Rhetorically Structured Content:  
Developing a Collaborative Single-Sourcing Curriculum

Charlotte Robidoux

Hewlett-Packard Company

Charlotte Robidoux is an Information Planning Manager at Hewlett-Packard Company. She leads collaborative teams implementing the single-sourcing strategy for the StorageWorks Division. She earned her PhD in Rhetoric from The Catholic University of America.

Structured writing is a method for developing consistent categories of information that can be “single sourced” or reused for various contexts. Creating distinct structures—such as concepts, procedures, and examples—prepares content for the application of XML markup elements that describe each category. A content management system identifies these structural elements, which facilitates reuse and repurposing. Students seeking positions in organizations that single source information must become proficient in structured writing and in writing collaboratively.

Consider how sophisticated the tools of writing have become over the last four millennia—from papyrus and the inkwell to well-formed XML (Bolter, 2001, p. 21). In recent years, corporations have begun managing “single sources” of structured content using XML and database technology, a practice that has raised concerns about a mechanized writing process. The emphasis on single sourcing has led to differences of opinion over the merits of single-sourced writing.

Albers (2000, 2003) questioned the premise that single sources of generic, automatically assembled pieces of content—the antithesis of the “craftsman” model of writing—can convey a coherent message to readers (2003, p. 336): “Each piece, however well written, contributes to discontinuities across the whole” (pp. 337-338). Weiss (2002, 1993) objected to so-called “egoless” writing, which drains writers of their “artistic impulse,” and to “putting thousands of chunks of information into a database stew” (2001, p. 3); he described this process as a “rhetorically neutral” enterprise (1993, p. 61). Similarly, Clark (2002) questioned the value of combining nonspecific content: ‘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…neutral ‘components’ we risk producing components that aren’t perfect fits for any contexts” (p. 22). In contrast, Haramundanis (2002) countered Weiss’s notion that automated authoring environments divest the “ego” from technical writing and by extension the “art” and “craft” of preparing manuals (pp. 12, 14). Likewise, Sapienza (2004, 2007) resisted the idea that a writer’s craft is in danger of being rhetorically insignificant: “one cannot yet claim that the "craftsperson" role of a technical communicator is endangered by such technologies…If anything, the process of modular writing evokes a deliberate and highly skilled process that is at the heart of any craft” (407).

Underlying these differing views is the common theme that organizations need proficient teams of technical writers, who can craft cohesive customer content. Ultimately, the range of arguments expressed here helps to broaden our understanding of technical communication. The dialog among experts reveals a genuine interest in the contributions of writers and in the effectiveness of the information they develop.

In this article, I argue that structured writing, in accordance with Horn’s method (1993), is the basis for creating effective content in environments that employ on single sourcing technology. What makes structured content effective is strategic collaboration within writing teams (see also Robidoux and Waychoff, 2005a, p. 88; 2005b). To that end, students of technical communication explicitly should be taught how to write and use structured writing in collaborative settings.

Horn (1993) conceptualized the structured writing method, which he developed in the late 1960s, as “the basic units of the subject matter…the information types that clustered a unique prescription for chunking information…into seven information categories” (p. 11): procedure, process, concept, structure, classification, principle, and fact. The most recognizable of his categories include procedure, process, and concept. The relevance of grouping information is apparent in the DITA (Darwin Information Typing Architecture) standard, which employs the “concept,” “task,” and “reference” as key categories. While the structured writing method was first known as information mapping, Horn recommended the generic term, structured writing, in the early 1980s (p. 16). The value of structured writing and single sourcing was recognized by Hall (2001), who used this method in a defense organization to “solve real content development and management problems” and produce coherent, usable documents (pp. 235, 244). Other organizations pursuing single-sourcing strategies are gathering initial feedback that reveals improvements across product documentation and the ability to meet customer requirements (personal communications with Lawrence, 2007; Wear; 2007; Ament, 2007; and Routh, 2007).

Without question, the concept of structure is broad, and the word can have many meanings, including that related to the organization, syntax, and style of a document. Structure also can be understood locally, as the “inner structure” within a book, and globally, as the “outer structure” across many kinds of publications (Boiko, 2001, p. 25). This complexity and the fact that the writing process is typically individualistic make it difficult to apply structure uniformly (Perlin, 2007, pp. 17-18). If left to the task of structuring text even with instructions and a template, writers often make different decisions about the use of headings, paragraph breaks, lists, tables, and the overall hierarchy. Some writers apply too much structure, and some do not apply enough. Inconsistently structured content can be confusing to customers. The process of achieving consistent structure thus requires clear standards, tools that reinforce that structure, and

1 To clarify terminology, I treat the terms “content” and “information” interchangeably as components that make up customer documentation.
a strong team of writers and editors who collaborate strategically. Team members must deliberate—engage in rhetorical activity—about documentation standards to develop guidelines that will benefit users. While Weiss (2002) maintained that "collaborative and structured methods" drain writers of the motivation and the satisfaction of crafting documents on their own (p. 3), the approach to collaboration in the documentation group to which I belong at Hewlett-Packard Company (HP) has had the opposite effect.

Structured writing proficiency and experience working as part of collaborative teams are essential for students of technical communication who seek positions in organizations that are committed to single-sourcing strategies. As described below, structured writing classes can help students write collaboratively for reuse. However Lancaster (2005) indicated that industry and classroom practices are not aligned: “single-sourcing and content reuse do not fit into the academic writing model, and in many cases contradict traditional rhetorical principles taught in technical writing classes; yet there is an expectation (at least implied) that students will gain this knowledge from formal education” (online dialogue). The need for a more extensive single-sourcing pedagogy is important as suggested by Hart-Davidson (2001):

By what means can technical communicators in the workplace and in academia work to shape the emerging technologies that not only affect the work we do but are growing up in our backyards?...technologies that seek the rhetorical expertise of technical communicators and leverage the ancient technology of writing….We need theory. By this I mean the ranks of working professionals and academics in technical communication should participate in activity that makes the core expertise of technical communication explicit. (pp.146-147)

Similarly, Elbe (2003) calls for a single-sourcing curriculum: “With classroom practice in information modeling and structured writing processes, our students will be better prepared to become productive members of the writing teams they join” (p. 349).

In the three sections that follow, I (1) summarize the enduring need for structure; (2) review the current methods of structured writing—those of Ament (2002), Rockley (2002), and Hackos (2002); and (3) provide recommendations for developing a collaboration-based structured writing curriculum informed by strategies implemented at HP.

THE ENDURING NEED FOR STRUCTURE

To appreciate why structure is central to modern tools of writing, it is useful to consider the important role of structure throughout the ages. Structure was an essential aspect of classical rhetorical theory, and, as such, contemporary technical communication can be connected to earlier writing theory: “The classification of discourse into different types has been one of the continuing interests of rhetoricians since the classical period” (Covino and Jolliffe, 1995, p. 282). In the Art of Rhetoric and Topics, Aristotle alluded to numerous topics, or lines of reasoning, that he identified as a means to structuring ideas—such as definition, division, relationship, and example. The topics organize content just like categories of information, which are their modern counterparts. Similarly, the progymnasmata (a Greek term for exercises that “come before”) represent basic exercises to help students develop self-contained themes that can be assembled as the sections of a classical oration (Crowley, 1994, p. 282). Made popular by the Greek rhetoricians Aphthonius and Hermogenes, these exercises, which required students to practice description, comparison, and thesis development, were so important that Rainolde (1563, reprinted 1945), adopted them as teaching tools in the sixteenth century. In the nineteenth
century, Bain (1877, reprinted 2001) also endorsed the use of modes of discourse, which included narration, description, exposition, and argument. Bain’s work eventually became aligned with thesis-driven compositions and, by extension, with paragraphs built around strong topic sentences—the basis of much twentieth century writing instruction (Connors, 1981, p. 449; D’Angelo, 1986, p. 431). Over time, the process of categorizing information into predictable structures has been an essential aspect of writing; integrating them into coherent texts with rhetorical significance makes technical communication an artistic enterprise—an outcome that is also viable with modern writing tools.

The widespread use of word processing and desktop publishing tools in the 1980s made it easier for writers working independently to create cohesive texts by cutting and pasting information to improve coherence, add illustrations, and assign style attributes. Modifying document formats such as bold headings, italicized titles, and paragraph alignment enabled writers to manage the rhetorical dimensions of their texts on many levels. As online help systems and Web applications became vehicles for disseminating technical information in the mid-1980s and early 1990s, writers began applying Hypertext Markup Language (HTML). In many cases, when assigned to document the same product, online help writers worked independently of those assigned to printed manuals. The use of HTML markup would specify the format and, to a lesser extent, the structure of the content—such as heading levels (H1, H2), paragraphs (P), or table (table border=0...width="100%"). However, because HTML is display oriented, unregulated, and primarily linear in terms of handling structure, it is limited in its ability to reuse content—to convey the same information presented in online materials as in printed materials (Rockley, 2002, pp. 281-282).

For this reason, standards were developed to manage structure. Initially, Standardized General Markup Language (SGML, established in 1986) made it possible to separate form from content. This descriptive markup language made it possible to categorize the structures within a document and established the importance of collaboration to make effective reuse possible; however, SGML was overly complex and difficult to use. Thus, XML was issued in 1998 as “a specification for defining markup languages,” a more straightforward variation of SGML with an adaptable array of tags (Rockley, 2002, p. 2). The improved technology was featured as part of a rich tool set that emphasized structure, as Table 1 below shows:

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Extensible Markup Language (XML)</td>
<td>Markup language designed to create structured content</td>
</tr>
<tr>
<td>Extensible Stylesheet Language (XSL)</td>
<td>Style sheet language used to render XML documents</td>
</tr>
<tr>
<td>Content Management System (CMS)</td>
<td>Software that runs on a database to store, manage, and control versions of digital information</td>
</tr>
<tr>
<td>Document Type Definition (DTD)</td>
<td>File containing the rules that govern the structure of the elements in an XML document</td>
</tr>
</tbody>
</table>

With the advent of XML came a more streamlined approach that would facilitate content reuse and increase the need for collaboration. Effective reuse required consistent structures composed of XML “element” tags, uniformly applied, which describe the function of the content, such as a procedure (<procedure> <step><para></para></step> </procedure>). This content would be available for writers to reuse across documents because users can search for markup elements in
a CMS. In addition, the CMS supports metadata—descriptive information about the subject of stored content—which enables authors to find information more easily and reuse it. In many ways, the CMS enables a more sophisticated method of cutting and pasting content to generate coherent and consistent meaning across documents. Because XSL stylesheets handle formatting, writers can spend less time manipulating styles and more time on crafting meaningful content. Skilled documentation teams that effectively apply structure collaboratively can automate the writing process so that they can create more rhetorically significant, customer-focused content.

Despite the benefits of single sourcing, challenges are inevitable when implementing the technology, and such challenges are directly related to the need to work in collaborative teams. For example, apart from the high price of a CMS and the difficulty of selecting the right system, there are other impediments like the extensive, often costly, training that is needed to develop expertise with markup language (Rockley, 2002, p. 408). And to ensure that the investment in training yields results, documentation teams must collaborate to establish and maintain standards that will guide reuse and consistency across documents. While writers are still responsible for building their own documents, they must always be mindful of the shared content that they are using and that others might use. For example, if they want to modify shared content, they must draw on editorial support to investigate the proposed change to determine its impact on other writers.

Documentation team members learn not only to work together, but they also learn to take on new roles or to adapt existing ones: administering a CMS, authoring content in XML, developing XSL stylesheets, and managing one or more DTDs as well as complex publishing systems. Moreover, there is substantial effort involved in managing these processes and role changes. Establishing a new method of developing technical documents is one of the most difficult adjustments for documentation teams. The content development process is interdependent, and it pushes teams to hone their rhetorical skills—to deliberate cooperatively on standards related to creating structure. Because rhetorical savvy also is essential for learning how to combine unique content and that intended for reuse, it is important for technical communication students to hone their persuasive skill by working within collaborative teams and in settings that mirror the contemporary workplace.

CURRENT METHODS OF STRUCTURED WRITING

In this section, I describe the structured writing methods used in many organizations, those of Ament (2002), Rockley (2002), and Hackos (2002). I focus my study on the classroom implications of the structured writing methods. This emphasis contrasts with the survey undertaken by Williams (2003); he conducted a review of single-sourcing approaches to understand their overall impact on technical communicators (p. 321). At times, it is difficult to draw parallels between Ament, Rockley, and Hackos because they do not always define the same terms in exactly the same way. Even so, all three experts expressed the importance of presenting coherent and reusable information, a basic requirement for authoring in a CMS environment.

Building Modules

In Single Sourcing: Building Modular Documentation, Ament (2002) provided a practical, step-by-step approach to developing content from the bottom up—within and across a set of documents. His goal was not to set forth a comprehensive, top-down strategy and examination of the CMS. Rather, he sought to “develop local, project-based standards,” and he employed the term “module” to refer to an information category (p. 1). In total, Ament provided
detailed guidelines for structuring 19 types of modules—including procedures, processes, topics (concepts), definition lists, and examples. While Ament’s aim was to advance single sourcing technology, his modular approach in fact downplayed technical dimensions: “Single sourcing is a methodology, not a technology. Although the software tools associated with single sourcing are complex, it is modular writing, not technology, that ultimately determines the success of your single sourcing projects” (p. 1). He also argues that applying information categories consistently across a set of documents—an essential step for reusable, modular content—ultimately strengthens documentation: “Although its primary goal is to save time and money, single sourcing improves the quality of your documentation…Although usability is a goal for most technical writers, single sourcing makes it the top priority” (p.3). Drawing a contrast between format-based, linear writing and content-based, modular writing, Ament emphasized the need for stand-alone modules to ensure effective reuse (pp. 4-5). Creating reusable module requires an interconnected team:

Documentation is a collaborative effort. Although individual documents are theoretically owned by writers, these documents are actually the result of a team effort….By making team members interdependent, single sourcing acts as a catalyst for team synergy….For single sourcing to work at all, the team needs to make hard decisions about many issues, big and small” (p. 10)….

Ament described three principles of modular writing: chunking, labeling, and linking. The type of information determines how the content should be chunked—into primary modules (such as topics, procedures, processes, and glossary items) or secondary modules (such as examples, tables, and figures) (pp. 6, 26). Chunking represents the process writers go through to create distinct modules. While primary modules can be stand-alone units, secondary modules must be integrated into primary modules (p. 30). For instance, a topic may include an example or graphic to illustrate a topic. Next, Ament referred to the importance of labels to characterize the subject of modules consistently, such as “Installing the XYZ Software” for a procedure. Ament’s third principle, linking, refers to the cross-references that relate modules to one another. That is, if “Installing XYZ Software” relates to “Configuring XYZ Software,” the writer would link the two modules.

While Ament’s approach to creating what he called modular documents entails ten steps, the process of developing the modules that make up documents is accomplished in the first three steps, building, assembling, and linking the modules to one another. The first step, building modules, involves asking basic questions—who, what, when, where, why, and how—that users want the modules to answer. This technique recalls the structuring principle used by journalists to ensure complete discussion of a subject (Kinneavy, 1980, p. 161). Ament explained that types of modules are poised to answer certain questions. For example, while a topic or concept often answers the question what, a procedure is focused primarily on providing answers about how to perform a certain task. To begin assembling modules, Ament’s second step, team members must evaluate the modules in relation to one another to determine how best to combine them for various audiences, purposes, and outputs (Ament, 2002, p. 7). For example, error messages might be designated for a troubleshooting guide, and categories of commands for a command line reference guide, both print-ready and electronic texts. Combinations of topics, procedures, and processes might be designated for online help systems integrated into software. The emphasis on what readers need in what context and in what order makes structured writing effective.

Once the content is assembled, authors can proceed with the third step, linking, to ensure
that modules are properly interrelated through structures that function as cognitive bridges: the table of contents, section contents, inline cross-references, and index items (p. 14). It seems important to reiterate that Ament presented a bottom-up structured writing approach, which emphasizes modularizing content at the local, project level rather than across projects from the top. Nevertheless, the process of building from the ground up is an important prerequisite for any publication group seeking a CMS solution across the enterprise.

**Modeling Information**

In *Managing Enterprise Content: A Unified Content Strategy*, a comprehensive guide to CMS implementation, Rockley (2002) described the importance of modeling information to “identify and document the framework” of an organization’s reuse strategy (p. 159). For Rockley, modeling entails determining what categories of information are essential for all documents associated with a product or set of products. It also involves determining how each document can achieve a certain purpose and ensure maximum reuse. This careful planning of content for various purposes reveals that the single sourcing process is inherently focused on users. Information modeling begins once a top-down audit and analysis of an organization’s information resources has been conducted. The process involves developing a unified structure across the enterprise to ensure consistency where it is essential. An information model should be prepared for all information types and products and wherever information is required. Components of a complete model include semantic information for specific XML elements, base information for generic XML elements, the granularity of the information (size of the parts), metadata, and a map indicating where reuse will occur and how. Semantic and base element requirements contribute important details for creating structured writing guidance: “The guidelines for creating structured content are contained in the models” (p. 388).

When she discussed writing practices, Rockley first examined the basic process of establishing structure by examining the underlying types of content and their logical relationship to one another (p. 382). Rockley cited Horn’s (1993) information mapping methodology as a source for her discussion of structured writing (Rockley p. 382). Arguing that a structured approach to writing improves the quality of the content, Rockley drew on cognitive psychology and the human need for patterns and consistency: “structured writing is how people read, process, and understand information…[A]uthors follow standards developed for certain types of content, ensuring content is always presented consistently” (p. 383). To ensure effective information modeling, she underscored the need for strategic collaboration:

From an organizational perspective, collaborative authoring requires that all authors have an understanding of how content is used in multiple situations and the ability to work together to create it…. [T]he information models…are a collaborative effort, and they need collaborative revisiting at the start of new projects to ensure that they still accommodate the information needs of the new project (368-369).

Rockley described five principles of structured writing, which reflect Horn’s influence: chunking, labeling, relevance, reusability, and consistency (pp. 384-386). A chunk of content should contain five to nine pieces of related information. For example, writers and editors make decisions about the need to split long procedures into a number of smaller chunks if it will make it easier for readers to comprehend aspects of complex tasks. By assigning labels to chunks, readers can immediately grasp the topic and decide whether they want to continue reading or choose another topic. This emphasis on audience reinforces the rhetorical facets of structured writing. Chunks should only contain information pertaining to the label, which encapsulates the
subject under consideration. Labels for procedures and processes often contain both a verbal component and a noun phrase—for example, “Troubleshooting the XYZ Feature.” Labels in general should be substantive so that readers can easily locate information. The uniform treatment of labels and chunks in terms of phrasing and organization facilitates reuse and ensures readability.

Rockley’s principles of structured writing function as a kind of recipe—bottom-up guidance for authors, architects, and reviewers on what kind of information is required and optional for a particular type of document—for example, a user manual, brochure, or training material (pp. 388-389). While procedures would be required in the user manual and training material, they would not be necessary in the brochure. Rockley termed the process of determining what content to include in a given output “the building-block approach” to structuring content (p. 390). To ensure that authors employ structured writing principles, Rockley recommended providing explicit writing guidelines that include best practices for applying “generic tags” or “base elements” (pp. 170, 389). She also provided useful samples of XML element placement—for example, a “semantically tagged procedure” (p. 255). Insofar as Rockley considers both a comprehensive content management strategy and explicit bottom-up guidelines, she endorses a bi-directional approach to structure.

**Developing Tiers of Information**

In *Content Management for Dynamic Web Delivery*, Hackos (2002) developed a theory of structure that consists of three components: an Information Model, information types, and content units. She illustrated these components as concentric circles, with the Information Model as the overarching, top-down strategy for managing an organization’s content and the basis of its metadata (p. 124). Although the three-tiered methodology targets information developing websites, her basic principles are also applicable to other kinds of publishing environments. The components that make up the dimensions of the Information Model include metadata, the product model, customer profile and requirements, information delivery, authoring requirements, and legacy information strategies (pp. 136-157). Ultimately, the Information Model establishes a practical taxonomy for categorizing and labeling information.

Information types exist within the Information Model: “The information types will provide your authors with the basis for creating well-structured modules that represent a particular purpose in communicating information” (p. 126). Like Rockley, Hackos also promoted a methodology that is bi-directional, placing content units, the smallest component, within information types: “The content units will describe the chunks of content that are used to construct each information type” (p. 126). Because content units are the building blocks used to create information types, they are complementary entities. Defining an information type as “stand-alone chunks of information,” or topics, Hackos considered the importance of topics as “the key to sound information design for content management” (p. 162). She recommended explicit guidelines for creating information types: “By creating topics according to rules established for the information type, authors can ensure that their information is consistent, well-structured, and reusable in a variety of contexts” (p. 162). Hackos identified the standard information types used in various industries: “concept, procedure, process, glossary entry, fact, example, figure, assessment, and exercise” (pp. 190, 192).

Hackos also believed that authors should have guidelines that define each type and specify when it is used and for what purpose. For example, a concept referring to a product should include a product description, overview, and benefits; it might also indicate when it should be used and how it should be written (p. 191). In the Hackos system, information types
establish the strategy for organizing content, whereas content units define the exact items to include for a particular audience: “To decide upon content units to include in an information type, understand the background, education, and prior experience” of the users (204). Evidently, Hackos made a distinction between information types and the content units that structure each type so that authors can modify information to meet the needs of various audiences. Like Rockley (2002), Hackos (2002) provided examples of XML element placement so that the markup code reinforces the written structures (pp. 206-207). She also stressed the importance of collaboration: “In information development, it is traditional to assign one author to one book….Using a collaborative writing team…authors become the subject matter and customer experts for a particular type of information” (p. 332). Hackos also referred to examples in which collaboration enabled authors “to make better decisions on the information that is common to all and the differences that must be accommodated” (p. 322). Her examples reinforce the idea that collaboration and single-sourcing can tailor content to customer needs, resulting in documentation with improved customer focus.

These three structured writing strategies have had an influence on various enterprises seeking to manage single sources of content. These strategies represent basic principles of content development, structure, and organization—principles described long ago by Aristotle, Apthonius, and Hermogenes. While it is simple enough to appreciate the idea that structure is a foundation for creating and managing information, implementing it is not so easy. Ament (2002) asserted that structuring content from the bottom up is a labor-intensive process, one often conducted on real projects under tight deadlines. Moreover, writing teams work with vast amounts of legacy material that requires extensive work to restructure. Evaluating structure from the top-down also requires considerable effort to ensure that content can be reused across many projects, products, and media. Only when teams apply structure within and across documents is it possible to reuse and repurpose content effectively in a CMS. That is, while restructuring content from the bottom up can improve the readability of information, it may not be reusable without top-down analyses.

**IDEAS FOR A STRUCTURED WRITING CURRICULUM**

When formulating a structured writing curriculum, it seems important to consider broad theoretical and practical dimensions that can function as an overarching focus for learning. Sapienza (2004) described such an interdisciplinary approach to single-sourcing content: Structured content development does not proceed independently from usability. The usability of documents requires the integration of multiple knowledge areas, including rhetoric, information architecture, usability, and computing. Each knowledge domain implicates the other at most every phase in the project (407).

Indeed, many domains of knowledge are equally important in content development. At the risk of being overly ambitious, I make recommendations below for a curriculum that is mindful of various specialties, knowing full well that such an enterprise extends beyond the scope of one course. Even so, my aim is to conceive of a course that transcends structure for the sake of structure, one that explores its value relative to a specific context and audience.

First, I summarize practices undertaken by our documentation team at HP. Then, I sketch out a structured writing course—fittingly made up of modules—that might replicate a content structuring process in industry (see Table 2). The modules could be expanded or contracted depending on the background of the students. The modules could be split over two semesters for
beginners, or combined into one seminar for advanced writers. Likewise, suggested readings, discussion topics, and assignments could be substituted as appropriate to accommodate particular points of emphasis.

Over the course of four years, our HP writing team has successfully structured texts for the purpose of creating reusable content modules about HP storage products. Before we even had a CMS in which to store modules, we recognized the need to apply structure and consistency to our texts to make them more accessible to customers, who needed to find information quickly. Thus began the process of establishing small teams of writers and editors who began analyzing our documentation and deliberating over the standards for our 30 different types of documents and 17 different information categories.

The team identified what content each category should contain and how it should be organized and expressed. We used these standards as a basis for structuring documents, a process that entailed close collaboration between writers and editors, who proposed recommendations and negotiated with each other and then with subject matter experts. What resulted from restructuring was an incredible awareness that we could remove many words and express ideas more succinctly and that placing large passages of text in definite categories made the information easier to find and comprehend. We also observed how common it is for writers to adopt their own favorite practices when working on books independently. Working in “silos,” as Rockley (2002) indicated, gave rise to inconsistency, which was difficult for editors to overcome across many thousands of pages of product documentation. Yet, when we systematically implemented structured writing guidelines from the ground up within a document set, as suggested by Ament (2002), we began to see clarity and improved consistency. This success prompted us to apply the technique to other projects incrementally. Eventually program teams commented on the improvements and saw our documents as providing more value to customers.

When our organization purchased XML and CMS software, many documents were ready for conversion to XML because their structure could readily be “mapped” to XML markup elements. The transition to XML, however, required the team to revisit our structured writing guidelines to ensure that XML markup was being applied uniformly—so that the tags themselves reinforced the categories of information. This effort resulted in revised guidelines that we call “information structures” for each information category (see Figure 1). Our comprehensive application of structured then enabled team members to begin “content mapping” in spreadsheets, differentiating among unique, similar, and identical content, a preliminary step to reusing content. Once identified, content flagged as similar and identical were configured as “modules” in our database for all team members to draw upon.

To convey how the structured writing process readies content for reuse, I provide an illustration of content before and after restructuring (See Figures 2 and 3). In the unstructured illustration (Figure 2), note that the large passage of text combines a number of points on the subject of an HP storage appliance without labels, making it difficult for readers to scan the text to determine if they should continue reading or move onto another section of the document. Although all the information in the first illustration is primarily conceptual, few structural features distinguish the points being made. In the structured illustration (Figure 3), we draw on our definition of a concept or topic to reorganize the information, a category that defines terms and provides information about a product feature, component, or complex procedure using an argument, description, exposition, or narration. The structured content separates what the product does from the benefits, system requirements, and installation options.
When content is separated into chunks, writers can apply XML markup tags that describe each concept in the overview independently. Structural distinctions provide options for writers and editors to collaborate and decide how best to reuse the content. For example, all four sections of the structured illustration can be set up as a reusable module in the CMS if the writer plans to share content for a printed document and for an online help system. Alternatively, if the content is will be used in marketing materials for storage products or in related product documentation, establishing these four chunks as separate modules in the CMS might be more practical—a decision that would come from the team collaborating on that project. That is, the first two modules, “CASA Overview” and “Benefits,” could be integrated into a marketing brochure; alternatively the first and third modules, “CASA Overview” and “Supporting Operating Systems,” could be featured in a technical document pertaining to storage interoperability.

Other options are possible in a CMS when drawing on this content as one or more modules. Writers can integrate the modules with unique or specialized content. Because the CMS stores information in many collections, writers mix and match modules with unique content targeted to a particular audience. While the modules, which I liken to LEGO bricks, are stored in a “modular” collection, documents that reference these modules are stored in a separate “book” collection. Thus, single-sourcing affords writers with enormous flexibility and opportunity for creating content with rhetorical significance. When writers draw on modules, they can add introductory and transitional material to place the “brick” in context for readers. To ensure that the modules and unique content, including transitions, are placed together correctly, editors review completed documents for coherence.

The system is also flexible enough to accommodate requests for changes to modules if the documentation team decides that the existing content no longer accommodates a given audience. The CMS provides a feature called “workflow,” which enables any team member to propose changes to modules. Once a proposal is initiated, editors and the team members whose work could be affected by the change are notified. Editors will schedule meetings for team members to consider the implications of the proposed change. If everyone agrees that the change is suitable to all documents using a module, the proposal is accepted and the CMS will automatically process updates wherever the module is referenced. However, if the team disagrees, it might be necessary to “clone” the module for another instance of reuse. Dialog among team members is vital for ensuring that content modules are ready for particular rhetorical contexts and to meet the needs of a given audience.

This process of structuring content and managing reuse just discussed forms the basis of my design of a structured writing curriculum. What follows is a description of a structured writing course that emulates the interactions of a documentation department in a real business setting. Depending on the number of students, the class can be divided into 3 or 4 project teams. Each team should consist of writers, an editor, and a project lead and will be responsible for structuring portions of a complete documentation set. Also, the class as a whole will be responsible for submitting a content management plan for real product documentation downloaded from a company website. It is not difficult to find large sets of unstructured legacy product documentation on company Internet. A complete set often includes, for example, a reference guide, installation and configuration guide, getting started guide, user guide, service and maintenance guide, troubleshooting guide, and release notes.

Volunteers will be selected as part of a group to develop structured writing guidelines for the whole group. The volunteers—at least one from each team—will devote 20 percent of their time formulating the guidelines and 80 percent of their time contributing to structuring the
assigned documents. Having an editor on working on the guidelines is beneficial. For this reason, editorial support may need to be distributed to assist the editor who is working on the structured writing guidelines.

Descriptions of the four modules for a structure writing curriculum follow and are depicted in Table 2, which provides specific suggestions for learning objectives, readings, discussion topics, and assignments.

Table 2. Suggestions for implementing a structured writing curriculum
Table 2. Suggestions for implementing a structured writing curriculum

<table>
<thead>
<tr>
<th>MODULE 1: Defining Structure</th>
<th>Learning objectives</th>
<th>Readings</th>
<th>Online/class discussions</th>
<th>Assignments</th>
</tr>
</thead>
<tbody>
<tr>
<td>- How to define structure</td>
<td>- Crowley, S. Ancient rhetoric for contemporary students</td>
<td>- Crowley, S. Ancient rhetoric for contemporary students</td>
<td>- What is the difference between structured writing and XML?</td>
<td>- Select a set of product documentation posted on company sites for the group to analyze, map, and restructure</td>
</tr>
<tr>
<td>- The relationship between structure and content</td>
<td>- Hackos, J. Content management for dynamic web delivery</td>
<td>- Hackos, J. Content management for dynamic web delivery</td>
<td>- How does structure help readers?</td>
<td>- Assign an equal number of documents to each writing team</td>
</tr>
<tr>
<td>- Structure at the micro and macro levels</td>
<td>- Hsu, A. Structured writing at twenty-five</td>
<td>- Hsu, A. Structured writing at twenty-five</td>
<td>- What are the different ways of understanding structure?</td>
<td>- Identify document type and information types represented by the documents set</td>
</tr>
<tr>
<td>- Inventory of document types and information categories</td>
<td>- Lay, M. and Karls, W. Collaborative Writing in Industry</td>
<td>- Lay, M. and Karls, W. Collaborative Writing in Industry</td>
<td>- Why is it important to understand structure at different levels?</td>
<td>- Identify requirements for structured writing guidelines for the group</td>
</tr>
<tr>
<td>- Overview of collaborative writing</td>
<td>- Pitblard, N. Instructional Design, Information Mapping, &amp; Structured Writing</td>
<td>- Pitblard, N. Instructional Design, Information Mapping, &amp; Structured Writing</td>
<td>- What does structured writing relate to single sourcing?</td>
<td>- Overview of structured writing guidelines for the group</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MODULE 2: Structuring Content</th>
<th>Learning objectives</th>
<th>Readings</th>
<th>Online/class discussions</th>
<th>Assignments</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Comprehensive content management strategy</td>
<td>- Annet, Hackos, Lay/Karls, Reddley (above)</td>
<td>- Annet, Hackos, Lay/Karls, Reddley (above)</td>
<td>What is a comprehensive content management strategy?</td>
<td>- Create structured writing guidelines for the group</td>
</tr>
<tr>
<td>- Initial top-level content analysis</td>
<td>- Abel, S. Content Management and the Need for Change in Technical Communication</td>
<td>- Abel, S. Content Management and the Need for Change in Technical Communication</td>
<td>- How do issues in the documentation affect readability and usability?</td>
<td>- Structure designated sections of documents</td>
</tr>
<tr>
<td>- How to identify issues: redundancy, inconsistencies, misrepresentations</td>
<td>- Snavely, P. How to Speak XML</td>
<td>- Snavely, P. How to Speak XML</td>
<td>- What benefits can restructuring have on readers?</td>
<td>- Streamline redundant text</td>
</tr>
<tr>
<td>- Defining documentation types and information categories</td>
<td>- Billings, C. Content Has Structure</td>
<td>- Billings, C. Content Has Structure</td>
<td>- How does XML markup relate to structured writing guidelines?</td>
<td>- Editorial review of structured content</td>
</tr>
<tr>
<td>- Understanding user-centered design</td>
<td>- Edlefsen, L. and Lumford, A. Collaboration and Concepts of Authorship</td>
<td>- Edlefsen, L. and Lumford, A. Collaboration and Concepts of Authorship</td>
<td>- What must be considered when deciding how to divide writing tasks among team members?</td>
<td>- Estimates of time to structure complete documentation set</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MODULE 3: Analyzing Content</th>
<th>Learning objectives</th>
<th>Readings</th>
<th>Online/class discussions</th>
<th>Assignments</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Document analysis</td>
<td>- Annet, Hackos, Lay/Karls, Reddley (above)</td>
<td>- Annet, Hackos, Lay/Karls, Reddley (above)</td>
<td>What is the impact of the expected content from style?</td>
<td>- List of XML markup elements that correspond to structured writing guidelines</td>
</tr>
<tr>
<td>- Product analysis</td>
<td>- Abel, S. Content Management and the Need for Change in Technical Communication</td>
<td>- Abel, S. Content Management and the Need for Change in Technical Communication</td>
<td>How can reusable content be mindbend?</td>
<td>- Ensure consistent application of guidelines across all teams</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>MODULE 4: Reading Content</th>
<th>Learning objectives</th>
<th>Readings</th>
<th>Online/class discussions</th>
<th>Assignments</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Comprehensive single source strategy</td>
<td>- Albers, M. Single sourcing and the technical communication career path</td>
<td>- Albers, M. Single sourcing and the technical communication career path</td>
<td>How can single sourcing improve or impact usability?</td>
<td>- Complete XML markup applied to one document in the set (Osvald, In-Depth)</td>
</tr>
<tr>
<td>- Single source strategy and content management</td>
<td>- The technical editor and document databases</td>
<td>- The technical editor and document databases</td>
<td>What topics prompt agreements? Disagreements?</td>
<td>- Complete content maps for the document set (Osvald, In-Depth)</td>
</tr>
<tr>
<td>- Build cohesive, consistent, coherent modular documents</td>
<td>- Harte-Carison, M. On writing, technical communication, and information technology</td>
<td>- Harte-Carison, M. On writing, technical communication, and information technology</td>
<td>How do you learn your audience?</td>
<td>- Complete content maps for the document set (Osvald, In-Depth)</td>
</tr>
<tr>
<td>- How to manage transitions with XML</td>
<td>- Harmandaris, N. Commentary on epistemology of writing</td>
<td>- Harmandaris, N. Commentary on epistemology of writing</td>
<td>How does your team overcome technical roadblocks?</td>
<td>- Complete content maps for the document set (Osvald, In-Depth)</td>
</tr>
<tr>
<td>- Establish best practices for collaboration</td>
<td>- Williams, J. The implications of single source technical communication</td>
<td>- Williams, J. The implications of single source technical communication</td>
<td>How can single sourcing improve or impact usability?</td>
<td>- Complete content maps for the document set (Osvald, In-Depth)</td>
</tr>
</tbody>
</table>
MODULE 1: DEFINING STRUCTURE

In Module 1, the class begins with a series of readings that will introduce students to structured writing, rhetoric, and XML concepts. Related assignments ask students to identify structure in ancient and modern texts, including the use of XML markup to describe types of structures used in particular texts. Ray (2003) provides basic information about markup in *Learning XML*, especially the second chapter on Markup and Core Concepts. Students can use this textbook, for example, to evaluate a technical advisory issued to customers. This type of document often contains the following structural components: document title, version, product name, and description. The XML markup of these components might appear as follows:

```
<advisory type=technical>
  <title>Technical Advisory</title>
  <revision>1.0</revision>
  <productname>Product XYZ</productname>
  <description><comment>Add descriptive information about product and reason for the advisory</comment></description>
</advisory>
```

After learning about basic structure, each team will need to recommend an integrated documentation set for the class to work on throughout the semester or semesters, presenting arguments that explain why the selection is suitable. The class will need to agree on the complete set of documentation and what specific texts each team will work on. Once the class is in agreement, each team will study the assigned documents, evaluating their quality, structure, and requirements for improvement. These points should be recorded and shared with the class so that overarching requirements for the complete documentation set are catalogued, including required types of documents and categories of information.

Next, each team will decide what team member should participate on the structured writing project to formalize guidelines. As soon as these members are selected, the whole class should begin studying structured writing theory, including the methods of Horn, Ament, Rockley, and Hackos. The class should discuss the similarities and differences of each method. The structured writing group will record each team’s preferences to begin preparing an initial draft of the guidelines.

MODULE 2: STRUCTURING CONTENT

The focus of Module 2 is to secure agreement on the draft structured writing guidelines and to implement those guidelines so that all documents in the collection adhere to the same standards. Because this effort entails significant teamwork, students should read about collaborative writing to learn how best to interact and work through issues that might arise. As the guidelines are being reviewed, the teams also need to continue evaluating the quality of the documents assigned to them. Initial analysis of the content will help the teams identify redundancies, inconsistencies, and readability problems. This information will help them verify the usefulness of the structured writing guidelines so that they can be modified as necessary. Each team should also try out the guidelines to determine if the document types and information categories are sufficient. Based on feedback from each team, the structured writing group can update the guidelines for the class to follow. Each team should then estimate how long it will
take to structure the assigned documents—to determine if all of the content can be structured completely. If not, the team must agree on what can be finished, by whom, and by when.

The teams should then begin structuring the documents. The tools used to restructure the text are not critical at this point. While I am not discouraging the use of an XML application, an XML should not be used if it draws attention away from structure and effective writing. Rather, students should study XML to understand how it works conceptually and how it reinforces structure fundamentally. Structure will emerge as students effectively employ the information categories, remove redundancy, streamline wording, clarify meaning, and standardize similar concepts expressed differently. As sections are completed, they should be submitted to the editors to keep the document workflow moving, preventing an editorial backlog. The editors need to verify the consistent application of guidelines across all teams. Subject matter experts are not part of this classroom experience, but students can learn how to work with feedback from individuals outside the class by conducting user-centered design tests related to documentation usability (see the STC Usability and User Experience Usability Toolkit). Once this feedback and editorial comments are integrated, students can identify what XML markup elements will support this structure, reflecting the structured writing guidelines.

MODULE 3: ANALYZING CONTENT

The emphasis in Module 3 is on analyzing content, which will enable the teams to devise content maps. When high-level or top down maps are analyzed alongside in-depth or bottom-up maps, the class has vital material for a comprehensive content management plan. Mapping content from the top down enables students to consider where topics might appear in various types of documents (output requirements), helping them to understand their audience, as shown in Table 3. The top-down map will require collaboration across all teams. A map from the bottom up allows students to study the topics addressed in each document and compare them to similar or identical ones in related documents, as shown in Table 4. While the mapping exercises theoretically can occur before structuring, the process is often more effective when carried out after content is structured—to obtain meaningful mapping results that more clearly show identical, similar, and unique content. Once each team has created bottom-up content maps for its assigned documents, all of the team maps must be compared against one another. This comparison will reveal information structures that are not yet identical when they should be the same. Because working with content maps requires intense teamwork, it is important for students to continue reading about collaborative writing strategies. Any change requests to the guidelines need to be incorporated by the group assigned to the structured writing guidelines.

Table 3. Top-down content map
## Top-down Content Map

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Product Overview</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Benefits</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Supported Operating Systems</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Installation Options</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Product Features</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Interfaces</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Peer Nodes</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>Hosts</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Storage Virtualization</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>User Privileges</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Switch Management</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Feature Upgrades</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>Product Management</td>
<td>X</td>
<td>X</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

### Table 4. Bottom-up content map

<table>
<thead>
<tr>
<th>Bottom-up Content Map: User Guides</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Legend</strong></td>
</tr>
<tr>
<td>S=similar</td>
</tr>
<tr>
<td>IC=identical</td>
</tr>
<tr>
<td>U=unique</td>
</tr>
<tr>
<td>none=not covered</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Book Contents</th>
<th>Product X1</th>
<th>XP10000</th>
</tr>
</thead>
<tbody>
<tr>
<td>About this document</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>Related information</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>Document conventions and symbols</td>
<td>IC</td>
<td>IC</td>
</tr>
<tr>
<td>HP technical support</td>
<td>IC</td>
<td>IC</td>
</tr>
<tr>
<td>HP storage web site</td>
<td>IC</td>
<td>IC</td>
</tr>
<tr>
<td>HP authorized resellers</td>
<td>IC</td>
<td>IC</td>
</tr>
<tr>
<td>Helpful web sites</td>
<td>IC</td>
<td>IC</td>
</tr>
<tr>
<td>Revision history</td>
<td>U</td>
<td>U</td>
</tr>
<tr>
<td>Chapter 1 Overview</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>Continuous Data Availability</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>Nondisruptive service upgrades</td>
<td>IC</td>
<td>IC</td>
</tr>
<tr>
<td>Connectivity</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>Fibre Channel</td>
<td>U</td>
<td>U</td>
</tr>
<tr>
<td>ESCON</td>
<td>U</td>
<td>None</td>
</tr>
<tr>
<td>FICON</td>
<td>U</td>
<td>None</td>
</tr>
<tr>
<td>Scalability</td>
<td>U</td>
<td>U</td>
</tr>
<tr>
<td>Data integrity and high availability</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>Summary of features</td>
<td>U</td>
<td>U</td>
</tr>
<tr>
<td>Web-based array management</td>
<td>S</td>
<td>S</td>
</tr>
<tr>
<td>Operating systems</td>
<td>U</td>
<td>U</td>
</tr>
<tr>
<td>External storage</td>
<td>U</td>
<td>U</td>
</tr>
<tr>
<td>Included components</td>
<td>U</td>
<td>U</td>
</tr>
</tbody>
</table>
MODULE 4: REUSING CONTENT

The goal of Module 4 is to create a systematic content management plan and to structure as much of the content as possible. The inventory of information obtained via content mapping contributes largely to the plan. Depending on the size of the documentation set and the experience of the students, structuring all the documents may not be possible. The objective is for the students to learn structured writing methodology and collaboration skills, not to structure every last sentence. If all the content will not be structured, the class should reach an agreement about what documents can be completed and commit to achieve this end by a certain date, a very typical real-world scenario.

Finalizing the plan also entails identifying what information will function as reusable modules in the CMS. That is, identical content items listed in content maps are ideally suited to become database modules. In addition, similar content may also be set up as modules when small differences are coded as “profiles,” such as the names of different operating systems, or “text entities,” such as variables (software versions) that will change throughout a document. Together, the teams must decide how the identical and similar content should be configured as modules for reuse. As described earlier, the identical content might be made up of a number of information categories or just a portion of one information category. The class needs to decide if it makes sense to configure identical content into one or more smaller modules that can be reused for different purposes. What constitutes a module depend on the needs of the audience. These decisions should be recorded in the content management plan. While students will not be practicing how to process change requests using workflow, they should understand that modules can be modified or reconfigured to accommodate customers.

In terms of combining modules with unique content, the plan should articulate how transitions will be handled—where they will be needed to link all the units of information. Editors can be asked to place special emphasis on these areas so that the documents are cohesive and coherent. The plan should also provide general information about metadata, the controlled vocabulary that should be established to describe the nature of the modules. A metadata strategy ensures that writers can locate information for ongoing reuse.

If time permits, each team should prepare a short a final paper or a list of best practices for achieving effective collaboration and communication. Alternatively, the teams might consider developing one of the discussion topics listed as a final paper.
CONCLUSION

Admittedly, there are many challenges facing those who want to implement structured writing and single sourcing. Yet the benefits of effective single sourcing strategies include clear, consistent information for customers and business efficiencies. While implementation depends on specialized tools of automation, the focus should be on methodology. Technology will have a much greater impact with a dynamic team that understands the overall strategy and has been trained to write and edit content effectively.

The differences of opinion among experts in academia and industry over the merits of single-sourced content are important for improving documentation that customers need to carry out daily operations. I share a real-world example of substantial improvements achieved by restructuring content for reuse at HP. The positive outcomes we have realized over the course of four years suggest that single-sourcing technology, when implemented in a collaborative environment and when there is clear direction, can improve technical documentation. Effective collaboration helps to overcome the concerns mentioned earlier, those raised by Albers (2000, 2003), Weisse (1993, 2002), and Clark (2002).

As suggested by a colleague, tools should be a vehicle for well-organized, effective writing:

I would argue that any approach to developing technical information can yield the full spectrum of outcomes, from masterpieces to outright catastrophes. Where you land on the spectrum is largely a matter of talent, skill, and execution. With modular writing, the ultimate challenge (and the real fun of our profession) is to produce high-quality documents that betray no evidence of their origins -- no fingerprints left over from multiple authors and the slicing and dicing of their tools. It is by no means easy to accomplish this....[Y]ou need a clever information architecture and a strong editorial function to enforce the architecture across topics and modules. In addition, you need tools that are developed with writer's and reader's needs in mind...; processes that bring writers and tools together in a productive, harmonious way; and a culture that facilitates and rewards innovation, collaboration, and continuous improvement. If all these things are in place and operational, an organization can produce top-quality documents and benefit from high rates of productivity, all at once. (Petrie, personal communication)

Without question, careful orchestration must occur among team members. Just as individual instruments retain their unique sound within the context of the whole musical ensemble, writers can maintain their identity in the presence of a well-synchronized documentation team. Organizations require highly skilled technical communicators to accomplish this synchronicity, which depends on a strong partnership between business and academia.
REFERENCES


Rockley, A. & Jodee Clore, J. (2002, May 4-8). Avoiding the content silo trap™, enterprise content management. STC’s 49th Annual Conference, Nashville, TN.


Formal example

Formal examples are not typically used in HP technical documentation. A formal example should include an introduction in the text that precedes it, and has a number and title.

Required elements

The required elements are:

- `example` to create a container for the example content
- `title` to provide the title of the example
- One of the optional elements, such as `para` or `inlinemath`

Optional elements

Some of the optional elements are:

- `inlinemath` for an equation
- `mediaobject` for a media object, such as a video
- `inlinemediaimage` for an illustration of a process or a procedure
- `inlinetable` for a table that provides supplemental information
- `para` to type an example
- `inlinemediaimage` for a screen shot
- `litterallayout` to type an example without line wrapping

Formal example in XML code

```xml
<example>
  <title>Sample output for ping www.yahoo.com</title>
  <screen>Pinging www.yahoo-ht2.akadns.net [69.147.114.210] with 32 bytes of data:
  ...
  Reply from 69.147.114.210: bytes=32 time=109ms TTL=50...
  Reply from 69.147.114.210: bytes=32 time=94ms TTL=50...
  Reply from 69.147.114.210: bytes=32 time=94ms TTL=50...
  Reply from 69.147.114.210: bytes=32 time=94ms TTL=50...
  ...
  Ping statistics for 69.147.114.210:
  Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
  Approximate round trip times in milli-seconds:
  Minimum = 94ms, Maximum = 109ms, Average = 97ms
</screen>
</example>
```

Figure 1 “Example Information Structure,” a partial instruction for writing for an example using XML markup.
ROBIDOUX 23

About the Appliance

The Continuous Access Storage Appliance (CASA) is part of a family of storage network offerings from Hewlett-Packard. The appliance allows users to centrally manage storage resources and share and protect critical business information. The appliance is installed between the host servers and the storage devices through Fibre Channel connections and enables intelligent control over the flow of data.

The appliance is designed primarily for heterogeneous data replication and data migration. The benefits of using the Continuous Access Storage Appliance include:

- Enterprise-level business continuity capabilities for heterogeneous storage environments
- Cost-effective local and remote data replication
- Increased storage utilization and control of heterogeneous storage capacity
- Online migration from old storage to new
- Increased productivity by enabling more online recovery of data

The appliance supports all major operating systems (HP-UX, Windows 2000, Windows NT, IBM-AIX, Solaris, and Linux) and distributed network applications, such as e-mail, Internet access, and enterprise resource planning. It is also compatible with HP Command View and HP OpenView software products.

HP installation specialists or approved third-party HP reseller partners typically install and configure the appliance. The appliance can be supplied either:

- Preinstalled in a full HP cabinet rack system (most common)
- Preinstalled in a rack with additional HP StorageWorks virtual arrays (provides a complete storage solution for small-to-medium sized companies)

Following initial installation and configuration, you can access the appliance with Internet Explorer 5.5 or 6.0 and manage the storage environment from any workstation that is mapped to the local network. In addition, HP offers a range of support options to assist you with the ongoing service and maintenance of the appliance.

Figure 2 Unstructured content.

CASA overview

The Continuous Access Storage Appliance enables you to centrally manage storage resources and share and protect critical business information. Authorized HP personnel install the appliance and connect it to your hosts and storage arrays using Fibre Channel cables.

Benefits

The appliance is designed primarily for heterogeneous data replication and data migration. The benefits of using the appliance include:

- Enterprise-level business continuity for heterogeneous storage arrays
- Cost-effective local and remote data replication
- Increased storage utilization and control of heterogeneous storage arrays
- Online migration between storage arrays
- Increased productivity by facilitating online data recovery

Supported operating systems

The appliance supports all major operating systems (HP-UX, Windows Server 2003, Windows 2000, Windows NT, IBM-AIX, Solaris, and Linux) and distributed network applications, such as Internet access and enterprise resource planning. It is also compatible with HP Command View and HP OpenView software products.

Installation options

The appliance can be installed in either:

- The standard HP rack system
- A rack that meets appliance requirements

Figure 3 Content after restructuring contains categories of information.