, 2003a); he further indicts PowerPoint for its forced sequentiality and low resolution, which he feels lead to corrupted statistical analysis and weakened verbal and spatial reasoning. In general, he suggests, "the PowerPoint style routinely disrupts, dominates, and trivializes content" (Tufte, 2003b). By way of example, Tufte argues that the extensive use of PowerPoint by NASA and Boeing contributed to the space shuttle Columbia disaster by reducing dense, critical technical information to overabbreviated bullet points; indeed, the Columbia Accident Investigation Board (CAIB) "views the endemic use of PowerPoint briefing slides instead of technical papers as an illustration of the problematic methods of technical communication at NASA" (Tufte, 2003c).

Tufte's status as a designer—Zachry and Thralls refer to him as "arguably the preeminent authority on data and statistical visualization" (Zachry & Thralls, 2004, p. 447), and Shwom and Keller's response (2003) is entitled "The Great Man Has Spoken. Now What Do I Do?"—has made the PowerPoint debate prominent. As a result, his argument led to numerous responses that argue that Tufte's argument is overwrought, ignores the widely varying techniques and skills of presenters, and in the process unfairly blames the tool for poor decisions made by users (cf. Norman, 2004a; Shwom & Keller, 2003; Simons, 2004). Norman himself argues that Tufte is correct in that "most talks are horrible and most PowerPoint

slides are bad—but that's not PowerPoint's fault. Most writing is awful, too, but I don't go railing against pencils or chalk" (2004a). Shwom and Keller echo Norman's pencil/chalk argument by suggesting the absurdity of putting a similar sort of blame on word processing software:

> Having read hundreds of poorly worded business letters in our consulting practice and teaching, as well as many dense and impossible-to-decipher engineering reports, would we be fair in saying that word processing software is just "not serious?" Had the Boeing engineers sat down to write a prose report on Debris Assessment using a word processing "tool," would the language in such a report have been less murky, the data less opaque, and the conclusion completely different? (Shwom & Keller, 2003)

Fair enough, and I tend to agree with Norman's assessment that Tufte's suggestions for improvement, which unfairly compare presentations to textbooks and full-length reports and advocate the incorporation of more and denser data, are unhelpful and misread the context of presentations. Still, I think Tufte has a point, and it is Norman's: The perceived affordances of PowerPoint encourage particular types of communication over others, and it is not unfair to hold the tool responsible and point out that unfortunate affordance when training students. PowerPoint, after all, is perfectly capable of creating the kinds of dense, data-rich displays Tufte favors (indeed, last semester I had a document design student create an extremely elaborate 3' × 4' poster using PowerPoint), but PowerPoint's basic functions encourage the display of information in short, chunky bulleted lists, and can enable the subordination of content to format by forcing users to reduce complex ideas and data into small, low-resolution chunks that can be seen from the back of a large room.

When Norman talks of affordances he means the perceived affordances of concrete, visible objects, like a coffeemaker or a website. Still, I find the concept useful for talking about technologies in general, so I will close with a discussion of the affordances of separating presentation from content in content management from several perspectives. First, it is important to remember that implementing content management, particularly a full-featured, granular presentation-separating content management system, requires enlisting allies from inside and outside of the organization. Few organizations, or technical communication departments, have the time, resources, or desire to build a content management system from scratch, and all worry about maintaining a system developed internally, by a small group of employees who may well leave. This means working with products developed elsewhere and sold to an organization after extensive research, multiple product trials, training sessions, and sales pitches both to and from management and IT. As a result, the affordances of the separation of presentation from content tend to be viewed from a wide variety of perspectives, resulting, potentially, in the following kinds of changes for technical communicators.

- **New business pressures.** Managers who have been sold on the benefits of a content management system have been persuaded, in part, by the emphasis vendors put on cost reduction, expediency, interdepartmental reuse, and the improvement of quality (cf. Andersen, this issue), and these become the clearest, most necessary affordances of the expensive technology they just purchased. While the smartest texts on content management (cf. Hackos, 2002; Rockley, Kostur, & Manning, 2002) keep their focus on users and usability, advising that some separation of content from presentation is ill advised, that occasional "one off" documents will be necessary in even the most controlled system (Rockley et al., 2002), there will be increasing pressure to fully utilize the systems, to make them comprehensive and all-encompassing. If some cost reduction and expediency are good, surely more is better.
- **New complexities in task and process management.** For full-featured content management systems, the split between content and presentation ends up being less a separation than a variable relationship that must be continually negotiated. Consider this excerpt from Rockley et al., demonstrating the difficulty of achieving granularity:

> Sentence fragments and individual words may not be appropriate for reuse. The smaller you break down your content elements for reuse, the more complex it is to reuse and manage the content. If individual words are the only component of information that changes, consider using variables, which can have a different value depending upon the instance. Variables are much easier to manage than individual word-size elements. If sentence fragments can be reused, consider creating the reusable element at the sentence level and creating a derivative element by changing the portion of the sentence that is not appropriate for reuse. (p. 41)

Technical communicators must become experts at negotiating that relationship, determining the amount of time and labor that should go into making content granular (we cannot do every word, or even every sentence) in terms of the relative payoff in performance and assumed quality and usability. And we must become expert at evaluating the process of doing so and learning to build this work into our expectations and planning.

- **Changes in what it means to write genres.** In some models of the separation, content development remains much as it was: Authors write and structure complete texts to which visual style is added as needed, allowing them to continue working flexibly within organizational genres and facilitating changes and the creation of new genres. But the implementation of a system that granularizes content and makes genre formation part of a multigenre, automatically generated presentation system means learning to write differently. In other words, as I have argued elsewhere, writing itself must be appropriately adapted for presentation via content management:

"What can be lost in attempting to create kairos-neutral chunks of content is thoughtful consideration of context, of the placement and use of information; in writing medium-, user-, and situation-neutral 'components' we risk producing components that aren't perfect fits for any contexts" (Clark, 2002, p. 22).

Still, in these systems communicators must find ways to write content that is appropriate for multiple uses and genres with only slight variations; for example, it may be easy to change a content module's tense, but difficult to change its tone. Style, Mikhail Bakhtin argued, is always situated within genre, and any individual stylistic iteration cannot be regarded as free-standing language, because genre is always situated socially. But in a separation of presentation and content that hands genre construction to presentation, writers must either write special, tagged chunks for each endeavor or, in the name of expediency and consistency, attempt to write genre-neutral content.

- **Standardizing and enforcing presentation.** Separating content from presentation and systematizing presentation affords the easy maintenance of visual consistency. In some ways, this is an obvious good because organizations can better maintain branding by taking visual style out of the hands of inconsistent human beings. Consistency is a significant part of the pitch for content management, which often defines quality in terms of consistency:

A unified content strategy helps to improve the quality of content. Content is clearly modeled for consistent structure; increasing its readability and usability. Most importantly, content is accurate and consistent wherever it appears. Issues of inaccurate content, inconsistent content, or missing content are reduced or eliminated. (Rockley Group, 2002)

But if presentation includes genre structure as well as visual style, the use of a content management system means that named genres are likely to become reified and enforced. Due to the time and resources required to construct the characteristics of a genre in a content management system, it is unlikely that less common, less developed, unnamed genres will be built into the system. The most likely candidates for inclusion are technical manuals, product sheets, quick reference cards, etc., end-user oriented genres rather than internal documentation and correspondence.

Genre, though, should not be mistaken for format. As defined by Miller (1984), Swales (1990), Berkenkotter and Huckin (1995), and countless others, genre is a typified textual response to a typified social situation**,** but we should not mistake this for templating. One instance of a genre need not be related visually or even structurally to another; in Winsor's discussion of work orders, she notes that the elements of work orders varied in ordering

and that those formal features "did not seem to … be what marked these texts as genre." Everyone knew what a work order was from the action they repeatedly performed, even when handwritten (Winsor, 2000, p. 169).

The separation performed by content management systems risks defining genres by their features rather than by allowing them to be specifically designed for effective use, forcing technical communication into a templated approach to generic problem solving rather than allowing for the potential complexities of genre. In general, an information model designed to produce a technical manual or a reference sheet will rely on standardized features (introduction, specifications, steps) that may be presented or removed from a given presentation. But there is little room for genre modification that might make the overall document more effective.

- **New user expectations.** Many of us are still writing and teaching the creation of fairly static technical content that is presented in fairly traditional ways, but the separation of presentation and content facilitates user control over information. Increasingly, Web users, in particular, expect alternative access to content, for example the ability to take content presented as Really Simple Syndication (RSS) feeds and repurpose it, scraping relevant content from irrelevant, presenting content chunks on their own pages, or creating new feeds that mash together content and presentation from other information streams. We are only starting to see the types of changes these practices will bring about in terms of authorship and authority; everyday work, and content and presentation may be redefined.

NEW POSSIBILITIES

There are lots of points to be made about the difficulty, expense, and failure rate of content management implementations and about the overly effusive and technophilic pitches of various vendors. Even small website content management systems can be disasters:

"Today, more than 60 percent of companies that have deployed Web content management solutions still find themselves manually updating their sites," wrote analyst Matthew Berk. Overcomplicated, end-to-end packages can as much as quintuple site operational costs over human labor alternatives. Unfortunately, the breadth of many vendors' all too-inclusive 'silver-bullet solution' vision has left these companies struggling with platform lock-in, overengineered site infrastructures, exorbitant technical maintenance costs, and per-business-user costs averaging as much as $25,000 per year. (Joyce, 2003)

In addition, while in the last few years some of us have enthused over the new leadership possibilities of content and knowledge management (Wick, 2000), others of us have worried more darkly about the potential negative effects of bad implementations. Albers elsewhere suggests that a problem of content management can be that "multiple writers at multiple locations contribute information to a document database which then, on reader request, dynamically generates a unique document fulfilling current reader needs. What the reader sees is not a document that an editor has carefully groomed, but rather a dynamic document that was compiled from a database just before the information was presented" (Albers, 2003). The clear implication here is that a poorly designed system, in particular, could negatively affect our readers and users.

Others have worried over the effects of even a good implementation. For some technical communicators, separating content from presentation has meant a change from authoring entire documents to writing and tagging content modules and handing over presentation to a machine that automatically implements an organizational style guide (Clark, 2002). Whether a writer sees this as "freeing" them to focus on content (Hackos, 2002) or regrets the loss of autonomy and the diverting work of presentation, implementing a content management system means making a new and specific technological argument for definitions of content and presentation and hard coding that argument into the structure of documents and daily work. To this, the best and perhaps only approach is advocated by Albers and others (Carter, 2003; Sapienza, 2002, 2004; Wick, 2000)—for technical communicators to get more educated and involved in the design, usability, and implementation of content management and its associated technologies. Some of what I have suggested about the separation seems glum, but I also see enormous possibilities and intriguing challenges associated with this way of thinking about technical communication.

In practical terms, technical communicators must begin by recognizing the complex nature of any separation of content from presentation, and I hope this article has offered a beginning to that discussion. I hope I have been successful in arguing that "separating presentation from content" is, to a fair extent, a catch phrase that may work as convenient shorthand but is ultimately inadequate for dealing with the specifics of content management. There is a significant history to separating presentation and content, and that history teaches us that the separation line can be drawn in different places. In content management, the line is drawn differently in different implementations, and where the line is drawn can have real implications for technical communicators.

So what does it all mean? Academics and practitioners should avoid the temptation to talk in simple terms about separations. We should instead talk specifically about the technologies we are using and how they enable and limit our ability to help our users achieve their goals. Was the system designed and implemented by IT specialists, limiting our ability to build communications that meet our standards? Then we must find ways to advocate for a critical understanding of the sepa-

ration, one that allows us the critical flexibility to structure and alter our genres. Does the system we are using promise to deliver the same content in multiple presentations? Then our work and research must focus on how that separation is realized and take on the interesting new work of guaranteeing our users an effective experience in multiple formats and media. Does the system granularize our content and promise to uncritically deliver the same content to marketing, human resource, and technical genres, with no regard for tone and style? Then we must learn to make arguments for the importance of avoiding standardizing for its own sake, of the importance of writing custom content for custom situations.

I hesitate to make an inevitability argument, but the direction in which content production is moving at this moment is pretty clear; also clear is that as researchers, teachers, and practitioners of technical communication we must learn to address these changes. For some academics and practitioners, this may mean leaving behind much of what we currently think of as technical communication in favor of new kinds of design and communication roles. Meanwhile, many technical communicators have simply gone about their work, learning to think about and work with these systems and becoming acquainted with a dizzying array of new tools, approaches, and vendors. Over time, (after passing through stages of denial and anger), I have come to accept content management and the changes it brings to technical communication and to believe that the separation of presentation from content may introduce interesting new types of rhetorical work to technical communication pedagogy and practice.

REFERENCES

Albers, M. (2003). Single sourcing and the technical communication career path. *Technical Communication, 50,* 335–343.

Berkenkotter, C., & Huckin, T. (1995). *Genre knowledge in disciplinary communication: Cognition/culture/power.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Berners-Lee, T. (1992). SGML. Retrieved April 21, 2007, from http://www.w3.org/History/ 19921103-hypertext/hypertext/WWW/MarkUp/SGML.html

Billig, M. (1987). *Arguing and thinking: A rhetorical approach to social psychology.* Cambridge, MA: Cambridge University Press.

Bowker, G., & Star, S. L. (2000). *Sorting things out: Classification and its consequences.* Cambridge, MA: MIT Press.

Broberg, M. (2004). A successful documentation management system using XML. *Technical Communication,* 51, 537–546.

Buchanan, R. (1989). Declaration by design: Rhetoric, argument, and demonstration in design practice. In V. Margolin (Ed.), *Design discourse: History, theory, criticism* (pp. 91–109). Chicago: University of Chicago Press.

Burton, G. (2007). Silva rhetoricae: The forest of rhetoric. Retrieved April 19, 2007, from http://humanities.byu.edu/rhetoric/

Carter, L. (2003). The implications of single sourcing for writers and writing. *Technical Communication, 50,* 317–320.

Clark, D. (2002). Rhetoric of present single-sourcing methodologies. *Proceedings of the 20th Annual International Conference on Computer Documentation*. (pp. 20–24). New York: ACM Press.

Cohen, M. (2004). Separation: The web designer's dilemma. Retrieved April 21, 2007, from http://alistapart.com/articles/separationdilemma/

Greenberg, C. (1982). Modernist painting [Electronic version]. In F. Frascina & C. Harrison (Eds.), *Modern art and modernism: A critical anthology*. New York: Harper and Row.

Hackos, J. (2002). *Content management for dynamic web delivery*. New York: Wiley.

Intentional Design, Inc. (2006). Types of content management. Retrieved April 21, 2007, from http://www.intentionaldesign.ca/index.php/weblog/blogcentre/content_management_types/

Joyce, E. (2003). Study: Content management tools fail. Retrieved April 19, 2007, from http://www.atnewyork.com/news/article.php/1690881

Kinross, R. (1989). The rhetoric of neutrality. In V. Margolin (Ed.), *Design discourse: History, theory, criticism* (pp. 131–143). Chicago: University of Chicago Press.

Kostelnick, C., & Roberts, D. (1997). *Designing visual language: Strategies for professional communicators*. New York: Longman.

McLuhan, M. (1994). *Understanding media: The extensions of man*. Cambridge, MA: MIT Press.

Meyer, E. (2003). The incomplete divorce. Retrieved April 21, 2007, from http://meyerweb.com/eric/thoughts/200310.html#t200310015

Miller, C. R. (1984). Genre as social action. *Quarterly Journal of Speech*, *70*, 151–167.

Norman, D. (2002). *The design of everyday things*. New York: Basic Books.

Norman, D. (2004a). In defense of PowerPoint. Retrieved April 19, 2007, from http://www.jnd.org/dn.mss/in_defense_of_powerp.html

Norman, D. (2004b). Affordances and design. Retrieved April 19, 2007, from http://www.jnd.org/dn.mss/affordances_and.html

Parapadakis, G. (2000). What's in a name? Retrieved April 19, 2007, from http://www.edocmagazine.com/expert_article.asp?ID=18350&header=e_expertcorner_header.gif/

Porter, J. (1998). *Rhetorical ethics and internetworked writing*. Greenwich, CT: Ablex.

Postman, N. (1985). *Amusing ourselves to death: Public discourse in the age of show business.* New York: Viking.

Postman, N. (1992). *Technopoly: The surrender of culture to technology*. New York: Vintage.

Rockley, A., Kostur, P., & Manning, S. (2002). *Managing enterprise content: A unified content strategy*. Indianapolis, IN: New Riders.

Rockley Group. (2002). What is a unified content strategy? Retrieved April 21, 2007, from http://www.managingenterprisecontent.com/

Sapienza, F. (2002). Does being technical matter? XML, single source, and technical communication. *Journal of Technical Writing and Communication*, *32*, 155–170.

Sapienza, F. (2004). Usability, structured content, and single sourcing with XML. *Technical Communication, 51*, 399–408.

Schriver, K. (1996). *Dynamics in document design: Creating text for readers*. New York: Wiley.

Shwom, B., & Keller, K. (2003). The great man has spoken: Now what do I do? A response to Edward R. Tufte's "The cognitive style of PowerPoint." Accessed April 19, 2007, from http://www.communipartners.com/documents/ComInsV1._000.pdf

Simons, T. (2004). Edward Tufte doesn't hate PowerPoint, he hates the culture that spawned it. Retrieved April 19, 2007, from http://www.allbusiness.com/services/business-services-advertising/4248692-1.html

Soegaard, M. (2003). Affordances. Retrieved April 19, 2007, from http://www.interaction-design.org/encyclopedia/affordances.html

Stein, B. (2000). Separation anxiety: The myth of the separation of style from content. Retrieved April 21, 2007, from http://www.alistapart.com/stories/separation/

Swales, J. (1990). *Genre analysis: English in academic and research settings.* Cambridge, MA: Cambridge University Press.

Tufte, E. (2003a). *The cognitive style of PowerPoint*. Cheshire, CT: Graphics Press.

Tufte, E. (2003b). PowerPoint is evil [Electronic version]. *Wired magazine.* 11.09.

Tufte, E. (2003c). PowerPoint does rocket science. Retrieved April 19, 2007, from http://www.edwardtufte.com/bboard/q-and-a-fetch-msg?msg_id=0001yB&topic_id=1

Wick, C. (2000). Knowledge management and leadership opportunities for technical communicators. *Technical Communication, 47*, 15–29.

Winsor, D. (2000). Ordering work: Blue-collar literacy and the political nature of genre. *Written Communication, 17,* 155–184.

World Wide Web Consortium. (2007a). Device independence principles. Retrieved April 21, 2007 from http://www.w3.org/TR/di-princ/

World Wide Web Consortium. (2007b). Goals. Retrieved April 21, 2007, from http://www.w3.org/Consortium/mission

Zachry, M., & Thralls, C. (2004). An interview with Edward R. Tufte. *Technical Communication Quarterly, 13*, 447–462.

Dave Clark is an assistant professor of English at the University of Wisconsin-Milwaukee. His teaching and research interests include community engagement, the rhetoric of technology and management, knowledge management, and information architecture. His e-mail address is dclark@uwm.edu.