



THE CMS REPORT

Web Content Management Products & Practices



Tony Byrne



Spring, 2003

SAMPLE VERSION

4th Edition

Background & Acknowledgements

Copyrights

This publication may not be copied, reproduced, or retransmitted in any form or by any means, *including storage and retrieval systems*, without prior written consent. You have received a single-use license only. *You may print and read this document yourself, but not share it with others, including others within your firm, agency, or enterprise.* For reproduction, distribution or site licensing (including Intranet) rights, please contact licenses@cmswatch.com.

CMSWatch and the CMSWatch logo are trademarks of CMSWorks, Inc. All other company names, product names, and service marks are used for identification purposes only and are trademarks of their respective owners.

All screenshots came from and belong to product vendors themselves. All other graphics are copywritten by CMSWorks, Inc. and may not be copied, reproduced, or reused without express written permission.

About the Author

Tony Byrne is Founder and Managing Editor of CMSWatch, and President of CMSWorks, Inc., a Web Content Management consulting and training firm. A former reporter, publisher, international educator, and 12-year Internet veteran, Byrne previously headed the Engineering and Production groups at IDEV, an Internet consulting and development firm. In his consulting practice, Byrne specializes in vendor-neutral RFP development and product selection. He is a frequent trainer and speaker on the subject of Web Content Management Systems.

About CMSWorks, Inc.

CMSWorks, Inc. is a Content Management consulting and training firm that helps companies develop Web Content Management strategies, including: audits, onsite training in WCM concepts and best practices, requirements definition, and WCM package selection. For more information, please contact CMSWorks at info@cmswatch.com, or www.cmswatch.com.

The author and CMSWorks, Inc. disclaim all warranties as to the accuracy, completeness, or adequacy of the information in this report and shall have no liability for errors, omissions, or inadequacies in the information presented.

The Complete Report

This is only a sample of the complete, 176-page report. The complete report may be purchased for US\$ 895-online at <http://www.cmswatch.com>.

Table of Contents

<i>Background & Acknowledgements</i>	2
<i>Table of Contents</i>	3
I. CONTENT, MANAGEMENT, AND SYSTEMS	6
<i>The Content Mandate: More, Better, Faster, Cheaper</i>	6
Larger and more complex websites	6
Content and e-commerce	6
More demands	7
Publish faster	7
Publish cheaper	8
<i>Do You Have a Content Management Problem?</i>	8
15 Symptoms of a WCM problem.....	9
CMS to the rescue	9
<i>What is a Content Management System (and what is it not)?</i>	10
<i>The Business Case: How to Justify a WCM Investment</i>	11
Hard benefits	11
Soft benefits.....	13
Which rationale?.....	14
When all else fails	15
II. CMS DISSECTED	16
<i>Information Management Alphabet Soup: DM vs CM vs KM vs AM vs SCM vs DRM</i>	16
Which kind of software do you really need?	16
Digital Asset Management (DAM)	16
Document and Records Management (DM and RM)	17
Knowledge Management (KM).....	18
Software Configuration Management (SCM).....	18
Digital Rights Management (DRM)	19
Content Management: “ECM” and “WCM”	20
<i>The Web Content Lifecycle</i>	24
PHASE 1: PRODUCTION	24
Role Management: Who gets to do what.....	25
Authoring and Transformation: How content gets into the system	26
Aggregation: Putting disparate pieces together	32
Tagging: Adding value to content through meaning and relationships	34
Workflow: Editing, revision, and approval	36
Library Services: Versioning and Version Control	39
Localization: Supporting multinational web production	40
Content & Code Promotion: Managing the steps to go-live.....	42
What a WCM product won’t do	43
PHASE 2: DELIVERY	44
Page Generation: Where it all comes together.....	44
Searching: How users really find content.....	46
Personalization: The 1-to-1 Web?	47
Privileges Management: Controlling who gets to see what.....	49
Caching & Replication: Speed and reliability matter	50
Other Formats: Wireless, E-mail, Print, CD-ROM.....	52
Syndication: Extending the reach of your content.....	54
Vending: Selling online	56

What a WCM product won't do:	57
III. WCM PRODUCT COMPARISONS	58
<i>Introduction and Classification</i>	58
Product Categories	58
Product Roots	58
Lifecycle Focus	59
Putting It All Together	60
Overview of the Landscape: Confusion and Turmoil	60
<i>Major Enterprise Content Management Platforms</i>	63
Vignette: V7 Content Management Suite	66
Divine - FatWire: Content Server 5.0	70
Documentum: Documentum 5 Platform, (4i WCM Edition)	73
Interwoven: TeamSite 5.5	77
Stellent: Content Management	82
<i>Upper Tier Companies</i>	85
Gauss: VIP 8 ContentManager	86
FatWire: UpdateEngine6	89
Percussion: Rhythmyx 5.0	92
FileNET WCM	96
Day: Communiqué 3.5	98
Microsoft: Content Management Server 2002	101
Tridion: R5	105
MediaSurface: MediaSurface 4.5	108
<i>Delivery-Oriented Application Servers and Portals</i>	110
ATG: Dynamo e-Business Platform	113
BEA: WebLogic E-Business Platform	115
Plumtree: Corporate Portal	117
<i>Departmental CMS Packages</i>	119
PaperThin: CommonSpot ContentServer 3.2	120
Merant: Collage 4.0	123
Ingeniux: Content Management System 3.0	127
RedDot: Content Management Server	130
<i>Low-Priced Products</i>	134
Ektron: CMS300 2.5	135
<i>Open-Source Alternatives</i>	137
Midgard: v 1.4	139
Zope: Content Management Framework (CMF)	141
<i>ASP Services</i>	143
Atomz: Publish	144
CrownPeak: Advantage CMS	147
<i>Do it yourself?</i>	150
IV. ADVICE, PITFALLS, AND BEST PRACTICES	151
<i>How to select a WCM package (and remake your WCM system).</i>	151
<i>Vendor Pricing</i>	156
Software	156
Professional Services and Support	157
Some Final Words on Pricing	158
<i>Twelve Common Pitfalls to Avoid (and Best Practices to Follow)</i>	158

<i>Trends in Web Content Management</i>	<i>163</i>
Do it Right, Instead of "Just Do It."	163
XML will continue to exert a centrifugal force	163
Content production and publishing will continue to separate	163
Buyers will see intensified competition in the mid-market.	163
Web Services will take WCM outside the corporate box.	164
A Brief Conclusion	165
 V. APPENDICES.....	166
<i>Appendix 1: Glossary</i>	<i>166</i>
<i>Appendix 2: Information Management Software Product Landscape.....</i>	<i>171</i>
<i>Appendix 3: XML and CMS.....</i>	<i>172</i>
Why XML for CMS?.....	172
XML and WCM Products.....	173
XML Storage.....	175

[NOTE: FOR THIS SAMPLE VERSION, WE HAVE ONLY INCLUDED AN ABBREVIATED VERSION OF THE FOLLOWING SECTION]

Do You Have a Content Management Problem?



You may sense that you have a WCM problem, but how can you be sure? The following “health check” can help. Use the checklist on the next page to tick off each potential symptom you presently experience on your firm’s website.

15 Symptoms of a WCM problem

- ☐ Content bottlenecks with a webmaster, IT department, or some other gatekeeper.
- ☐ Site visitors have difficulty finding what they need.
- ☐ Content contributors have difficulty finding what they need.
- ☐ Some content is inaccurate / outdated / redundant / unauthorized.
- ☐ The home page does not provide a full, up-to-date portal into the rest of the site.
- ☐ The website exhibits inconsistent design and navigation schemes.
- ☐ Contributors occasionally overwrite content / files accidentally.
- ☐ Web managers need to “roll back” the site to a previous version -- perhaps for legal or regulatory reasons -- but cannot.
- ☐ Content contributors are unable to pre-publish content to appear at a specified later date or time.
- ☐ Website managers cannot associate the company’s products and services to articles or news on the site (or vice-versa).
- ☐ Content has feet of clay: web managers cannot easily reuse / share / distribute / import it.
- ☐ An inability to protect or control access to content keeps good material offline.
- ☐ Marketing and product managers cannot customize content for customers, partners, and other important visitors.
- ☐ Internal company staff is not invested in web communications.
- ☐ Company staff lament, “Our website is not as good as we are.”

How did you score? Add up the number of checks and consult this chart.

# checked	Prognosis
0-3	Congratulations. You reside among the minority of organizations successfully addressing online content challenges. You may still benefit from some niche technologies for syndication, versioning, searching, or other problems you face.
4-6	Your WCM system is a little creaky. Model your existing processes and workflow from content inception to publishing (see “Lifecycle” in the following section) and look for opportunities to improve the way you are managing content. Certain problems you face may have little to do with content management – for example you may suffer from inherently poor graphical design or navigation – but applied technologies could help you substantially in other areas, like content contribution.
7-12	You have a bona fide content management problem. Read this report and start creating a business case for rebuilding your content management system.
13-15	Deficient content management is seriously hindering you from doing business online. If content is critical to your business, and the Internet is an important part of your company’s overall strategy, then you cannot act too soon.

You might want to engage others in your organization to take the same health check to build support for action. Find an interactive version online at <http://www.cmswatch.com/healthcheck>.

[NOTE: FOR THIS SAMPLE VERSION, WE HAVE ONLY INCLUDED AN ABBREVIATED VERSION OF THE FOLLOWING SECTION]

The Web Content Lifecycle

Vendor software categories and your particular requirements and can both be understood in relation to the “Web Content Lifecycle.”

Web content is managed in two phases:

- 1) Production, where content goes “from thought to click.”
- 2) Delivery, where content actually gets consumed by end-users.

Some vendors who focus solely on the first phase of the lifecycle equate Content Production with Content Management. Indeed, you may join a growing number of customers who license separate Production and Delivery products. However, we approach Web Content Management from a holistic perspective, incorporating both the Production and Delivery phases.

Both phases contain specific attributes that need to be addressed in any WCM plan. By understanding your specific needs and opportunities in each phase, you can begin to construct the outlines of a new WCM system. You will also have a roadmap for evaluating the suitability of various package offerings.

In the pages to follow, we will identify the key system attributes for both phases of the Content Lifecycle that you should consider when developing your requirements.

Production Phase	Delivery Phase
Role Management	Page Generation
Authoring & Transformation	Index & Searching
Aggregation	Personalization
Tagging	Privilege Management
Workflow	Caching & Replication
Library Services	Other Formats
Localization Support	Syndication
Promotion Path	Vending

Each attribute will be accompanied by a “Requirements Builder” questionnaire to help you focus your analysis. Don’t worry if you can’t readily answer all the questions. The job of a good consultant or attentive vendor is to help you explore options and weigh trade-offs.

At the end of each of both phases we will also identify what a WCM product *won’t* do. There are some things that you can’t – or shouldn’t – automate.

If you already have a good understanding of the concepts in the chart above, you can consider skipping this section and thumbing straight to the product surveys. *Note, however, that the product section of the report makes heavy use of the concepts and terminology discussed below.*

PHASE 1: PRODUCTION

CM vendors sometimes call this the content “creation” phase, or simply “workflow,” although the latter is really only a subset of content production. Other vendors that focus on this phase use the all-encompassing label of “content management” to describe it, though we consider WCM to encompass both Production and Delivery.

We define Production as everything that happens to content before the end-user sees it.

Although first in chronology, this phase is typically last in line to be automated by web teams. Indeed, most companies tend to focus first on delivery systems – be it through portals, application servers, or other forms of dynamic publishing. Yet, the Production phase is where many substantial cost, control, and efficiency gains can be realized.

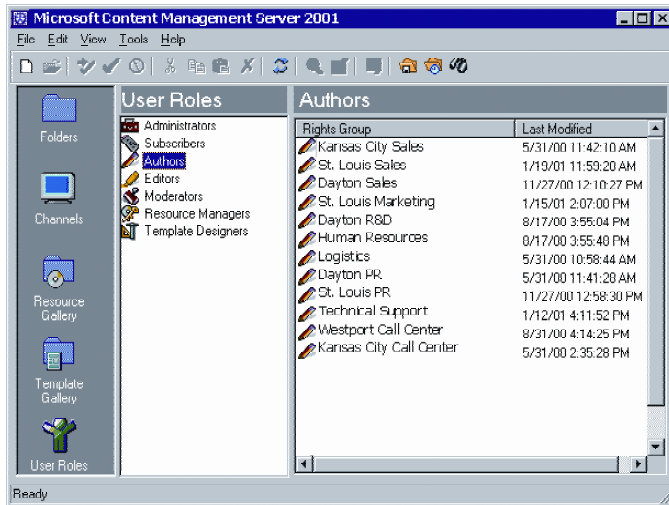
Role Management: Who gets to do what

Many people can be involved in the production of even a departmental website. And since one of the key advantages of WCM is to distribute content maintenance capabilities directly to content

owners, implementing a new CMS leads to even more people becoming involved in this process.

Therefore you need a system to manage internal access and privileges (access and privileges for users outside the corporation are addressed later) that is much more robust than that required to support only one or two webmasters updating an entire site.

Users can be assigned privileges based on the *role* they play (the types of things they can do), or *group* to which they belong, which circumscribes their authority – and typically the scope of the content areas they can edit. It is possible (and indeed quite common) for a user to make up a group of one.



Consider the following chart of Groups and Roles for a generic corporation.

Sample Groups	Sample Roles
HR Managers – can only add and modify jobs	Superuser – performs any function in the system
Product Managers – can update catalog content only	Editor – approves content
Graphic Designers – can create and modify image files and HTML templates sitewide	Author – contributes content
System Administrator – can update scripts and change file permissions sitewide	Intern – adds metadata to site content

You may need a system that lets you assign a combination of Groups and Roles. Using the chart above, let's say that Nancy serves as an Editor in the Project Manager Group. She approves the additions and updates that Authors have made to catalog content.

Most CMS packages will tie into existing corporate directory systems (such as LDAP servers), but note that the way they do so will vary markedly among competing products. For example, some products will access an LDAP repository in real-time, whereas others require that the LDAP server synch up with the product's own access control lists on some sort of regular schedule. In the latter case, there can be periods where a user whose rights have been expunged in the corporate repository may still have access to CMS privileges.

Role Management Requirements Builder:

What level of granularity do you need now, and anticipate needing in the future?

Some packages proscribe certain roles or limit the number of groups you can define. In the screenshot above, you can see that MS CMS 2001 comes with seven roles “out of the box.” You can theoretically extend these to create new roles, but the level of effort and potential for problems at upgrade time might make it more worthwhile to simply select a package that could support the number of different roles you need from the outset.

Do you prefer a particular authentication mechanism?

Internal users can be authenticated by the operating system, the webserver software, through internal application logic, or even by a content database. Each has its advantages and disadvantages, but nearly universal experience reveals that you will trade off potency of security with ease of use and maintenance. Don’t underestimate maintenance overhead here.

Do you require or prefer a certain operating system?

Obviously, this is a global consideration, so let’s address it right up front. If the WCM system uses NT permissions to manage roles and authentication, then you are not going to be able to run it on the new Sun Solaris server you just bought. Operating system choices present the first of many chicken-or-egg predicaments in selecting a WCM package. In an ideal world, you would be free to select the most suitable product regardless of any platform limitations; in the real world however, the IT team that will have to support the application going forward may be much better versed in one operating system over another.

Do you need to integrate with other role management systems?

If your users can access the WCM system through an existing corporate log-on they will be more inclined to participate actively. If you use LDAP or Microsoft’s Active Directory for corporate directory services, look for a WCM package that supports those protocols. Perhaps you are using your corporate Portal to support single sign-on across the enterprise; clearly you’d want your CMS package to tap into those permissions.

III. WCM PRODUCT COMPARISONS

[NOTE: FOR THIS SAMPLE VERSION, WE HAVE ONLY INCLUDED AN ABBREVIATED VERSION OF THE FOLLOWING SECTION]

Introduction and Classification

Product Categories

This section provides an overview of the web CM product landscape along with surveys of 22 representative products in 7 distinct categories:

- 1) Enterprise Platforms
- 2) Upper-tier Packages
- 3) Publishing-Oriented Portals and Application Servers
- 4) Departmental / Mid-market Products
- 5) Low-Cost Products
- 6) Open-Source Packages
- 7) ASPs

Each category is described in detail in the following pages, with representative product surveys. *CMSWatch selected these products indicative of the type of offerings available in their class.* They were not chosen because we believe they are inherently stronger, weaker, or more popular than others in their same group.

As an introduction, it is also meaningful to divide products according to:

- 1) their roots, and
- 2) how they address the WCM lifecycle.

Product Roots

Understanding the origins of different packages enables you to see deeper into their relative strengths and weaknesses. We divide the WCM universe into 3 different ancestries:

- Pure-play Web Content Management Packages. These currently predominate in the marketplace. Many of them are newer companies.
- Application Servers and Enterprise Portals. **Section 4** of the product reviews below addresses this class directly.
- Document Management / Workflow Origins. We take a closer look at these in the following paragraphs.

Putting It All Together

Here is a telegraphic categorization of the packages mentioned in this report (along with a few others, for comparison purposes). These products and their categories are reviewed in greater detail in the pages that follow.

Category	Production-oriented	Delivery-oriented	Full-Cycle
Enterprise	Interwoven	Oracle 9i BEA IBM (WebSphere)	Vignette Documentum divine Stellent Broadvision
Upper-Tier	FileNET Percussion Tridion	Plumtree ATG Hummingbird	MediaSurface Gauss FatWire Microsoft CMS Day
Departmental	RedDot Atomz CrownPeak	MS Sharepoint	Merant Obtree PaperThin Dynabase/Engenda Roxen Ingeniux
Low-cost	PureEdge	UserLand	Ektron Infosquare Clickability
Open Source		Midgard Cocoon / AxKit	Zope RedHat (ACS) OpenCMS

KEY:

Pure-Play WCM Products
 DM / Workflow Roots
 Application Server Portals
 Pure-Play Portal Packages
 ASP ("Hosted") Solutions

Note that *there is no "Magic Quadrant" here*. The package that makes sense for you depends on your budget and your requirements.

Overview of the Landscape: Confusion and Turmoil

In your search for a WCM package, you will find an increasingly confusing array of technology players. At last count, more than 225 software vendors have assigned a "content management" label to their products. As the web content expansion hit critical velocity in late 1998, large and small firms alike sensed the vacuum described in Section I of this report and leapt into the fray. Some large firms, like Broadvision and Macromedia, didn't want to risk a hole in their product line and acquired WCM technologies from other firms (Macromedia then cancelled the CMS package it inherited, called "Spectra").

Other companies converted one-time applications developed in-house or as a result of consulting work into full-blown commercial products. This trend continues even into 2002, as integrators seek to bring their own offerings to market as semi-packaged software.

And now we see a whole raft of products that *call* themselves CMS platforms but only address a small piece of the WCM puzzle, from web development packages to catalog management tools, to network caching devices, and everything in-between.

It is important to recognize – and many analysts have already pointed out – that WCM is a fairly immature space. Antecedent products that addressed Document or Asset Management may have bequeathed some key concepts like Workflow and Library Services, but the WCM problem domain remains fluid enough that true product categories are only now emerging. In nearly every case, the products themselves are only a few years old, and even then, many have been completely rewritten from scratch during the interim.

The nearly ubiquitous marketing jargon that obscures true product niches compounds the dilemma for would-be WCM buyers. Hopefully, this section of the report will provide you with an accurate roadmap of the current WCM product landscape.

The WCM arena has also seen substantial mergers-and-acquisition (M&A) activity, as well as outright corporate death. Longtime niche vendor eBT closed its doors in 2001, ending support for its Dynabase and engenda product lines (though they were picked up and enhanced by a systems integrator). Similarly, Astoria closed up shop in 2002. And it is not just the little firms that are getting out. Major analyst shops make a living comparing the financial strength of WCM providers, but even successful firms have made difficult decisions to cancel products; for example, Macromedia decided to discontinue “Spectra” shortly after its acquisition of Allaire, despite Spectra’s ample market-share.

In general, however, there has been less consolidation than most analysts have been predicting repeatedly since 2000. But don’t be surprised if other major players decide that the WCM market is too competitive and elect to focus on other core technologies.

All this confusion and turmoil has led to more than a little fear and angst in the marketplace, especially as war stories proliferate about stupendously difficult or even failed WCM implementations. So it is not surprising that would-be WCM buyers often experience extreme paralysis, despite the evident pain they endure managing their existing web publishing efforts.

But if you believe that the effectiveness with which you manage the production, publishing, and distribution of your web content will be critical to your company’s future competitive position, then you need to get off the dime. Just go in with your eyes wide open, armed with clear requirements and best practices.

Matching Features to Requirements

To help you compare and evaluate products based on your requirements, we will use the following charts from the discussion above. The first (at left) is an analysis of product features by lifecycle phase, with accompanying explanatory key (below, right). The final chart (following page) outlines general specifications for a package.

Phase / Attribute	✓
PRODUCTION PHASE	
Role Management	✓
Authoring & Transformation	✓
Aggregation	✓
Tagging	✓
Workflow	✓
Library Services	✓
Localization Support	✓
Promotion Path	✓
DELIVERY PHASE	
Page Generation	✓
Index & Searching	✓
Personalization	✓
Privilege Management	✓
Caching & Replication	✓
Other Formats	✓
Syndication	✓
Vending	✓

✓	Key
✓	Product provides this feature
✓+	Product excels at this feature, <i>relative to other products in the same category</i>
✓-	Product provides this feature, but is not as mature as its rivals
o	Feature provided through an O ptional module or product
p	Feature offered through a P artnership
—	Product does not provide this feature in any meaningful way.

Specsheet	Product Summary
Integration	What other systems does the product formally integrate with?
Partnerships	What other notable players can be added to the solution?
O/S & Databases	What Operating Systems does the product run on? Which databases will it use?
Client	How do content contributors and administrators access the system?
Standards	What industry standards does the package adhere to?
APIs	What languages can you use to integrate and customize the system?
Reporting	What canned reports are provided about both internal and external activity?
Licensing	How is the product priced?
Ownership	Is the company publicly or privately held? Market cap at the end of April, 2003 and percentage change from last editions of this report(s)?
Comments	A general summary.

[NOTE: FOR THIS SAMPLE VERSION, WE HAVE ONLY INCLUDED AN ABBREVIATED VERSION OF THE FOLLOWING SECTION]

Upper Tier Companies

Surveyed:

- Gauss VIP 8 ContentManager
- FatWire UpdateEngine6
- Percussion Rhythmyx
- FileNET (eGrail) WCM
- Day Communiqué
- Microsoft CM Server 2002
- Tridion R5

Upper Tier offerings occupy a moderately expanding space in-between departmental installations and full-blown Enterprise licenses. Relative to their financially struggling upmarket competitors, these vendors tend to be performing reasonably well in the marketplace right now. To be sure, each of these platforms can point to some enterprise installations, though in general, their offerings tend to be less-easily integratable with other corporate systems than, say, Vignette's product line.

At average license fees running from US\$100-200,000, their price points tend to be more attractive than the Majors, but their architectures can nonetheless support reasonably scalable projects. Note, however, that these products are typically not as "out of the box" as the mid-market offerings at a tier below. Plan on integration costs ranging from 1 - 2X your licensing cost.

Not reviewed but worth noting:

- XyEnterprise Content@ (www.xyvision.com). This is a small company, but they've been dealing with content management of one kind or another since the dawn of the Web, working towards the holy grail of "single sourced" content and document management: write once, publish anywhere. These folks really understand mark-up languages. Consider them primarily for technical publishing needs.
- CoreMedia – Content Application Platform (www.coremedia.com). This is a Java-based, tightly-coupled Production and Delivery package that dares to turn the clock back on two fronts: it takes a more platform orientation (i.e. less "out of the box") and a thick (Java-based) editorial client that connects to the server real-time. Founded in Germany, CoreMedia may have incubated there too long before branching out to the UK. Still, it will be interesting to watch this company evolve.
- Enigma Insight (<http://www.enigma.com>). Enigma used to be a somewhat sleepy DM company, known for search engine technologies and CD-ROM delivery tools. Some of its largest clients were manufacturers, and as underlying DM technologies migrated to those company's extranets over the past 3 years, Enigma discovered a niche as a "support chain enabler," helping companies manage aftermarket customer support, services and parts sales. This is a smart place for Enigma to be right now, but unless you are looking to manage a manufacturing extranet exclusively (as opposed to a more general corporate web publishing effort), then this product is probably not a good fit for you.

Percussion: Rhythmyx 5.0

<http://www.percussion.com>

Founded in 1994, Percussion began life developing add-on tools for Lotus Notes and Domino, centered principally on workflow and integration. The company developed Rhythmyx in 1999 in an effort to break into a broader market beyond Lotus installations, using newer technologies such as XML, XSL, and Java.

The product is modeled on an “element” view of WCM, as opposed to a page-centric approach, and is therefore best suited for environments where content tends to be highly structured. Workflow, assets, and text are all defined and managed on element level. This level of granularity offers a powerful and extremely flexible approach, with a very separation of content and presentation. You may find it overkill for simpler implementations.

XML abstraction plays a central role in how the product works. Although Rhythmyx uses a database to store content, the elements are represented in the system as XML. All business logic is also stored and manipulated as XML behind the scenes. This brooks some extra overhead, but that may not matter much since the system is essentially a “baking” mechanism that serves content at approval time rather than run-time. There are also advantages to relational database storage, including improved search and retrieval, and easier connectivity to existing corporate RDBMS repositories.

Rhythmyx uses XSL to transform the XML data into useful, consumable content. With this approach, you can quickly build different views into the same content objects, as opposed to permanently “assigning” information to a single template the way most mid-market WCM products do. Percussion asserts that using XSL to assemble and disassemble content elements means that you don’t have to commit to a particular scripting or programming language like Perl or Java; this is

Phase / Attribute	✓
PRODUCTION PHASE	
Role Management	✓
Authoring & Transformation	✓
Aggregation	✓-
Tagging	✓
Workflow	✓-
Library Services	✓
Localization Support	-
Promotion Path	✓+
DELIVERY PHASE	
Page Generation	✓
Index & Searching	p
Personalization	p
Privilege Management	-
Caching & Replication	✓
Other Formats	✓
Syndication	p
Vending	-

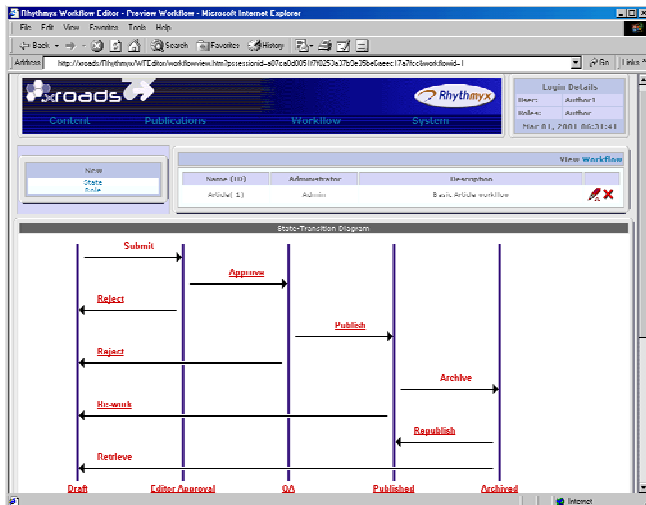
true, and XSLT is an open standard. On the other hand, XSL may be a new skill in your IT group, and its learning curve can be steep. Percussion provides a built-in tool that automatically creates XSL from a specially-tagged HTML template document. This works for most insert-data-here type templates, but to create smarter templates with logic or non-HTML output (i.e. WML), you’ll need to delve into the XSLT.

Version 5 sports an all-new interface (see screen, left), run completely as a Java applet. This new UI was a response to criticisms that the product’s previous control panels were too



client-server like. Under the new format, you can organize content according to a variety of different views, using a folder metaphor to group objects with similar metadata attributes. In another bow to end-user needs, Rhythmyx offers in-context template and content editing, to allow editors to mix and match elements in a WYSIWYG environment, as well as list element dependencies.

Rhythmyx is a purely object-oriented approach, which allows for substantial flexibility in the hands of an experienced architect with broad leeway for system design. For example, different workflows and stylesheets can be applied based on an element's metadata (most systems "glue" these parameters to a content object or area of the site).



Rhythmyx applies a state-transition methodology to workflow (see screen, left). Under this approach content always exists within a given status. In that state, people with suitable privileges can do things to it, before it is transitioned to its next state. Many architects find a state- transition approach more intuitive for workflow modeling, although it doesn't handle branching (parallel workflows) very cleanly, and end-users typically don't find it as easy to visually validate.

One workflow issue that some Percussion clients have raised is the difficulty Rhythmyx can have when the same content object needs to reside in two states simultaneously, or branch into two versions, one succeeding the other upon approval. This might happen, for example, when an article is

updated, and the newer version is still traveling through the workflow (as yet unapproved), but you don't want the existing version removed from the production site. The package may be better suited for news and other similar sites where content tends to accumulate in batches, but then is not repeatedly edited and re-edited thereafter.

Like other WCM vendors, Percussion bundles a visual editor to enable business managers to make changes to workflows. They can do this in real-time, with changes applied to all content in affected states. However, there is no visual cue to show an editor where any particular object they are reviewing resides in the workflow (the system just tells you what state it is in at that time).

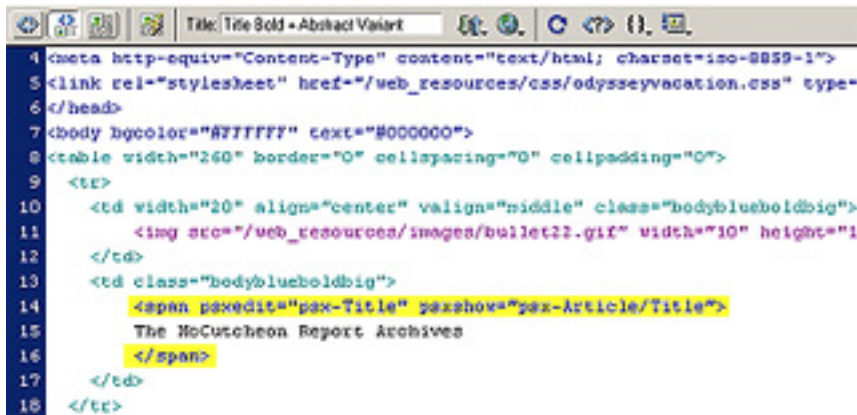
Version 5 Rhythmyx introduces some real reporting (see screen, right). For example, editors can run an "impact analysis" of dependencies among content objects (FileNET does this too) and shows what the implications are for making a change to any discrete object. For example, if I change an image, what are the implications on all the pages where that image appears? It can also show what workflow tasks might get triggered when changing some "placeless" text. This is particularly important for sites requiring

Name	Checked Out	State	Content Type	Author
Syndicated Product Catalog		Draft	Document	John Lee
Tax Planning from Enterprise Investments		Draft	Document	Eva Perez
Enterprise Investments Funds		Draft	Document	Eva Perez
Bush plan - Who benefits, by how much		Public	Article	Paul Baker
Open and Closed Mortgages		Public	Article	Paul Baker
The gre...		Draft	Article	Eva Perez
Taking A...		Review	Article	Eva Perez
Woman		Public	Image	Paul Baker
Image fr...		Draft	Image	eva perez
Fixed-rat...		Public	Product	Paul Baker

translation. For example, on a Canadian government site, the English version of some content cannot one go live until the corresponding French version is also approved.

Percussion also brings a new twist with its notion of “active assembly.” In addition to enabling authors to move elements around on a template, Rhythmyx enables you to develop specific “variants” ahead of time for particular elements. For example if you want to make a minor change to the way a bulleted list of links appears, you can easily modify the element so that some or all the links also show an associated abstract. This avoids some of the “straitjacketing” that other CMS products with more rigid templating models can impose on editors. Of course, it also implies substantial forethought in your content models. You can’t publish abstracts unless someone thought to add an “abstracts” field to the repository, and authors have actually filled it in with content. The flexibility of variants presupposes that a sufficiently granular chunking of content has transpired. Contrast this approach with FatWire and others, who let non-technical administrators create new content elements without physically updating the database, but who in turn make it a bit more difficult to revise templates accordingly.

We quibble a bit with Percussion’s assertion that these variants can be set and modified by non-



```

1 <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
2 <link rel="stylesheet" href="/web_resources/css/odysseyvacation.css" type="
3 </head>
4 <body bgcolor="#FFFFFF" text="#000000">
5 <table width="260" border="0" cellpadding="0" cellspacing="0">
6 <tr>
7 <td width="20" align="center" valign="middle" class="bodyblueboldbig">
8 
10 <td class="bodyblueboldbig">
11 <span paxedit="pax-Title" paxshow="pax-Article/Title">
12 The McCutcheon Report Archives
13 </span>
14 </td>
15 </tr>
16 </table>

```

technical editors. You identify editable elements by using Rhythmyx -specific `<span>` tags in HTML (see screen, left); this is a nice approach that should be much more portable than other vendors’ placeholders. Then you need to map content from the database to your XML format using a visual tool that Rhythmyx provides (see screen shot in Appendix B). Then you set some configurations in another browser interface, with the XSL code for generating output helpfully

generated for you. So does this require a developer? Not necessarily, but your template designer or managing editor needs to know basic HTML and have a high comfort level with datamodels and technically-oriented wizard interfaces.

Also, note that the templates themselves live on the file system outside of the CMS. That means you need to version and workflow them separately, using 3rd-party tools. More generally, you’ll want to consider carefully whether this product will scale successfully if you have multiple large sites and want to share infrastructure and repositories in a services-oriented architecture. Out of the box, the product is really designed to support one site per physical repository. There are ways to work around this using the Percussion’s notion of “content communities” in the product, but this is not the same as serving multiple large sites – separated logically – within one repository.

On the delivery side, Rhythmyx can publish content to web servers (the most common configuration), other content databases, or to application servers, via publication “indexes.” Indexes are similar to the concept of “editions” in other WCM systems, such as “staging,” “Extranet,” “production,” and so forth.

In short, Rhythmyx “bakes” content to the delivery environment, therefore doing the expensive XSL transformation at design-time (rather than run-time). In this respect, it is somewhat similar to Interwoven – except that Percussion:

- Is cheaper (and doesn’t require client licensing)
- Uses a database to store your content
- Is end-to-end XML enabled—all content can be repurposed in other formats using XSLT stylesheets

- Is perceived by most end-users to have less attractive user interfaces out of the box. Your mileage will vary.
- Is generally less feature-rich, particularly in the areas of editorial review, scalability, load-balancing, and content delivery
- Enjoys fewer direct partnerships with Delivery software vendors

With its new version, Rhythmyx makes the beginnings of a play for the “ECM” market, adding optional document management features and connectors to the MediaBin (DAM) and Kofax (imaging) packages. The DM functions are a relatively short extension of the product’s existing XML assembly capabilities (except now you call the results “documents,” instead of “pages”). It does not offer any special features for handling binary documents. Therefore, at an extra \$50,000 in additional license fees (on average), we are not yet sanguine about the value proposition for Percussion’s ECM version. Nevertheless, the WCM version, which runs from \$85,000-150,000, handles basic Web content management quite capably.

Specsheet	Percussion Rhythmyx Summary
Integration	BEA, IBM
Partnerships	Kinecta, Epicentric, ArborText
O/S & Databases	Windows NT/2000; Solaris. Oracle, MS SQL Server. IBM DB2
Client	Browser. Developer “workbench” runs on Microsoft desktops only.
Standards	XML, ICE, SOAP
APIs	Java, XSL
Reporting	Workflow and element status; other production information is logged.
Licensing	Server-based. Median licenses run \$85-\$125,000.
Ownership	Privately held. No financials available.
Comments	• Solid Production product from a company with deep workflow experience. • Percussion tends to be stronger in engineering than product and interface design; tends to sell more successfully when IT groups heavily involved. • You will probably want to hone your XML and XSL skills before venturing forward with this product. • May be better suited to news sites and portals than corporate brochure sites. • Watch carefully for scalability issues where there are sizable numbers of concurrent contributors. • Company may be a possible take-over target. • Compare to Interwoven, Ingeniux, Tridion.

[FOR 24 OTHER SIMILAR PRODUCT SURVEYS, SEE THE COMPLETE REPORT, AVAILABLE AT <http://www.cmswatch.com>]

IV. ADVICE, PITFALLS, AND BEST PRACTICES

[NOTE: FOR THIS SAMPLE VERSION, WE HAVE ONLY INCLUDED AN ABBREVIATED VERSION OF THE FOLLOWING SECTION]

Twelve Common Pitfalls to Avoid (and Best Practices to Follow)

Throughout this process, you can take advantage of the lessons learned by those who went before you. Having led several WCM implementations and after debriefing dozens of practitioners, CMSWatch has seen much go wrong – and much go right. Here is a distillation of common pitfalls, along with best practices to follow to improve your likelihood of success.

Pitfall #1	Best Practice
Selecting a WCM package before developing solid requirements and business case.	Gather thorough requirements -- but only after establishing a business case to shape and discipline the process -- <i>then</i> select a package that truly meets your needs.

Talk to ten software integrators and nine of them will tell you that the biggest single predictor of a failed implementation is when a company chooses a specific package before requirements have been adequately established. The second biggest predictor? A requirements phase that drags on because no unified business strategy exists against which tough choices can be tested and measured.

When companies select WCM packages before they agree on business objectives and adequately plumb stakeholder needs, the technology inevitably ends up driving the system – the people, the business rules, the editorial processes, even the content itself – rather than other way around. You’ve probably worked amid systems like that, and you probably didn’t like it.

Perhaps more importantly, you will find along the way that you have critical forks in the road with important decisions to make. Do we impel authors to use forms based entry or focus on Word conversion? Do we integrate our catalog content using special connectors, or not? There are no simple answers to such questions, but a clear business rationale will help you make solid choices.

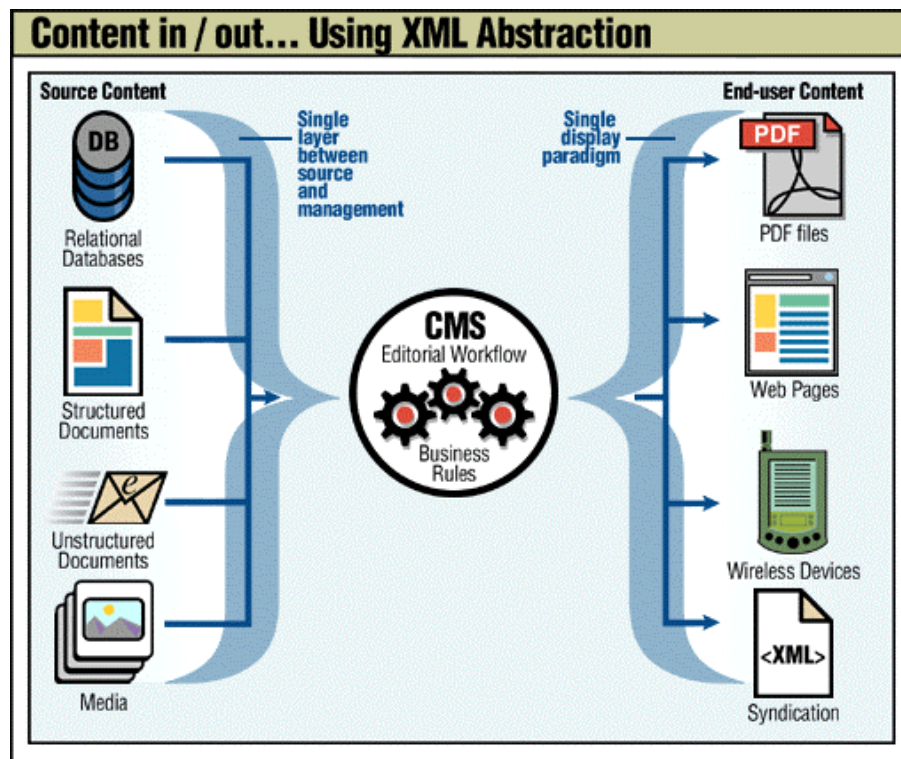
Appendix 3: XML and CMS

[NOTE: FOR THIS SAMPLE VERSION, WE HAVE ONLY INCLUDED AN ABBREVIATED VERSION OF THE FOLLOWING SECTION]

Why XML for CMS?

We argue that XML has a natural place in Web and Enterprise Content Management. Here's why:

- It is a completely open standard based on common syntax, but infinite semantics. That means everyone who uses it needs to follow basic rules (that makes it portable), but you don't have to bend your business to a predefined data model. Your content will probably have its own unique structure – or “semantics” – and XML will extend with you.
- It enables universal data interchange. Disparate systems and enterprises can share content via XML without having to expose internal data models or invest in complex integrations. Therefore, XML is most useful for “data in motion.” Of course, companies are trying to get more value from content precisely by putting it in motion. By the same token, this means that if your content is going to be delivered in only one format (e.g. Web), there is no need to invest in XML – it will add little or no value.
- The “eXtensible” approach typically enables more granular control and adaptation. The holy grail of content management is separating content from site map (“where it lives”) and content from layout (“what it looks like”). This enables you to repurpose and redeploy the same content to multiple locations, devices, and skins. XML does precisely this: it tells you what the content *is*, not where it resides or how it appears. Databases can accomplish this too, but it is generally easier to update an XML document or schema, and in any case, XML is better suited to hierarchical content structures that you typically find in text documents (as opposed to relational structures that typify catalogs).



- Flexible tagging means more sophisticated searches with tag-aware search engines. XML enables you to more easily assign *meaning* to your content. Search engines that can leverage the tagging in your XML repository, as well as the inherent structure of your content, will generate far superior results compared to simple keyword queries. Users can search within particular nodes within your content hierarchy, and enjoy more relevant results that have taken advantage of all the metatagging you did (and you do apply metatags, right?).
- XML has become the “Lingua Franca” for aggregating disparate content elements. XML makes it substantially easier for Web publishers to assemble atomic bits of content (in all its varied forms) in an organized way within one site, or indeed, on a single page. There are two sides to this. First, with respect to accessing source content from within a CMS, XML can provide a common unification environment to work within – a single layer between source content and its actual management. Among other useful features, XML can provide a file system-type interface to database data (more about this below). Then, on the output side, XML can provide a sole source – and in fact, a single paradigm -- for generating diverse consumable formats, such as HTML, PDF, WML, and even (with some wrangling) print.

In short, XML can add value across the WCM lifecycle. The following charts describe this in more detail.

Where Can XML Add Value?	✓
PRODUCTION PHASE	
Role Management	✓--
Authoring & Transformation	✓+
Aggregation	✓+
Tagging	✓+
Workflow	✓-
Library Services	✓
Localization Support	✓+
Promotion Path	✓-
DELIVERY PHASE	
Page Generation	✓
Index & Searching	✓+
Personalization	✓
Privilege Management	✓-
Caching & Replication	✓-
Other Formats	✓+
Syndication	✓+
Vending	✓

✓	Key
✓	XML can add value at this step
✓+	XML can add substantial value at this step. You may wish to focus your XML activity here
✓-	XML might add value in specialized situations

XML and WCM Products

[SEE COMPLETE REPORT]