The Definitive Guide to Interwoven TeamSite

Brian Hastings and Justin McNeal

Apress®

The Definitive Guide to Interwoven TeamSite

ISBN-13: 978-1-59059-611-1

ISBN-10: 1-59059-611-0

Printed and bound in the United States of America 9 8 7 6 5 4 3 2 1

Lead Editors: Jason Gilmore and Keir Thomas
Technical Reviewer: Tom Shell
Editorial Board: Steve Anglin, Ewan Buckingham, Gary Cornell, Jason Gilmore, Jonathan Gennick, Jonathan Hassell, James Huddleston, Chris Mills, Matthew Moodie, Dominic Shakeshaft, Jim Sumser, Keir Thomas, Matt Wade
Project Manager: Denise Santoro Lincoln
Copy Edit Manager: Nicole LeClerc
Copy Editor: Kim Wimpsett
Assistant Production Director: Kari Brooks-Copony
Production Editor: Kelly Winquist
Compositor: Diana Van Winkle
Proofreader: Lori Bring
Indexer: Toma Mulligan
Artist: Diana Van Winkle
Cover Designer: Kurt Krames
Manufacturing Director: Tom Debolski

Distributed to the book trade worldwide by Springer-Verlag New York, Inc., 233 Spring Street, 6th Floor, New York, NY 10013. Phone 1-800-SPRINGER, fax 201-348-4505, e-mail `orders-ny@springer-sbm.com`, or visit `http://www.springeronline.com`.

For information on translations, please contact Apress directly at 2560 Ninth Street, Suite 219, Berkeley, CA 94710. Phone 510-549-5930, fax 510-549-5939, e-mail `info@apress.com`, or visit `http://www.apress.com`.

The source code for this book is available to readers at `http://www.apress.com` in the Source Code section. You will need to answer questions pertaining to this book in order to successfully download the code.

CHAPTER 1

What Is Content Management?

Hello, and welcome to *The Definitive Guide to Interwoven TeamSite*. You have made an extremely important investment for yourself and your company that will pay large dividends for your content management system (CMS) implementation. We have written this book hoping to assist others in tackling the daunting task of implementing an enterprise CMS (ECMS). We have drawn upon our combined decade of experience implementing and managing the Interwoven CMS, gathering tips and tricks learned along the way, and combined the knowledge into one source, this book. With the help you'll find in this book, you will have fewer sleepless nights and a more successful CMS implementation.

What Is Content?

Before we speak about content management, you should take a moment to better understand the concept of content itself. Only in this way can you more fully understand content management, the impact it has had in modern-day e-business, and the effectiveness, usefulness, and vital role of CMSs.

We define *content* in an organization as any organizational informational asset that exists in an electronic medium. Although it can be argued that any physical information resource can be classified as content, in the context of this book we will not consider any source as content until it exists in an electronic form. Some typical examples of content include e-books, manuals, publications, web pages, video files, music, instructional material, promotional material, help text…the list goes on and on. You can classify anything as content for an organization if it fits into all of the following sections.

Content Has a Classification Type

Typically the classification type is organizational; in other words, content can be categorized by an organizational unit such as marketing content, legal content, general content, privacy content, and so forth. A company that performs only legal services may use contracts, wills, deeds of trust, billing invoices, summons, and other legal type content, while a medical publishing company may depend upon drug data sheets, patient handouts, medical books, and drug news as content.

Content Has a File Type/MIME Type

Any piece of content has an associated file type, that is, the file extension that is tied to a particular program or standard. On the Windows platforms, file extensions tell the operating system the program or application to launch to service the specific file type. For example, an image file will generally have an extension such as `.gif`, `.jpeg`, `.jpg`, `.bmp`, `.png`, or `.tif`. In a CMS, file types are important to drive storage placement and delivery requirements, as well as to determine how certain files will be viewed. For example, in CMSs, the file type may control whether certain files are displayed in an iframe, a separate window, or the current window.

Content Has Metadata

Content has data attributes that describe it. These data attributes are described collectively as *metadata*. Metadata is used for many functions within a CMS:

Indexing data for search-related capabilities: You can intelligently add keywords to each content type via a defined taxonomy. A *taxonomy* is an intelligent mapping of taxons (the highest-level grouping) and taxa (subgroups under each taxon) that are unambiguous and, when taken together, consider all the possible values. For example, have you ever played the guessing game 20 Questions? This game usually starts with someone thinking of something that you then have to guess based on the responses of the 20 questions you get to ask. Usually you start by asking something similar to "Is it an animal, vegetable, or mineral?" If the other person playing with you answers your question by saying "Animal," then you may ask "Is it a dog, cat, or human?" You will then continue until you guess the subject or you run out of questions. Each of those high-level categories of Animal, Vegetable, or Mineral would be a taxon. The taxa would then stem from the higher-level taxon; for example, the taxa for the Animal taxon might be Dog, Cat, and Bird.

Clustering on metadata and exposing that clustering to search: For example, by clustering metadata, you can list all books or all documents from the marketing department or list all the content about a certain topic. With clustering metadata, you can capture important information such as the author name or publish date. When exposed to a search engine, you then provide the ability to list all books published by a certain author or, for example, all books published from 1995 to 2001. You could also capture the subject for specific topic-based searching. By clustering with metadata, you can accomplish advanced searches such as retrieving all the books published by a certain author, of a certain subject, *and* published from 1995 to 2001. By having clustering data available, you can group search results and create dynamic drill-down capabilities by dynamically rebuilding the search query.

Automating system-created content using an intelligent method: You can create new and complete content dynamically, where pieces of content are combined or aggregated into an entirely new piece of content based on system requirements or at run time. If content authoring is performed where content is created at the smallest possible level, then this is possible but still difficult to accomplish. Synthesizing content is especially useful for content such as marketing collateral and presentations that should be shared across an organization or several diverse sales teams.

Determining when data should be archived and when data should be removed: These topics are often overlooked in a CMS application. By setting metadata attributes and associating those attributes with content, you can manage the demotion or deletion of content. Additionally, if your corporation has retention standards, you can programmatically control when content is archived and even where the archived content is stored.

Determining when content is promoted or published: By associating metadata with your content, you can program that content's promotion date and time. A business content review process will control when the content has reached an acceptable publishing state, but why not also control when that content is made live?

Content Requires Storage

Simply put, content takes up space, which is typically database space or space on a file system. Some examples of storage space used to store content include network-attached storage devices such as those provided by EMC, relational databases such as Oracle or SQL Server, and file systems such as the Unix file system or NTFS. No matter what the content or the content's type, you must have adequate space in which to store it. All content has a certain number of bits and bytes that compose it. This collection of bits and bytes takes up disk space and must be accounted for in a CMS.

Content Has a Purpose Not Related to the CMS

Configuration files, templates, or any other files that are used by the CMS should not be considered actual content. Basically, any file that does not provide value to a consumer of the content is not considered content. People who use content are also known as *content consumers*.

Defining Content Management

Content management is the organizing, directing, controlling, manipulating, promoting, and demoting of content and/or digital informational assets within an organization. *Promoting* content means deploying content from the authoring environment to the content delivery environment, which is usually a web server. *Demoting* content means removing or rolling back content from the content delivery environment to the content authoring environment.

A CMS manages those various pieces of content described earlier. This is extremely important for organizations because as a business grows, so do the complexities of its content. Businesses have hundreds, thousands, and sometimes millions of pages of content, and with the overwhelming flood of this content, they need help! A CMS is much more than an off-the-shelf piece of software that a company purchases and configures for its specific needs. To understand content management, you not only have to have a view of the proverbial forest but also need to know where each tree belongs. Implemented properly, content management and a CMS can do the following:

- Improve delivery time from content creation to content promotion
- Improve content quality

- Reduce the cost of managing an organization's global or departmental content
- Reduce redundancies in content and reduce human error
- Eliminate orphaned files and broken links
- Automate content notifications and the review process
- Enforce legal and branding standards
- Improve content visibility and accountability throughout the delivery chain

Implemented poorly, a CMS system can be a virtual money pit, where no efficiencies are gained and many hours are wasted on a project that may provide little to no value to end users or consumers. You must understand two important facts regarding a CMS:

- A CMS will not improve your business process. You will have to expend some analysis cycles and rethink existing processes and procedures to enhance your business flow. Only by improving your existing process will you realize the true benefit of content management.
- A CMS is not free. Skipping the cost of the solution itself, you will be burning hours to implement a system. Once everyone starts to use the system and find out what the system can do, they will want more. This will also lead to additional implementations and maintenance. This is a topic we cannot reinforce strongly enough: setting up a CMS is more expensive than just building an initial website. You must be willing to accept the initial cost of building and deploying a CMS before you experience a return on investment. But you will have to trust that all of the hard work, project hours, and budget allotments will pay off in the end.

Recognizing the Business Need for Content Management

In the early days of the Internet, the need for sophisticated content management mechanisms did not exist. At that time, all you had were a few engineers who were more interesting in transmitting technical and scientific data than they were in using the Internet as the incredible _____. Well, you fill in the blank. These days the Internet is commonly relied upon as a marketing vehicle, supply-chain management channel, and distribution channel, among many other uses. Indeed, the Internet as a medium or transportation vehicle for content has come a long way. The bulk, complexity, and necessity of content management were not a problem in the past. The techie renegades did not worry about how their content would look, about when it would reach the other end of the world, or about how their company image (or brand) may be injured or helped by how their data was arranged or how fresh or stale that content was; they didn't even care about performing any specialized integration with unique application or business logic servers.

When companies came around to using the power of their content, you still found the renegades who knew the technology inside and out and the business folks who would try to explain exactly what they wanted for their content. These content renegades were in high demand because they were the only ones in an organization who understood the technology,

and therefore they were the only ones who could update that content. This was a problem in the past because the content was so tightly coupled with the technology.

In today's world, the needs of corporations are complex, and equally complex, if not more so, are their content (basically the data that makes up their Internet, extranet, and intranet sites) management needs.

GETTING TO KNOW SOME TERMS

An *Internet site* is any website that is publicly available over the World Wide Web. The Internet uses the public telecommunication networks infrastructure, HTTP over TCP/IP, and any computer that is connected to it to connect billions of users around the world. A company's Internet site usually contains marketing content and other nonconfidential information and may also serve to provide an entry point to secured site areas.

An *extranet site* is any website that is privately available and serves to connect users of the extranet site to the company that manages the extranet site. An extranet site uses Internet technologies and requires security and privacy that is usually managed via encryption, passwords, and tokens, as well as uses virtual private networks.

An *intranet site* is any website that uses Internet technologies and is available only to internal resources. An intranet is usually available only to employees of an organization and is usually used to provide information to those employees or to facilitate their working environment.

As the complexities of content has grown, the complexities of available tools required to access, implement, and modify that content have been reduced, allowing the ability for the true content experts (the marketing, sales, administrative, and managerial staffs) to better manage that content.

The business need for content management cannot be explained with a one-size-fits-all mentality; rather, it is driven by your own organization's need. Generally, all companies need some content management. However, the level of content management for individual companies will vary greatly. Your company's need for a CMS is determined by the following factors.

Amount of Dynamic and Static Content

Does your organization have several thousand pieces of content? If your organization has a large amount of content where the storage and management of that content is complex, then a CMS will help you. Managing many pieces of content and making that content available for content consumers can be a tricky undertaking. We typically find that the more discrete pieces of content an organization has, the more it must increase its information technology (IT) staff or administrative staff to manage this content. Having a CMS means never having to worry about where documents and content are stored or how to retrieve that content.

Complexities of Your Content

Complex content needs demand a CMS. Content built dynamically from many sources and content that depends on other content can lead to content management headaches. Without a CMS, content updates are forgotten, navigation is updated incorrectly, and content quality suffers.

Frequency of Updates or Additions to Your Content

When content changes frequently, typically more than once per day, a CMS is imperative. Humans simply cannot keep up with frequent content updates. A good rule of thumb is that if your content has more than 20 to 30 updates a month and one or more of the other factors apply, then you need a CMS. A CMS can contain the intelligence to know about dependent pieces of content and can alert the author to make those required changes. Without this system in place, controlling content updates can be confusing and time-consuming. A CMS will also streamline the authoring process with authoring tools such as what-you-see-is-what-you-get (WYSIWYG) editors.

■ **Note** A WYSIWYG editor is a piece of software that allows content creation without the use of complicated markup characters and shows the content author exactly how the finished content will look while the authoring is taking place. For example, creating a Microsoft Word document in Print Layout mode gives you a WYSIWYG view. Some popular Hypertext Markup Language (HTML) WYSIWYGs include Microsoft FrontPage and Adobe GoLive.

Archival and Retention Requirements of Your Content

CMSs can facilitate archival requirements by housing all the content in one location. Interwoven TeamSite has a concept called an *edition*, which is a snapshot of all web content at a point in time. With editions, entire content stores can be moved to tape or another backup media or retained in the CMS itself. If editions are maintained in the CMS, browsing and retrieving the content is as simple as selecting the edition and browsing to the appropriate content. By providing snapshots of content retention, requirements are easily met. For example, we have worked with clients who had to meet a U.S. Securities and Exchange Commission (SEC) retention requirement of three years. This meant at any time the SEC could mandate that this company's content be reproduced exactly as it was originally displayed anytime during the past three years. With editions, solving this requirement was easy. (You'll learn more about editions in Chapter 7.)

Statutory or Legal Requirements for the Use of Your Content

Do you have requirements to always publish content in two formats, one for the Web and one for print or optical media? Why continue to produce content in two formats manually when a CMS can produce this content automatically? Addressing legal issues—such as the Health Insurance Portability and Accountability Act (HIPAA), which requires you to always include legal approval for each content deployment—is easy with a CMS and a workflow.

Requirements of External Sources to Supplement Your Content

Content feeds from external entities, whether they are news feeds, stock feeds, or weather updates, can greatly enrich content and boost the usability of that content. A CMS can automatically detect these types of external data feeds, notify the appropriate content approvers, and begin the deployment process. Without a CMS, once again you rely on human interaction, which is error prone and lacks the efficiency of a CMS.

Delivery Requirements for Your Content, Including Different Presentation Types and Data Layouts

A CMS allows you to publish content in a variety of presentation types. By separating your data and presentation in a CMS, you can support clients that need print-ready documents, web-ready documents, and even documents for personal digital assistants (PDAs) or cell phones. By keeping the data separate, you can combine multiple data sources into an aggregate document. Think about this—if you have a single document or piece of content that requires multiple content authors, the CMS is the ideal solution. For example, company XYZ is producing a product catalog for a new gadget being produced in the company. Company XYZ has marketing authors for the marketing section of the product catalog, sales authors for the price and licensing section of the catalog, and product engineer authors for the technical specification sections of the product catalog. By maintaining each section as a discrete piece of content, all authors can work collaboratively at the same time on what will become the finished product catalog. When the content components are ready to be published, the CMS can facilitate the approval process and bind the components together into the final product catalog. The CMS can even generate the product catalog into any number of supported output formats such as Portable Document Format (PDF), HTML, or Wireless Markup Language (WML). Figure 1-1 depicts how content can be turned into the final format by combining a presentation with the data.

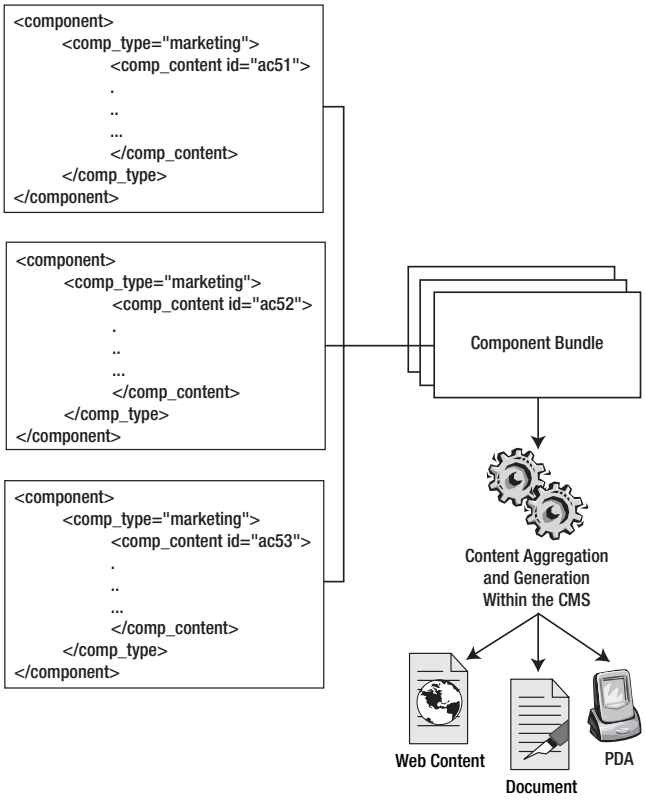

Figure 1-1. *Delivery requirements and componentizing of content in a CMS*

Authoring Requirements for Your Content, As Well As the Audience of That Content

Content authoring without a CMS is limited to technical authors and authors with specialized authoring tools. Dependent on the output format, an author will need HTML editors such as FrontPage or Macromedia Dreamweaver, PDF creation programs such as Adobe Acrobat, and any number of others depending on the publishing requirements for your content. Authors without a CMS will also need specialized knowledge of the content, knowledge of how to use the authoring tools, and technical knowledge of markup characters for some output formats. All this specialized knowledge means hiring people who have technical skills and who can be expensive. Maintaining and supporting several different authoring tools across your enterprise is also expensive. Licensing and deploying these authoring programs can equate to thousands of dollars per year. With a CMS, you eliminate these significant costs. CMSs have built-in editing and content authoring capabilities that eliminate the need for specialized editing and authoring tools. Additionally, the CMS editors have a user-friendly interface that anyone with basic word processing skills can use. Within a CMS you can also intelligently specify a specific organizational glossary where common terms can be reused automatically. This means you will provide content consumers with a consistent corporate voice, if you will, thereby reducing confusion with interpreting your content.

Metadata and Searching Requirements for Your Content

ECMSs such as Interwoven TeamSite have connectors to metadata creation programs. One such world-class solution is MetaTagger, also from Interwoven. Using these applications to manage your content, you can automatically associate the correct metadata with all your content. By having a consistent metadata schema, search engines are more productive, and indexing overhead is greatly reduced.

Syndication and Deployment Requirements for Your Content

Syndication is the process of integrating external content into your own company's content through a paid subscription. Without a CMS, this can be a manual process. The person responsible for content syndication will either be notified of content updates via an email or be notified by actually checking the source location constantly for content updates. If this person is sick or on vacation, content updates are missed, meaning that the subscription fees are not being utilized to their full potential. Once content updates are identified, the person responsible for adding the updated content usually has to copy the updated content and then use a predefined content authoring method to integrate that newly acquired content into their own content. Using this manual process, content updates are frequently missed, or worse, content is outdated before it even reaches the destination location. Often, a manual process such as this requires the newly acquired content be reformatted extensively to match the destination's style requirements. A CMS eliminates these problems. Content updates can be performed automatically within the CMS. The CMS can be notified by the subscription service and can intercept the email to ingest the content update. The CMS, once notified, can integrate the content automatically and appropriately. The CMS can then route the content for approval by a predefined human user of the CMS. When the content updates are approved, the content can be deployed to the destination location. By working with a subscription service, you can define the ingestion format ahead of time, and the CMS can programmatically modify the content to match the destination format.

Separating Data and Presentation

Separating data and presentation is a core concept and practice in regard to a CMS. We touched on that it is essential to separate your data completely from the presentation. Doing this allows your business solutions to be flexible. By separating or decoupling the data and presentation, you greatly improve data reuse. Enterprises spend vast amounts of resources manipulating their data into usable assets. For this reason, a major consideration when creating your data is the importance of storing it properly from the beginning. If successful, the remaining steps in the process for creating a CMS should come much easier.

Some industries have requirements that require the data be stored in its reproducible form. This means if a presentation has been placed on the data before it was published, not only would the data have to be kept for historical reasons but there would also have to be a way to reconstruct the document, unless the final document is stored as well. This could cost considerably more resources to store the same data that has been manipulated by several different presentations.

Presentations can be defined as the final look of the data or content. This finalized and presented content may be of various mediums, such as a website, a document for printing, or perhaps an email to all of your current customers. It does not matter what the final product will be; each presentation can be completely different. The fact is the same data can be used many times, in many time spans, and in many data formats. Figure 1-2 shows the concept of separating data from presentation.

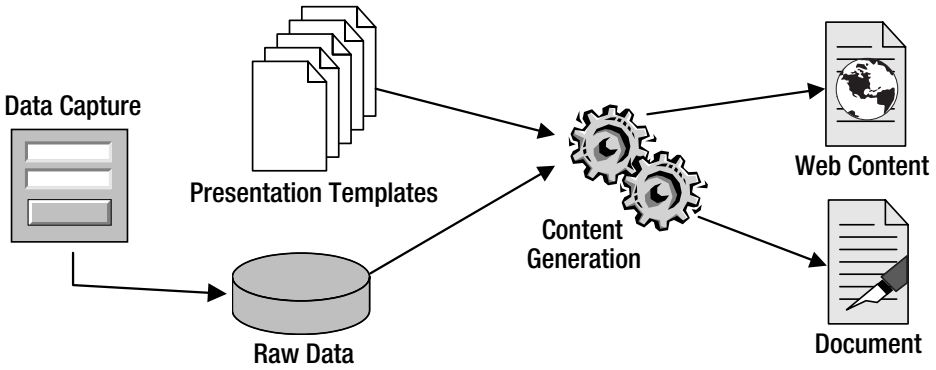

Figure 1-2. *Separation of data and presentation*

In Figure 1-2, data is entered into a data capture screen. This data is then stored in eXtensible Markup Language (XML) format in the raw data store or repository. Predefined presentation templates are then used to merge with this raw data to create the final generated output formats, one for the Web and one for print.

To illustrate this concept further, refer to the sample XML data file in Figure 1-3.

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- <?xml-stylesheet type="text/xsl" href="genHTML1.xsl"?> -->
<?xml-stylesheet type="text/xsl" href="genHTML2.xsl"?>
<famousquotes xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="quotes.xsd">
    <quote>
        <movie>Robocop</movie>
        <quoteline>Dead or Alive, you're coming with me.</quoteline>
        <spokenby>Alex Murphy/Robocop</spokenby>
    </quote>
    <quote>
        <movie>Cobra</movie>
        <quoteline>You're a disease, and I'm the cure.</quoteline>
        <spokenby>Marion Cobretti</spokenby>
    </quote>
    <quote>
        <movie>Superman II</movie>
        <quoteline>Come to me, son of Jor-El! Kneel before Zod!</quoteline>
        <spokenby>General Zod</spokenby>
    </quote>
</famousquotes>
```

Figure 1-3. *Famous quotes XML data file*

Even if you do not understand XML at this point, bear with us for a minute. We'll discuss XML in detail in Chapter 8, but for now it is not important to understand XML but rather to examine the general context of this file. This XML data file contains groups of famous movie quotes; each stored quote contains the movie title (annotated by a `<movie>` tag), the text of the actual quote (annotated by a `<quoteline>` tag), and the character who spoke it (annotated by a `<spokenby>` tag). Important to note is that this file contains only data. A web browser does not know how to display a quote, but we will tell the browser how to display this data by applying "presentation" templates to it.

By separating the data, in this example the movie quotes, from the presentation, we can then generate and present that data in multiple formats. Figure 1-4 shows an example of this data presented with an HTML table.

In Figure 1-4 the XML data, containing famous movie quotes, has been transformed via a template into an HTML file in table format. But the same XML data can be presented in any number of formats, depending on the number of presentation templates applied to it. Figure 1-5 shows the same data, presented in an HTML list format.

By keeping the data and presentation layers separate, data can easily be presented in multiple formats. If the data was coupled or intermingled with the presentation, the arduous task would then be to strip all the data from its current presentation and insert that data into another differently formatted presentation. This would be nonefficient, error prone, and expensive.

Figure 1-4. *XML data movie quotes file presented in an HTML table*

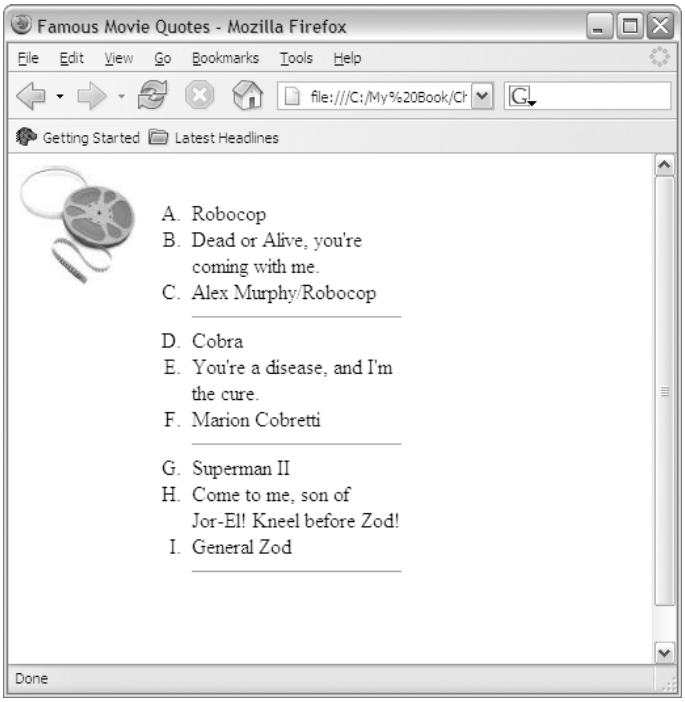

Figure 1-5. *XML data movie quotes file presented in HTML list format*

Presentations can be layered on top of each other. A good example of this is in an organization's corporate website. In a corporate site, a style guide provides the overall look and feel, and it can specify the various presentation templates of how everything fits together. Within an organization's content, several types of pages usually exist. There are frequently asked questions (FAQ), site maps, and general content. Instead of creating several pages of content containing redundant information, one can create content sections. Defined content sections within the overall presentation can allow one to build smaller pieces that will fit within the overall structure. This is often referred to as *synergy*, where the sum of the pieces as a whole is greater than the sum of those individual parts.

■ **Note** *Generating* content is the process of taking the data record, usually in ASCII (XML or text) format and combining it with the presentation template to "generate" the desired end product. The presentation template's responsibility is the overall look of the final product. The generated content is usually constructed through the combination of the two structures. The first would be the presentation template, and the second would be the XML data record. In a CMS, the generation process or generation engine may be a proprietary application, or it may be based on open standards.

Introducing Metadata

Metadata is data that describes other data. Picture a book for a second. Who wrote the book? How many pages does the book have? How much does the book weigh? What color is the book? How many chapters does the book have? All of this information describes the book and is known as *metadata*. Metadata is collected so that content can be categorized and searched. To illustrate this point, imagine the following scenario: You have just started a job at a large hospital, and a doctor would like for you to retrieve Mr. Smith's medical chart from the records department. Upon entering the records department, you are inundated with a plethora of shelves filled with file folders, any of which could be Mr. Smith's. Now, you could wander aimlessly around each shelf flipping through charts until you get lucky enough to stumble upon the correct chart, or you could use the records department lookup terminal to point you to the exact location. When you type Mr. Smith's name into the terminal, a search is executed on all the metadata collected and indexed for each patient. Any patient who has a match with the search terms you entered is returned to you as a search result, probably with directions on how to locate the patient's physical chart. Mr. Smith is returned as a search result, and you are able to quickly locate his chart.

Let's dig a little deeper—using this example, you can see how the following types of information would be useful as metadata for each patient:

- Social Security number (SSN)
- Patient last name
- Patient first name
- Date of birth
- Primary care physician
- Phone number
- Gender

For the example, the SSN, patient last name, patient first name, and so on, are effective pieces of metadata. However, you must use the most appropriate metadata for the type of content used in your organization. Although it may make sense to gather the SSN as metadata for a person, it definitely does not make any sense when your content is of type Book or any other content that does not have an SSN.

Discovering and collecting appropriate and useful metadata is one of the difficult steps that must be performed when implementing a CMS. To assist you, and the team you are working with, on this grueling task, you must ask five questions regarding each type of content in the proposed system:

- What data describes this content type?
- If I were looking for this content and I wanted to understand this content type, what information would I want to know?
- What about this data makes it unique among other pieces of content?
- What about this data makes it similar with other pieces of content?
- Is there any CMS feature or downstream function that must be done to the content and that relies on certain metadata for this purpose? For example, you may have archival requirements where it is necessary to have metadata such as archival date, published date, and expiration date. These dates would be set at content creation and content publish time so that the CMS application could then archive, delete, and so on, this content at the appropriate date.

Finally, always work with the content owners during this process, because they will know the best way to describe their data and will be instrumental in the metadata-uncovering process.

To illustrate the concept of metadata further, Figure 1-6 shows the metadata stored for the file `Ibm.html`.

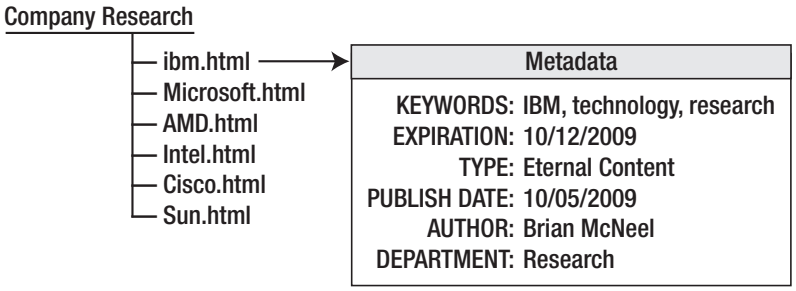

Figure 1-6. *Metadata stored for the* `Ibm.html` *file*

Imagine for a moment that within a website directory structure there is a directory named `Company Research`. This `Company Research` directory stores several pages of content: `Ibm.html`, `Microsoft.html`, `AMD.html`, and so on. Each respective page of content possesses its own associated metadata. From the IBM example, it is evident that keywords, expiration, content type, published date, author, and department are stored for each page of HTML content.

No matter what type of metadata is captured for your content, remember that time spent analyzing and capturing the appropriate metadata for your content is time well spent. Keep in mind that discovering halfway through the implementation process that you should have been keeping another piece of metadata can be frustrating and costly.

Using Templates

Templates within a CMS allow content to be created once and used multiple times in a variety of formats. You may need to display information for printing, information for a website, or information for wireless devices. You can accomplish all this via templates. Using one XML data file and three templates, you could generate PDF files for printing, Extensible HTML (XHTML) files for web publishing, and WML files for wireless devices such as cellular telephones or PDAs.

You can create templates in many ways depending on the CMS implemented. Typically in a CMS two main types of templates exist. The first type is the presentation template, and the second type is the data template. The presentation templates generate (believe it or not) presentations, or the look and feel, of the content. The second type collects the data that will be stored in XML format and be used as the substance, or the meat and potatoes, so to speak, of the content itself.

Figure 1-7 shows an example of what a typical presentation template configuration could look like. In this example, several sections are created separately and then assembled together as one page. The page includes header template and footer template sections so that the entire generated web page could have the same header and footer. The navigation section could be generated, and the promotions section could also be generated based on what content is being viewed on this page in the main content area of the page. This HTML page (comprised of different template sections) could be generated in advance to help reduce the strain of creating it each time it is requested. This is usually determined based on how often the content is updated, the processing power required to generate the pages, and what architecture is being used to serve the content.

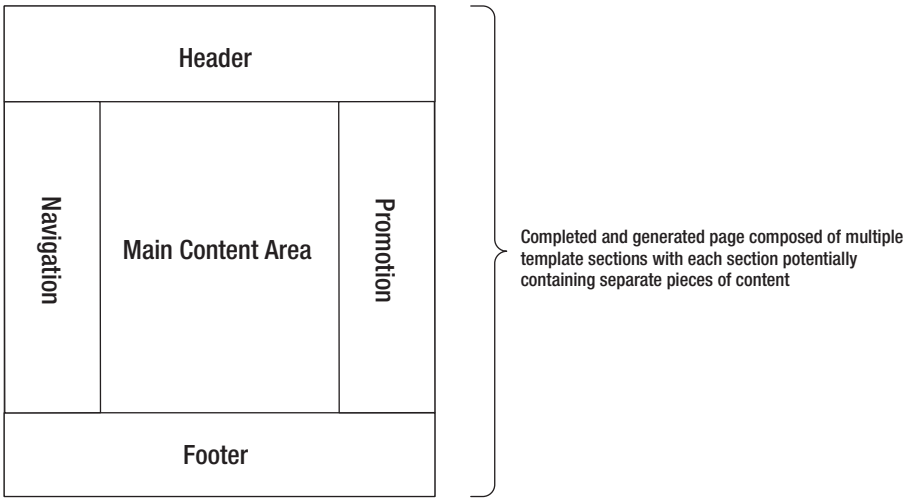

Figure 1-7. *A typical presentation layout*

The data template ensures that the data that is collected is the same in each record or XML file and could be as simple as an HTML form. By using data templates, it is easier to maintain data integrity because the data-entry tasks can be rigidly controlled and validated at the time of data entry. Without this integrity, the template system does not work.

In Figure 1-8 and Figure 1-9, we have taken the template concept and applied it toward an actual website. In Figure 1-9, the previously described template components are outlined to show how a generated page could be built based upon discrete template sections.

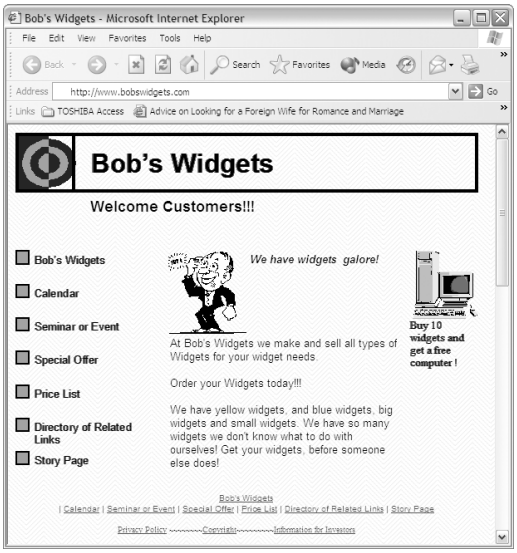

Figure 1-8. *Generated and completed site*

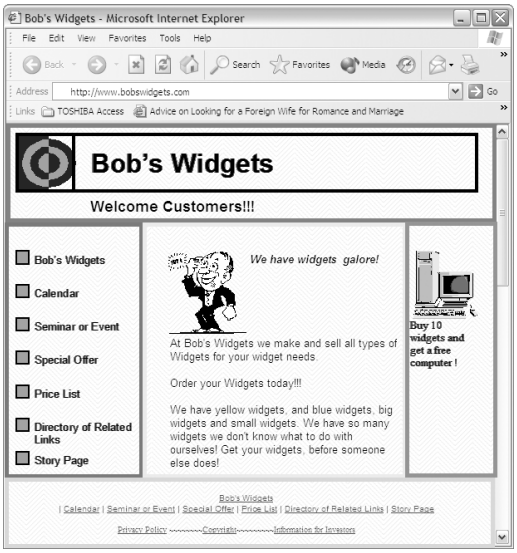

Figure 1-9. *Generated and completed site with content sections outlined*

Understanding Workflows

For almost anything you have to do in life that requires a predefined process, you must follow certain steps. In the CMS world, this process of steps is known as a *workflow*. More specifically, the process for which a document goes through during development can be turned into a series of steps that is a workflow. The workflow should model your business process as closely as possible. However, do not neglect the need to examine your existing process and make the business process more efficient before implementing that process as a workflow.

A workflow allows the CMS to enforce steps that could otherwise be averted. You can perform steps such as mandatory validations as needed. Workflow ensures that each step is fulfilled before the content is released to the public.

Figure 1-10 illustrates a simple workflow process. In this scenario, the content contributor submits content to the content approver. The approver can approve the content, sending it to the legal department, or they can reject it and return it to the contributor. If the content is rejected, then the content contributor will receive the content with instructions on what needs to be changed. After alleviating the problems, the content is then resubmitted for approval. Once the content approver has approved the content, the content goes to the legal department for review. If the content is rejected at this level, it will be returned to the original contributor for modifications, and the process repeats until the legal department gives the final approval needed to publish the content. The content is generated and distributed in two ways. The first is generated in a web-ready format in the form of HTML, PDF, or some other web format. The second is generated in a format that is optimized for print. As this illustrates, the system, even at a simple level, can be effective in contributing to the document development process.

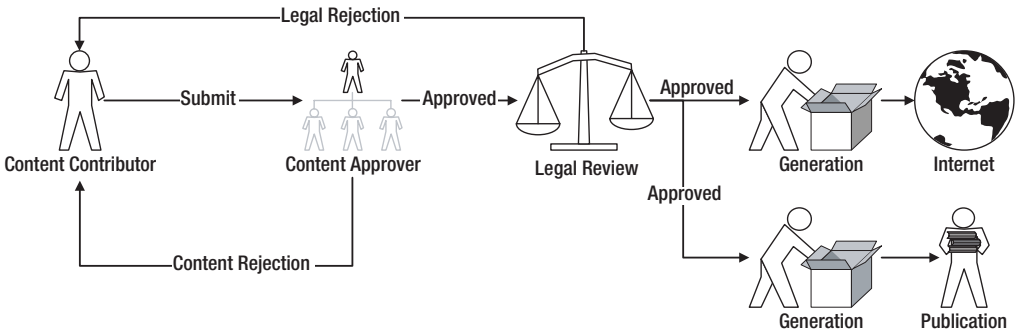

Figure 1-10. *Simple workflow process*

The logical conclusion of a workflow is the deployment of content. The deployment of the content deals with the path that the content takes through the physical and networking infrastructure of an organization.

In Figure 1-11, the workflow would execute in the content server, and upon workflow completion, the three types of presentation would have been generated and moved to the deployment server. The deployment server would have access through the corporate firewall to the web server. A process running on the deployment server would be responsible for placing the content into the appropriate location on the web server. The web server is then responsible for serving the content to its various requesting clients.

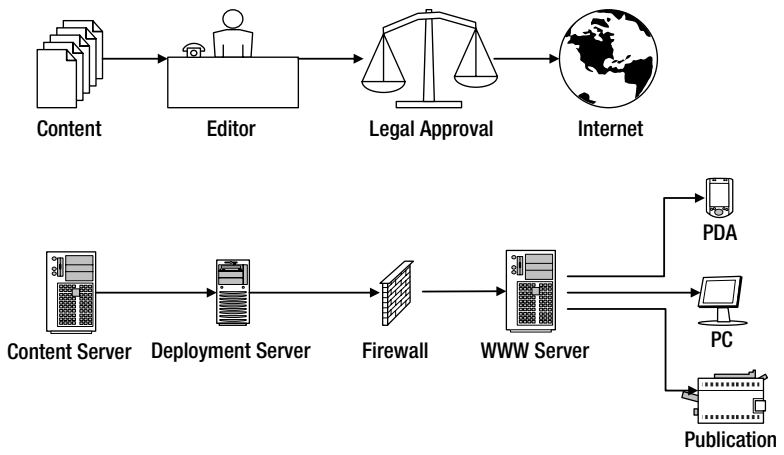

Figure 1-11. *Deployment path of content*

Introducing User Groups and Permissions

User groups and permissions are common to all technology-based systems, but if implemented properly in a CMS, they become one of the most powerful and important parts of the system. Any CMS implementation has many types of users; however, most user types can be divided into one of the following roles or groups:

Initiator role: This role can start a workflow for content creation, content deletion, or one of any number of defined workflows in a CMS.

Content contributor role: This role is responsible for creating and modifying content.

Reviewers/approvers role: This role is responsible for reviewing, approving, and rejecting content.

Hybrid role: This role's responsibilities can combine any of the previous three job roles. This role may also have distinct capabilities that are not present in other roles such as the ability to bypass workflows, approve processes, and submit multiple changes. Typically this role is reserved for administrators and super or power users.

Multiple interfaces into the CMS may be constructed and utilized based on the user type. There are many reasons to implement multiple user interfaces:

Training: Because you are using multiple interfaces, the interface can be kept clear of extra functions and features, which may confuse the trainee and may not be used by the specific role for which they are being trained.

Security: By eliminating additional functionality that is outside the security boundaries of this role, you can avoid many mishaps. Why allow access to an interface that, for example, allows the user to delete content when the user role should not even have that ability? Keep in mind that not all security breaches are purposeful, but many are caused accidentally. The interface should remove user temptation to find out what "this option does."

The bottom line is that if the role should not perform certain actions in the system, then do not present them with that option.

Branding: By having separate interfaces—between your clients, for example—personalization options can present a corporate brand image in each interface.

■ **Note** If you decide to implement multiple interfaces, you will be need to maintain both interfaces when adding new functionality. Also, individuals with multiple roles may be forced to log in to different interfaces depending upon what action they are performing.

Introducing Repositories

Repositories, or the data stores in a CMS, usually hold two types of data: binary data and ASCII data. ASCII data often comes in the form of XML. XML data, coupled with presentation files, will actually compose the generated content. XML data within the CMS is usually grouped into logical containers to facilitate the creation of the content and the generation of that content. Often times this logical structure is ordered in much the same way as a typical directory structure.

The binary data can take many forms and is typically proprietary in nature, such as PDF files, images, audio files, and video files. A quality CMS should not restrict you to one repository but should allow open and flexible integration with many different architectures and systems. Some possible integration points may include directory structures whether Unix or Windows based, relational database management systems (RDBMSs), Lightweight Directory Access Protocol (LDAP), and web services. The following sections cover some questions to consider and answer when constructing your repository.

Do You Understand Your Data?

Do you understand how your data will be used in the enterprise? How will data be packaged, and in what formats must that data be presented? Understanding the purpose of your data, which will become your content, is critical to a successful CMS implementation. We recommend meeting not only with your database administrators but also with your business owners. Do you understand what each field in your data repository (database or file system) represents? You must answer all these questions in order for you to make sound decisions while implementing a CMS.

How Much Data Will Be Stored?

What is the current size of your data that will be used to form content for your company? By knowing the current size of your data, you can derive requirements for storing that data. You must become intimately familiar with the current size of your data because this will impact the retrieval, archival, and retention requirements of your CMS.

What Is the Anticipated Future Growth of Your Data?

You do not want to have your company potentially spend millions of dollars on a system that is not scalable for the immediate and midterm future. The best way to evaluate your future growth needs is to first look at the amount of storage your current data requires. Next look at any anticipated new sites or new data stores that will be migrated to your CMS once the system is in place. Finally, you should develop a multiplier to apply to the size of your current data plus any anticipated new site or new data stores. Your multiplier should take into account many factors, including acquired data from internal sales and new business development. Your final data size should include the ability to scale for the next three to five years.

Luckily, Interwoven has developed some metrics to assist you with your sizing requirements. Let's look at an example to illustrate this point. Assume for a second that your current data storage is 500MB and the planned new site migration and new data store integration size adds up to 1GB in size. Interwoven states that the TeamSite CMS requires more storage space for TeamSite metadata, space to store multiple versions of the site content, and space to allow for future site growth. So taking all of this into consideration, you have 500MB + 1GB, which equals 1.5GB of total storage size. Remember the multiplier number—Interwoven recommends multiplying by 10 to 20 times the current storage size to allow for future growth. We recommend taking the higher number. Based on all of these factors in the example, the estimated storage requirements for the data in the example should be approximately 30GB of storage space.

How Complex Are the Data Relationships?

Does your data cross multiple repositories? What is the average size of each content component? Do you have numerous many-to-many relationships? A *many-to-many relationship* is one where data entities have many possible relationships to each other. An example of this is if you were modeling an article of clothing such as different models of pants. Each model of pants could come in different sizes, and each available size could relate to multiple models or types of pants.

What Are Your Data Retention Requirements?

This is really a question of how much money you are prepared to spend to support your requirements. How much data storage are you prepared to purchase? Do you know how long that data must be maintained for historical purposes? Do you plan on purchasing any additional storage space to provide data redundancy for your system? You must answer all these questions to ensure a successful implementation.

What Are the Data Retrieval Requirements of Your System?

How quickly must content be served up or presented from your repository to the calling program? Does the data have to be repackaged or transformed into a different format? Do style sheets have to be applied based on the system requesting the data? For example, one repository can serve multiple systems: one legacy system and a web-based system. The legacy system requires its data to be in a predefined EBCDIC 80-column format, while the web-based system requires its data to be presented in a base64-encoded XML format.

Knowing What to Look for in a CMS

The following information will allow you to draw from our considerable content management experience and provide you with CMS best-practice recommendations for the attributes that a quality CMS should possess. This information is extremely valuable when you or your company is evaluating a CMS for purchase.

Open and Flexible Architecture

A quality CMS should not force you to use proprietary technologies. It should be extensible with other technologies and highly scalable within an organization. The actual system should be modular, allowing the purchase and installation of pieces on an as-needed basis. If you need only workflow, then you can purchase only workflow functionality; if you need templating, then you need only purchase templating functionality.

Support for Multiple Data Formats

A quality CMS should allow integration with a variety of data sources, RDBMSs, XML database management systems (XDBMSs), and LDAP, for instance, and should allow storage of ASCII and binary data.

GETTING TO KNOW MORE TERMS

In an RDBMS, a management program allows you to create, update, and administer a relational database. A relational database is a structure where data is organized in formally described tables that allow you to retrieve and use that data without having to reorganize the data. In a relational database, the data is arranged in a way that maintains the data's relationship to the other data in the database. Some industry-known relational databases include Oracle, Microsoft SQL Server, and IBM DB2.

In an XDBMS, the data is arranged in a hierarchical XML format. This database format is extremely document-centric and allows extremely fast data retrieval but for specific purposes. Arranging the data in an XML format allows you to easily prune that data for specific component retrieval. For example, in the publishing environment, data can be placed into the XML database in its raw XML format. Then when you want to retrieve a specific chapter, a specific section, or even a specific paragraph, you can prune the data quickly, and only the section of content requested will be served. Typically, XDBMSs use the XML-based XQuery language to retrieve their data; some industry-known XML databases include Mark Logic Content Interaction Server, Xyleme Zone Server, and Virtuoso.

LDAP is based on the X.500 standard. In LDAP the data is represented as objects. Each object can be queried based on supplied attributes of that object. This allows objects to be queried without the requesting program knowing the location of that object.

Integration Capabilities with Existing Systems

A quality CMS should allow integration with many disparate systems, including repositories, legacy systems, and platforms.

Separation of Data and Presentation

This is a fundamental quality of a CMS; to facilitate the reuse of content and deployment in multiple presentation formats, data and presentation must be kept separate.

Flexible and Configurable Workflow Engine

A CMS should allow multiple workflows to be constructed, which will have the ability to replicate 90 percent of existing business processes. You will want a CMS that can house multiple workflows for multiple user groups, with multiple tasks or steps in each. Workflows must have the ability to move content through a delivery chain that is not part of the CMS; in other words, a quality workflow should have the ability to make changes in systems external to the CMS.

Effective Use of Standards

Does your CMS support XML as defined by the World Wide Web Consortium? Does it support PDFs, XHTML, XSL Transformations (XSLT), and connections to standard RDBMSs? The effective use of standards by a CMS platform will ensure that your organization is not locked into a proprietary format.

Content Authoring Toolset

This is a powerful factor in selecting a CMS. Does the CMS allow you to import non-template-created content? This would include Word files, PDF files, and Dreamweaver templates. The CMS should also inherently possess its own method of entering content, without the dependency on external or third-party tools, and may include such noteworthy features as spell check and autocomplete.

Multiple-User Authoring Environment

The CMS you choose must allow team-based collaboration of the content. If you have enough content to warrant a CMS, then you must be aware of the need to have multiple content authors, content contributors, and reviewers. The system must allow multiple users from multiple locations to manage, review, approve, and create content. A CMS is an enterprise application.

Metadata Creation

A CMS should allow for the seamless integration of metadata with the content it is defining. The metadata should not be limited in the data format that can be used. The system should be able to store binary metadata as well as ASCII metadata. Additionally, a quality CMS should be able to automatically extract some metadata from content files, such as when you need to extract PDF properties for the author, title, and date of creation from an imported PDF file.

Easy to Use

This cannot be stressed enough: a quality CMS should not force you to use highly technical resources to manage your content; with minimal training (which is therefore much less expensive), the administrative staff should have few problems navigating and using the CMS.

Version Management/Archival Capabilities and Rollback Capabilities

Version management can help eliminate risk from content authoring. The ability to roll back to any version gives the author the chance to easily correct any inadvertent mistakes such as forgetting to change an image on a certain web page. Also, the ability to archive those versions and easily restore each specific version may be essential for historical and legal requirements. Most good CMS have versioning and rollback functionality built into the repository. We will go into more detail later about this when we address the repository in Chapter 7.

Should Meet Your Specific Content Needs

This may be the most important consideration when choosing a CMS. If the system does not match your content, you will be forced to make customizations to the CMS to ensure it fits your organizational needs. Some level of customization may be (will most likely be) required; however, choosing an appropriate CMS from the beginning should always minimize this.

Should Have a Strong Customer Base and Stable Performance Track Record

Does the CMS software vendor have several clients similar to your organization, with a proven history of successfully meeting their content needs? Will the vendor allow you to contact some of their current clients for feedback regarding their product? A quality CMS vendor should be able to provide you with a résumé of sorts of its accomplishments. The vendor should participate in trade shows, industry special-interest groups, and best-practice organizations. Investigate all the product offerings to find the best match for your organization's needs and future needs. See whether other CMS vendors are attempting to play catch-up with this CMS vendor. Look for awards and achievements relating to the CMS product. Also look for quality support and a good history of deploying this CMS solution to various clients. You do not want to purchase a CMS from a fly-by-night company that will leave you with a solution that cannot be upgraded, that will not have service packs available, and that will not be able to meet a support contract.

Should Match Your Budget and Future Growth

Does this CMS have the capability to add storage, are patches and service packs released regularly, how scalable is this CMS architecture, and does this CMS allow usage by your European offices? A CMS is an expensive investment for an organization. You may be rolling out the system only to your United States offices now, but what about in five years when you expand into the European market? How much will your content grow in five years when all departments in your organization are using it? Balancing your current technology budget with your future forecasted growth is a difficult decision but is one you must take seriously. The CMS you choose must allow for expansion.

Investigate the Features and Functionality That This CMS Cannot Provide for Your Organization

If the vendor cannot identify some of its product's weaknesses when addressing your needs, then perhaps the vendor does not fully understand your content needs. CMS is not a magic bullet; do not be too wooed by slick marketing and salespeople hype to forget that the CMS will not do some tasks. Specifically, will you need to customize the product, hire specialized and technical support and development staff, or rely on third-party applications for your content? Is this CMS vendor aware of its product's weaknesses, and does the vendor have a plan, experience, and suggestions on how to defeat those weaknesses?

Your organizational needs and budget will determine the type of CMS it needs; however, most companies will benefit from some level of content management. Implemented properly, a CMS is one of the smartest purchases that an organization can make. Using this chapter as a guide, you will be well informed to make recommendations on that purchase and assist with the implementation.

Summary

In this chapter, we discussed content and what content is. We then described content management and let you know the business need for content management. You learned how CMSs separate the data and the presentation to ensure content reuse. Next we discussed metadata and how critical metadata is for content intelligence. We then discussed templates that allow for content creation and generation and for workflows that are system-based models of your business process for the content creation life cycle. We talked about user groups and security permissions in a CMS and repositories to store your content and content generation. Finally, you learned about the most critical attributes of an enterprise-class CMS.

In the next chapter, we will introduce you to the CMS case study that the remainder of this book will be based on, and we will jump into the first task, namely, defining the scope of the case study for FiCorp.