TSW 01-02
September 2001

JISC TechWatch Report: Content Management Systems

Paul Browning[1] & Mike Lowndes[2]

1: Information Strategy Co-ordinator, University of Bristol. E-mail: paul.browning@bristol.ac.uk
2: Web Manager, The Natural History Museum, London. E-mail: mikel@nhm.ac.uk

Table of Contents

List of Figures

List of Tables

Executive Summary

Institutions are struggling to maintain their Web sites. Out of date material, poor control over design and navigation, a lack of authority control and the constriction of the Webmaster bottleneck will be familiar to many in the HE/FE sector.

The pre-millennial Web has been characterised by highly manual approaches to maintenance; the successful and sustainable post-millennial Web will have significant automation. One vehicle by which this can be achieved is the "content management system" (CMS).

In reality a CMS is a concept rather than a product. It is a concept that embraces a set of processes. Institutional needs are often highly individual (reflecting the heterogeneity of their processes and back-end information systems) and so the task of implementing a CMS will inevitably contain a significant bespoke component. There is little argument that the solution will be a mixture of "buy-and-build". The key issue is one of balance - "how much must we build before we get what we need?"

The market place is very crowded and continues to grow. Many analysts are predicting a shake-out in the medium term. Currently, investing in a CMS is potentially more of a risk than for other, more mature categories of information system.

Vendor literature is generally over-hyped and jargon-ridden. Terms like 'personalisation', 'syndication', 'asset management' and 're-purposing' abound. Moreover, the boundaries of the CMS space are blurred. Substantial overlaps exist with document management systems, knowledge management systems, enterprise application integration systems, e-commerce systems and portals. We also contend that there are significant (but as yet not generally recognised) overlaps with intranet groupware and virtual learning environments. Indeed, it may turn out that one institution's 'managed learning environment' is another's CMS.

We detect resistance within traditional campus computing services and MIS departments to the concept of the CMS (and indeed the need for an automated, post-millennial Web). Fundamentally a CMS devolves control over content to the owners of that content (rather than the technician) and then scales without increasing management overheads. The investment can run to a six-figure sum if the perceived market leading solutions are considered, though many more modest (and indeed no-purchase cost) solutions are available.

Senior managers need to be aware of the costs and consequences of not embracing the post-millennial Web. The cost of investing in a CMS can be on a par with procuring a student record system - but which, in terms of efficiency and effectiveness, is likely to give the biggest return on investment? In fact, there is no choice, because both will be essential items. But we suspect, in terms of competitive advantage, it will be the CMS that will be more important in terms of differentiating institutions.

Key words

Authority control, business processes, content re-use, information architecture, institutional web site, multiple authorship.

The Concept

A Content Management System (CMS) is not really a product or a technology. It is a catch-all term that covers a wide set of processes that will underpin the ‘Next Generation’ large-scale web site.

The pervasive nature of the Web means that it has become the preferred vehicle for content delivery. ‘CMS’ should therefore be read as ‘Web Content Management System’. Institutions have no shortage of ‘content’ - be it data, information or knowledge. When the creation and publication of content is well managed then the organisation functions more cost-effectively; it is also likely to lead to better decision making.

Process/Benefit	Prospectus example
Engendering the re-use of information by allowing the ready integration of data from diverse sources.	A Web prospectus page describing a programme might draw together information from such sources as the student record system (for curriculum data), the personnel system (for details of teaching staff) and an imagebank (containing attractive pictures).
Permitting the efficient re-purposing of information.	The same prospectus page might be rendered a PDF (for high quality hard-copy), as plain text (to be sent as an e-mail message) or optimised on screen for a partially-sighted user.
Allowing information maintenance to become devolved but at the same time preserving central control.	The prospectus entry can be devolved to its academic director but, before going live on a pre-assigned date, a member of the marketing department would first check the amended text for factual and stylistic consistency.
Ensuring presentational consistency by separating the design of Web pages from the content they display.	The academic director would be provided with a template to enter the information about the programme ….
De-skilling the task of putting information on the Web.	…. which reduces the task to no more than ‘filling in the boxes’ on a Web form or a word-processor document.
Facilitating good information management practice so that appropriate metadata are captured at the time of content creation or modification.	The relevant prospectus page is ‘stamped’ with the name of the maintainer, the creation or modification date, an expiry date (which would later cause the automatic generation of e-mails to the maintainer of the information at regular intervals before this date) and incorporation of keywords to ensure indexing by search engines.
Permitting some past state of the Web site to be re-created or restored.	The edition of the prospectus from two years ago can be re-constructed.

Table 1: The business benefits of a CMS

The key goal of a CMS is the increased integration and automation of the processes that support efficient and effective Internet delivery. The means by which this is achieved, placed in the context of a university prospectus, are summarised in Table 1.

We limit the scope of this report to those features that might be regarded as ‘mandatory’ for a CMS in the HE/FE sector. This feature set can be expected to change in the near future. There are several related applications that are often included in the feature lists of content management software. We will not cover e-commerce systems, though several large-scale commercial CMSs are essentially e-commerce management packages. We will not discuss post-deployment applications such as Web site personalisation, searching, noticeboards, guestbooks, etc.

The present report therefore concentrates on the content management issues relevant to the ‘outward facing institutional Web’. It is ironic that, because of the growth of intranets, the outward facing Web will become a quantitatively minor part of the institutional Web (Fig.1). However, in the context of current challenges and risks, the outward facing Web will continue to exercise the minds of senior managers in the medium term.

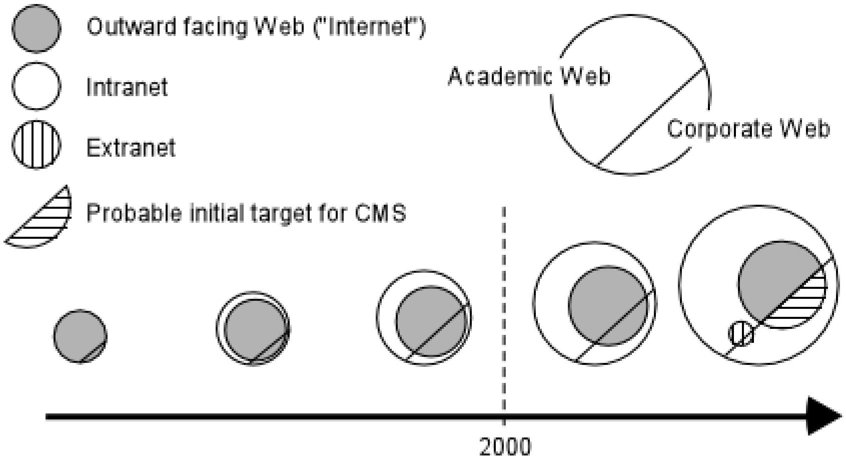
Outward facing Web ("Internet")
Intranet
Extranet
Probable initial target for CMS
Academic Web
Corporate Web
2000

Figure 1: The evolution of the Institutional Web

Within this scope, we have placed the functions of a CMS into four categories: **Authoring, Workflow, Storage** and **Publishing** (Fig. 2). A CMS manages the path from authoring through to publishing using a scheme of workflow and by providing a system for content storage and integration.

Authoring is the process by which many users can create Web content within a managed and authorised environment, whether it be a simple line of text (e.g. ‘The University administrative offices will be closed next Monday’) on a ‘What’s New?’ page, an entry in an online course discussion group, or the entire Postgraduate Prospectus.

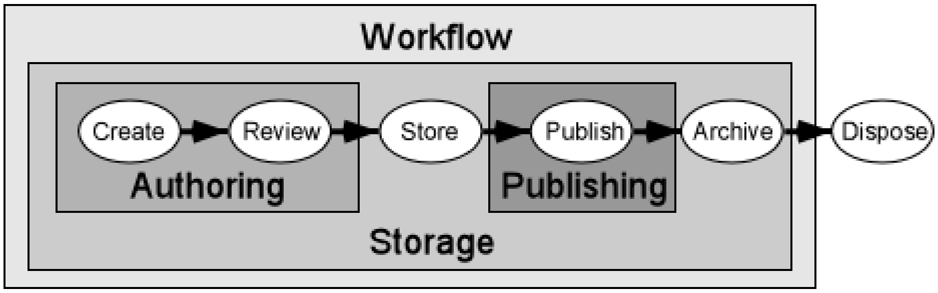

Figure 2: CMS functional scope and the content life-cycle
(after Ort [1] and Vidgen et al. [2])

Workflow is the management of steps taken by the content between authoring and publishing. Typical steps could be link checking and review/signoff by a manager or legal team. If workflow has existed at all in traditional Web site management it has been an off-line affair and not built in to software processes.

Storage is the placing of authored content into a repository. Beyond this it is also the versioning of the content, so that access conflicts between multiple authors cannot arise and so that previous versions can be found and restored if required. It can also mean breaking down content into structured, meaningful components such as `<job title>`, `<course>` or `<description>` which are stored as separate elements. These can be stored as records in a database or as Extensible Markup Language (XML) files.

Publishing is the process by which stored content is delivered. Traditionally this has meant 'delivered to the Web site as HTML'. However, it could also mean as an e-mail message, as an Adobe PDF file or as Wireless Markup Language (to name but a few). In the near future multiple delivery mechanisms will be required, particularly as accessibility legislation starts to bite.

We have summarised the advertised features of CMSs into a 'CMS Feature Onion' (Fig. 3). Table A1 (see Appendix) details these features and explains them. The feature sets typical of the different styles of CMS are reviewed in greater depth below.

The Issue

Institutions are struggling to maintain their Web sites. Out of date material, poor control over design and navigation, a lack of authority control and the constriction of the Webmaster (or even Web Team) bottleneck will be familiar to many in the HE/FE sector.

The pre-millennial Web has been characterised by highly manual approaches to maintenance; the successful and sustainable post-millennial Web will have significant automation. One vehicle by which this can be achieved is the CMS.

The concept of 'self-service authoring', whereby staff do not need special skills to edit the content for which they are responsible, can be regarded as a major step towards acceptance of the Web as a medium for communication by non-web specialists. Providing this is the key advantage of a CMS. As local information systems integrate and become more pervasive, self-service authoring extends to the concept of 'write once, re-use anywhere', in which the Web is treated as just another communication channel along with email, word processor files and presentations, etc.

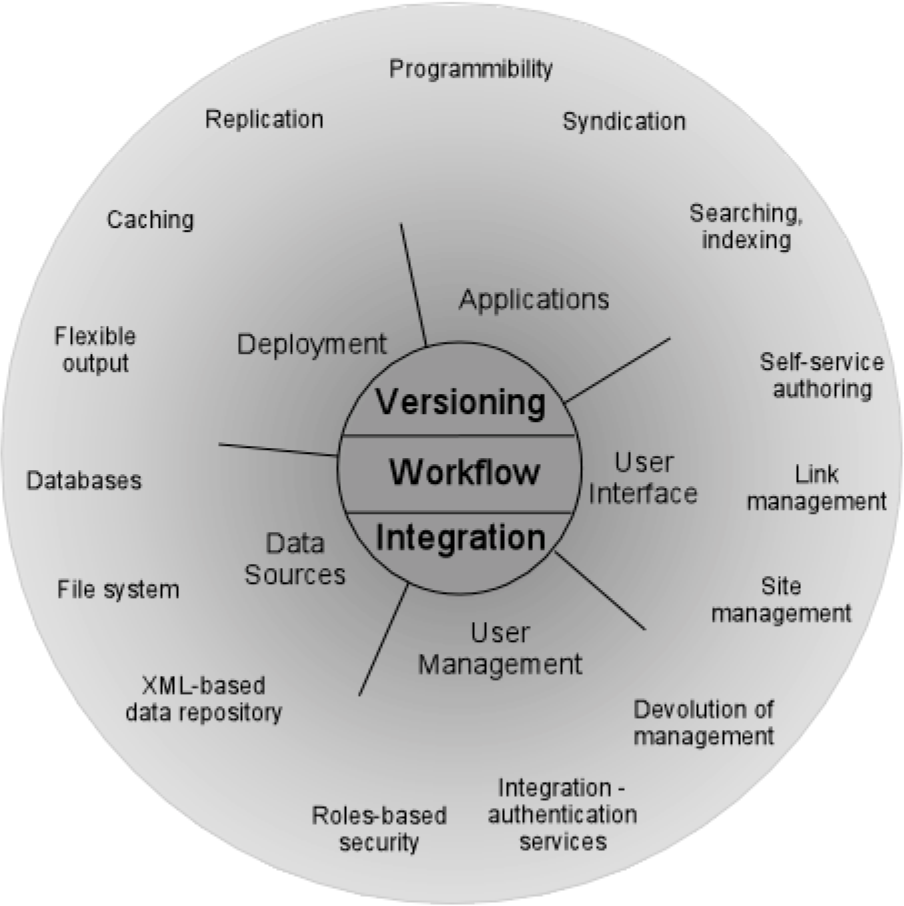

Figure 3: The CMS Feature Onion

The market place is very crowded and continues to grow (Faulkner Information Services estimates the market will grow to $65 billion by 2003 [3]). A list of 98 products claiming to offer a CMS (or components of one) can be found in Table A2 (Appendix). Comparing a list of products compiled in October 2000 with Table A2 reveals a number of newproducts and the loss or re-orientation of many others, although the number of products available continues to increase and diversify. Currently, therefore, investing in a CMS is potentially more of a risk than for other, more mature categories of information system.

This state of affairs, however, should not dictate inaction. The core features offered by most CMSs are sufficiently well developed to make the conversion of a traditional web site an undertaking that should be welcomed by all stakeholders, without the fear of having to re-engineer the web site again in the future.

The boundaries of the CMS space are blurred. Substantial overlaps exist with document management systems, knowledge management systems, source control systems, enterprise application integration systems, e-commerce systems and portals. We also contend that there are significant (but as yet not generally recognised) overlaps with intranet groupware and virtual/managed learning environments. Indeed, it may turn out that one institution's 'managed learning environment' is another's CMS.

Products

The core features of a CMS

In order to provide the functionality required by a complex, large scale, multi-author anddynamic web site then many features are desirable. Some CMSs try to contain them all, but it is unlikely that everything you may need will be available in a single product. It is the authors' experience that a pragmatic 'buy and build' approach is best for the HE/FE sector [4]. The features likely to be of interest to the HE/FE sector are displayed in the 'Feature Onion' (Fig. 3).

To be called a CMS a product or set of tools will, in our view, provide three core functions:

- **Versioning,** so that groups of individuals can work safely on a document and also recall older versions.
- **Workflow**, so that content goes through an assessment, review or quality assurance process.
- **Integration**, so that content can be stored in a manageable way, separate from web site design 'templates', and then delivered as web pages or re-used in different web pages and different document types.

This core feature set is augmented by a list of additional functions that varies significantly from product to product. These additional CMS features can be grouped into the five major categories shown in Table 2.

The three core functions and five associated categories summarised above are broken down further into specific features in Table A1. A complete product – feature matrix is beyond the scope of this report, though others have attempted this [5, 6, 7]. However, it is hoped that the table will give the reader an impression of the diversity found in such systems.

The CMS product marketplace

There is a tendency for those new to the technology to lump all CMSs together, as has been done in the product list Table A2. In some comparative reviews of CMSs, products with widely different origins, functionality and goals are compared as like with like, often because not enough information about the systems is readily available.

Category	Description	'Standards'
User Management	Assigning a role to a user, providing access rights and perhaps the level of interaction with the system. This can often use existing authentication schemes.	LDAP, Active Directory, ACAP
User Interface	Preferably a browser-based application for both content provision and CMS and/or web site administration.	HTML, Javascript, Java, HTTP, FTP, WebDAV
Data Sources	These include the managed storage of created content, plus external data in so-called 'legacy systems' (Word or Excel files, for example, could come under this heading) or other CMSs. Storage methods can be file systems, flat file databases, relational databases, and more recently, object oriented databases and XML files. The key is in the flexibility of the system to cope with its intended use. Storage also requires that the data itself is described. This is known as metadata, and creating it should be a requirement of storing content.	SQL, XML, Dublin Core, RDF
Applications	These integrate the content with existing data and authentication systems, and perform specific software manipulations on the content to aid consistency, simplicity and management. The key application is usually a form of 'templating' allowing control of web site 'look and feel' to be centralised and making style and navigation changes simple to implement. It can also include the 'middleware' that connects database records to dynamically created web pages.	Perl, PHP, Java, Python, ASP, XML-RPC, SOAP, ODBC, JDBC
Deployment	Publishing the web site to the live web server(s). Some CMSs do not distinguish between development and production servers, running the web site itself from the same software as the development system, creating pages dynamically on demand. Popular pages can be built in this way and then 'cached' in memory or on disc, speeding up future retrieval. Other systems have a strict partitioning of 'staging' and 'public' environments requiring separate web servers, often residing on separate machines. In this case the entire structure may need to be replicated if all pages are dynamically created. In other cases, certain elements of pages are pre-rendered and published as static content, with only specific dynamic content being accessed via the public server. It can be said that there are almost as many different methods of live publication as there are products available.	HTML, CSS, XHTML, XSL, XSLT, RSS

Table 2: Non-core CMS feature categories and related 'standards'

Type	**Origin**	**Strengths**	**Examples**
Document Management Systems	Software designed to manage the storage and internal publication of ‘corporate’ information	Document lifecycle workflow, metadata, document translation	Documentum, Panagon 2000, ChangingPages
Electronic news/magazine publishing	Tools developed to aid online publishing of magazines and news websites, and electronic discussion groups	Simple workflow, speedy publication of simple content, authoring tools, information management (structure), timed delivery	Eroom, Expressroom, Conversant, SlashDot, Frontier
E-business / E-commerce	Software underlying online shopping and electronic customer relationship management	Simple database management, website personalisation, built in transactional systems	Vignette, Broadvision, ATG Dynamo, Open Market
Source / versioning management control	Software engineering process control/source control among groups of programmers	Roles-based authoring and version control, workflow, templating systems.	Content Management Studio, Interwoven TeamSite
‘Middleware on steroids’	Tools for dynamic web site creation from filesystem and database assets. E.g. PHP, ASP, ColdFusion, JSP. Products such as WebObjects, Cache Objects for the Web and Tango 2000	Asset management, dynamic delivery and simple authoring environments.	Enhydra, Midgard (PHP), Dmind DSM (ColdFusion), SiteGenesis, Obtree.
Web content management frameworks	‘Second generation’ tools, built from the ground up for dynamic website creation and management. Can have very diverse conceptual grounding.	Variable, most try to cover all functions.	ACS, eGrail, Engenda, Mediasurface, NetObjects Fusion, Spectra, Xpedio, Zope
XML processors	A ‘third generation’ of CMS products appearing based upon XML technology.	Granular control and re-use of content. Though many of the above systems can now utilise XML, these products are written specifically to create websites using XML to store data, and related technology (XSLT, RDF) to manage and deliver it.	Cocoon, Interwoven TeamSite Templating, Lychee, Rhythmix, Tamino, POET

Table 3: Different CMS types

Vendor literature is generally over-hyped and jargon-ridden. Terms like ‘personalisation’, ‘syndication’, ‘asset management’ and ‘re-purposing’ abound. This can cause confusion among those who are confronted with major technological and financial decisions.

However, it is possible to define several generic types of CMS based on their apparent provenance, and this can help in assessing their suitability for a particular use. Feature sets are converging, though products are often aimed at different markets.

Table 3 provides a list of the broad approaches that appear to have been taken by CMS developers and gives selected examples.

Which of these could be a best fit for the HE/FE sector? Given the diverse nature of content that an institutional web site must deliver, it would be misguided to consider a product with a specific focus on e-commerce, or with a single database as content repository for all applications. Similarly systems with a document management provenance are likely to be tuned for intranet usage rather for the outward facing institutional Web.

Generally, more ‘open’ and framework-like products are more likely able to handle the broad range of content a HE/FE institution wishes to communicate. Care should be taken to examine the ‘out-of-the-box’ features as this will define how much post-purchase customisation is needed. The experience of mainstream business is that many large scale CMSs cost more to implement than to purchase. It is never a case of simply ‘buy’, but ‘buy and build’.

Developments

The large number of competing products in the CMS space has already been noted. Many analysts see a market shake-out as inevitable (not unlike that which occurred as the relational database market matured). The recent acquisitions of NCompass by Microsoft and Allaire by Macromedia are perhaps the start of a pattern, which will be repeated by other big players. Once this happens it will be hard for minnows to compete if they have an orthodox business model.

An alternative approach to web content management is to build a CMS from scratch using ‘middleware’ plus other component tools (only some of which are listed in Table 3). This option has been fuelled by the open source movement (and some high profile expensive failures [8] of CMS projects using ‘market leading solutions’). It has led to the emergence of a new business model (which is perhaps exemplified by Digital Creations - the providers of Zope) in which a company ‘gives away the family jewels’ but then generates revenue from consultancy work.

As well as the ‘buy’ and ‘build’ routes, a third option is also available - ‘rent’. The ‘application service provider’ (ASP) model has attempted to get a foothold in many sectors and the CMS space is no exception. However, the same issues (e.g. security of information, vulnerability of service) that have led to a slow embrace of the ASP option in other sectors, applies equally to CMSs [9].

Standards in this area are still forming, with no clear ‘winner’ beyond the consensus that XML will be an important framework for inter-application communication. XML and XSLT can be used to separate content from presentation, but are far from universally accepted, as they require a fundamental restructuring of ‘legacy’ content. On the authoring side, few tools are yet available for easy creation of XML content. For new content initiatives however, the ability to use XML as a ‘container’ for content will be desirable. We can also expect interesting developments that are driven by a maturing WebDAV standard and the commercial mainstreaming of object databases.

The UK government has recently made recommendations for metadata in the public sector, using the Dublin Core [10], and we expect this endorsement to accelerate developments in this area.

Assessment and Recommendations

The pre-millennial Web can’t scale. The Webmaster bottleneck has been replaced by a Web Team bottleneck and our information systems are hard to join-up. The CMS can allow users to bypass the Web Team bottleneck and allow integration and re-use of information.

We feel that at the core of any CMS one should find a robust set of tools for content **versioning**, content **integration** and process **workflow**. This core feature set will augmented by a list of additional functions that varies significantly from product to product.

Typical future uses within HE/FE are easy to identify. At a workshop run by the authors [4], participants identified the following as the most desirable features of a CMS[1]:

1. Template-based self-service authoring for non-technical content providers (‘frictionless publishing’)
2. Roles-based security
3. Workflow management – submit, review, approve, archive
4. Integration with existing data/databases and user authentication systems
5. Metadata management
6. Flexible output – write once, publish many times

We recommend that these functions form an initial set of global requirements through which prospective CMS systems can be filtered.

[1] Perhaps the CMS isn’t as novel to HE/FE institutions as one might suppose. As one reviewer of this report observed ‘this is what our library management system does’. Acquisitions staff use Web-based templates to create records for new stock. Cataloguers add value to the record (with less experienced staff being subject to an approval process). Metadata is recorded automatically detailing who entered/modified content and when. A subset of the record is then re-used in several ways - as a page on the Web OPAC, as paper recall notice or an e-mail recall notice.

In an immature market place the risks to investment are substantial. If you ‘buy’ you cannot be sure whether your vendor will survive or whether the product will turn-out to be a technical cul-de-sac. If you ‘build’ (and smaller institutions may not have this as an option) you carry the overheads and risks of recruiting and retaining specialised staff.

At present it appears unlikely that a single product will cover the requirements of a complex organisation. There seems little argument that the solution will be a mixture of ‘buy and build’. The key issue is one of balance - ‘How much must be built before we get what we need?’. Institutions will also be exercised by the question of ‘Who builds it?’

We detect resistance within traditional campus computing services and MIS departments to the concept of the CMS (and indeed the need for an automated, post-millennial Web). Fundamentally a CMS devolves control over content to the owners of that content (rather than the technician), and then scales without increasing management overheads. The investment can run to a six-figure sum if the perceived market leading systems (e.g. from Vignette, Broadvision and Interwoven, though the former two are heavily biased towards commercial applications and thus, perhaps not a good fit) are considered, though many more modest (and indeed no-purchase cost) products are available.

The emergence of ‘portal frameworks’ (open source or otherwise) has done much to highlight the overlap and convergence of document management systems, knowledge management systems, enterprise application integration systems, e-commerce systems, intranet groupware, virtual/managed learning environments and CMSs. There is a pressing need, in our view, for institutions to think holistically (reinforced by their work on information strategies) and to invest in and develop open and extensible information systems.

Senior managers need to be aware of the costs and consequences of not embracing the post-millennial Web. The cost of investing in a CMS can be on a par with procuring a student record system - but which, in terms of efficiency and effectiveness, is likely to give the biggest return on investment? In fact, there is no choice, because both will be essential items. But we suspect, in terms of competitive advantage, it will be the CMS (perhaps as part of an overarching portal framework) that will be more important in terms of differentiating institutions.

We know that readers of this report would prefer if Table 4 were more exhaustive. A proper evaluation of a CMS, as with any information system, is a major undertaking. Moreover, pricing information is notoriously hard to come by [11].

As it stands Table 4 is based on anonymised feedback from participants in the CMS Parallel Session at the Fifth Institutional Web Management Workshop held at Queens’ University, Belfast during June 2001. Each vendor or product had been investigated (however briefly) by at least one of the participants’ institutions.

Participants in the CMS Parallel Session also had the opportunity to read and comment on a draft version of this report. In terms of what the sector might do next in the area of CMS, support was voiced for:

- Funded case studies of CMS evaluation exercises (or at least funding to encourage institutions that had already undertaken such exercises to share their findings with the sector)
- The formation of both formal and informal consortia
- A mailing list to facilitate the formation of consortia and over which UK experiences could be shared
- A generic template which might be used in invitations to tender
- The purchase on behalf of the sector (in the manner that was done for reports on information security) access to reports on CMSs from consultancies such as Gartner Group or Forrester Research

Price range	**Vendor/Product**	**Evaluated by College/University X**
£0-20000	Communique (£2k) Frontier (£99) NetObjects Fusion (£10k) RedDot (£20k) Zope (£0k)	
£20-50000	Communique (£30k) Obtree (£40k)	
£50-200000	ATG (£200k) Communique (£100k) Interwoven (£120-200k) Mediasurface (£170k) Diagonal (£120k) Radius (£120k)	
>£200000	Interwoven (£200-500k) Vignette (£500k)	

Table 4: The information HE/FE needs

Acknowledgements

Rob van Buuren, Richard Vidgen, Geoffrey Ford, Barry Taylor, Bruce Dupee, Stuart Brown, James Currall, Darren Chapman, Amanda Wheatley, Jethro Binks, Jacki Hargreaves and the participants of the CMS Parallel Sessions at the Fourth and Fifth Institutional Web Management Workshops.

References

1 Ort, E. (2000) Ten Things to Know About Selecting a Content Management System, Dot-Com Builder, http://dcb.sun.com/practices/howtos/selecting_cms.jsp

2 Vidgen, R., Goodwin, S., & Barnes, S., (2001) Web Content Management, In: O'Keefe, R., Loebbecke, C., Gricar, J., Pucihar, A., & Lenart, G., editors, Proceedings of the 14th Bled Electronic Commerce Conference, Bled, Slovenia, June 2001, pp. 465-480.

3 Faulkner Information Services, http://www.faulkner.com/

4 Doyle, L. (2000) Content Management Systems Workshop Report, Fourth Institutional Web Management Workshop, University of Bath, http://www.bris.ac.uk/ISC/cms/summary.htm

5 Barrett, C. (2000) Content Management Systems, http://www.camworld.com/cms/

6 Rapoza, J. (2001) Get A Grip On Your Site, PC Magazine, http://www.zdnet.com/products/stories/reviews/0,4161,2688945,00.html

7 Thomas, M. (2001) CMS decision matrix, http://www.cornerconsulting.com/wcm/ (cited at http://cms.filsa.net/archives/cms-list/2001/Feb/0225.html)

8 Walker, D. (2001) Content management systems: short-lived satisfaction, http://www.shorewalker.com/pages/cms_woes-1.html

9 Sweeney, T. (2000) ASPs Answer The Security Question, http://www.informationweek.com/789/asp.htm

10 e-Government Interoperability Framework, http://www.govtalk.gov.uk/egif/home.html

11 Kontzer, T. (2001) Vendor Selection: Beyond The Hype, http://www.informationweek.com/839/online_cmside.htm

Glossary

ACAP	Application Configuration Access Protocol
Accessiblitiy	At one level. this refers to the compatibility of websites with web browser software. At a second level, it refers to the suitability of content for browsing via text only, large text or text-to-voice / text to braille systems.
Active Directory	Microsoft’s directory service, proprietary but based on LDAP.
API	Application Programming Interface. The software command set provided to programmers by large-scale software applications.
Application integration	Methods and tools for sending data and commands between different software applications.
ASP	Active Server Pages: A commercial dynamic web page and database access language
ASP	Application Service Provider: a company providing software or services that run over the internet
asset management	The management of digital objects such as text, images, movies and sound.
cache	An area of computer memory or disk space where dynamic web pages can be held for speedy delivery.
Cold Fusion (CFML)	Cold Fusion Markup Language: A commercial dynamic web page and database access language.
CRM	Customer Relationship Management. A goal of web ‘personalisation’: Commerce based CMS’s include this or hook in to CRM applications.
CSS	Cascading style sheets. A method for separating content from style in web pages.
document management	Software for storing, organising, archiving and publishing documents in electronic form.
Dublin Core	Dublin Core ‘Simple’ is a ‘standard’ set of metadata fields, such as Author, Title etc. which can be applied to web pages.
Dynamic generation	The creation of a web page from a set of elements (objects such as images, templates and database records) ‘on the fly’ when a user requests the ‘page’ via a web browser. Can allow instant customisation of the page, depending on the data supplied by the user.
FTP	File Transfer Protocol, an internet standard for communication of data that preceded HTTP. Still very popular.
groupware	Software for on-line collaboration and messaging.
HTTP	HyperText Transfer Protocol. The Web communication protocol.
Java	A computer programming language.
JDBC	Java DataBase Connectivity. A system allowing software written in Java to communicate with databases.
JSP	Java Server Pages: a dynamic web page and database access system based on the Java language.
knowledge management	Software and information concepts for managing the context of and relationships between bits of information
LDAP	Lightweight Directory Access Protocol
Metadata	Data about content, such as the author, date created, expiry date, subject area.

middleware	Software for connecting web pages to ‘back end’ software systems and databases.
MLE	Managed Learning Environment: A system that uses technology to enhance and make more effective the network of relationships between learners, teachers and organisers of learning, through integrated support for richer communication and activities. An MLE usually embraces a VLE.
navigation	The structure of a website as defined by the links between pages.
NT PDC	Primary Domain Controller, an authentication server for NT networks.
ODBC	Open DataBase Connectivity. A system allowing software to communicate with databases.
open source	Software that is sold or given away with the intent that the software user can make modifications, so long as those modifications are put into the public domain.
PDA	Personal Digital Assistant: handheld computers.
PDF	Portable Document Format. A commercial electronic document format.
personalisation	A set of software techniques for tailoring a website’s style and content to a user’s preferences.
PHP	P(ersonal Home Page) Hypertext Processor: a dynamic web page and database access system.
portal	A website or web page that gives access to a list or database of other web resources on a given (or any, in the case of major internet portals such as Yahoo, Excite) subject. This word has been used more recently to mean any content-centric website.
pre-rendered	A static web page that was aggregated and generated from its component elements before publication to a ‘live’ web server.
Python	A computer programming language.
RDBMS	Relational Database Management System: software for managing the data and relationships between data in different database ‘tables’.
RDF	Resource Description Format: An XML syntax for communicating and storing metadata.
re-purposing	Displaying the same data in different ways for different purposes.
RSS	Rich Site Summary: an XML standard used for news dissemination.
self-service authoring	Authoring web content using familiar tools.
session	A web site ‘visit’ by a single user. Dynamic web systems can assign unique IDs to ‘sessions’ thus allowing personalisation and audit tracing etc.
SOAP	Simple Object Access Protocol: an open standard for inter-application communication that has grown out of XML-RPC.
source control	Versioning of documents and tracking changes to documents (comes from software development systems).
SQL	Structured Query Language: a syntax for manipulating databases.
staging	A version of a website that is complete but not ‘live’ on the Internet. Can be on a separate server. Used for integration of multiple author’s work, and final quality assurance before publication

syndication	The automatic aggregation of external content with a CMS or website, or the automatic deployment of content to an external ‘client’ CMS or website.
template	A framework web page (or set of elements) that has limited editable parts, freeing content authors from designing whole web pages.
VLE	Virtual Learning Environment: A software system that combines methods of online communication (such as e-mail, discussion forums and chat rooms) with the ability to deliver learning materials via the Web.
WebDAV	Web Distributed Authoring and Versioning: a versioning protocol.
Wireless Markup Language (WML)	A simplified form of HTML/XML for devices such as mobile telephones and PDAs.
workflow	The passing of content between people in the development chain, including authorisation steps.
XHTML	A reworking of HTML to make it ‘XML compliant’, mostly by making all tags balanced (closed by a /)and depreciating certain tags that mix content type and formatting roles.
XML	Extensible Markup language: <xml-definition>XML provides a way of describing a unit of content, such as this sentence</xml-definition>.
XML-RPC	A means of passing messages formatted as XML between a client and a server so that a program is run and the results returned (Remote Procedure Call).
XSL	Extensible Style Language: formats XML for display on a range of devices (including screen and printer).
XSLT	Extensible Style Language Transformation: allows one XML dataset to be transformed into another.

Table A1: CMS Feature List

Category	Feature	Description
Versioning	Auditing capabilities	An audit trail records the sequence of activities that occur on any given file or body of content within a content management system.
	Version control - check in/out, locking and merge	Can be applied by authors or developers responsible for producing various types of content, including application code, graphics, text, and other file-based content. If multiple authors are involved in developing content, the version control system 'locks' (makes read-only) a file in editing, or between editing and reviewing, to ensure that no two authors modify the same document at the same time. A development of this is the ability for multiple authors to edit content simultaneously, followed by a managed 'merge' process.
	Version control - roll-back	Systems also save past versions of content elements, files or entire projects, track the date and time of changes, and provide an option to keep a comment log. Everything from a text file to an entire site can be "rolled-back" to an earlier version.
Integration	Asset management	Manages the storage and reusability of elements of content. Allows multiple users to edit content and make changes without running over each other. See also Version Control.
	Custom content/asset types	Support for arbitrary content/asset types. Do you have to build your own administrative interface from scratch for each new asset type?
	Conversion of non-HTML documents	Automatic translation of format X to XML/HTML. Can be extended to include translation to text only, PDF, WML. Allows the use of standard desktop applications to create web content.
	Metadata management	Content classification systems enable you to organise key metadata value and attribute pairs to aid in web applications that fall outside core CMS functions, such as site analysis, personalisation, content targeting, web site and internet search results.
	Retrieval	When adding content to large collections of data in a CMS where you'd like to draw relationships between the data, the author needs to be able to retrieve and review the relevant info before confirming the relationship.
	Staging	Content management systems can explicitly provide virtual staging of Web site content, thereby supporting a formal quality assurance phase before final publishing to the production servers. The staging area is designed to emulate the production site in every way possible.
	Support for external thesaurus and other controlled taxonomies	Controls the application of metadata describing web content. See also "Metadata management".

	Templating	Content management systems and other development tools allow the site designer to easily create and establish templates to give the site a uniform look. These templates may also be modified when desired to radically alter the appearance of an entire site or specified areas. Consistency of look and feel - as well as the logical organisation of the site's content - is simplified when templates are used. Designers should design and authors should develop content. Having well-prescribed roles simplifies site management.
Workflow	Integration - messaging	Integration of other channels of communication into the workflow (i.e., email). Eg. on completion of new content, the CMS sends an email to the next person identified in the workflow, probably a content reviewer.
	Workflow	Workflow systems enable you to define multiple step processes involving varied content types, cross-departmental staff, and required actions, such as submit, review, or approve. Event-driven workflow systems can be built into larger content management systems or employed as a standalone service that integrates processes, people, and content as needed.
Data repository	Database storage	Storage of assets in a database integrated with the CMS can aid content re-use and management, especially where content is repetitive. It increases granularity of your assets beyond the 'file' level of HTML. This is often confused with content management itself, but is only part of the story. See XML-based data repository. Storage of assets in a file system can be more efficient, up to a point, especially on the public website. Flexibility in content storage is an advantage.
	Integration - existing data - databases	A data storage mechanism managed independently of the operating system by server applications. The applications can either store and retrieve data natively from disk or store and retrieve data from a file system object. Data stored within databases are only accessible from database application interfaces. These can be a core part of the CMS. Databases are designed for rapid, efficient search and queries for structured data. This can be achieved pre-deployment to create a 'flat' web page, or on-the-fly when requested, in a dynamic web page which may then be ‘cached’ for instant retrieval.
	Integration - existing data - file systems	A data storage mechanism natively managed by the server operating system. File systems allow operating systems to store and retrieve data from disk. Data is stored on disk logically categorized using into directories following a file cabinet metaphor. File systems are designed for rapid, efficient, scalable disk access for most common forms of saved data.

	XML-based data repository	Generic repository systems help you maintain and manage large amounts of diverse content. By storing content as XML, vendors claim improvements in content re-use and re-purposing. XML-based systems also include transformation services and content validation. The primary advantage of XML-based systems is that XML permits you to separate content, format, and business logic. Documents and Web pages can be dynamically constructed by combining many individual elements from the repository.
User management	Devolution of management	Can you have users who have users who have users? Can you devolve the administration and publication of parts of your website while maintaining ‘quality control’
	Integration - authentication services	You have 4000+ staff with e-mail accounts - you don't want the overhead of administering 4000+ separate CMS accounts. The CMS should be compatible with existing NT PDC, LDAP, etc.
	Roles-based security	Workflow systems allow administrators to control allowable actions based on each team member's role. Fine-grained, role-based security is important because it allows content creators and developers to interact with business managers by limiting them to the appropriate content access levels at different stages of a process.
User interface	Browser-based ("TTW - Through The Web")	Client and administration tools allow team members to configure, control, and participate in the system. Many of these interfaces are provided in cross-platform browser client form. Browser-based authoring permits contributors to work from any location as well as minimising support overheads.
	Editor, authoring tools	Site designers are not restricted in their choice of HTML editor. You have staff with no web skills who need to update content, but you also have very experienced web designers. A CMS can tailor the user interface to each type, providing a 'Web form' to some and a 'virtual sandbox' to others for use with any web design tool. See also "Self service editing..."
	Link management	See "Reporting".
	Planning, mapping and modelling	Tools to create a site structure, lay out the sections of the site, figure out the navigation, mark areas for future development, etc.
	Reporting	The reporting tools included with most content management systems are generally extensive and reflect the breadth of coverage of its supported functionality. Examples include performance, workflow, link status, page size, possible slow pages, daily submissions. If the CMS runs the live website, then this can include access and error log file analysis.

	Self-service authoring for non-technical content providers	Content authors should be able to quickly create materials using standard desktop applications, as well as image, video, and other types of media files. After materials have been developed, content providers should be able post them to the Web using pre-built design templates that provide the proper formatting and corporate branding elements. Non-technical content authors should be empowered to keep content up-to-date and be able to directly post content to either a staging environment or a live Web site.
	Site management	Most Content management systems provide a 'big picture' view of the Web site that allows site designers to create an overall navigational model for the entire Web site.
Applications	Common API	Though there is a desire by vendors to lock in their customers to use their solution only, they also see the value of providing methods for other tools to integrate easily with their content management systems. Some vendors provide this interoperability through a published Application Programmers Interface (API) that enables onsite developers or third party vendors to tap into the functionality of their system.
	Integration - marketing tools	With advertisement servers, marketing campaign management tools, and group-filtering products.
	Localisation	Your content is available in a variety of languages.
	Programmibility	Can you script the CMS (or link to an application server) so as to automate tasks or generate dynamic pages?
	Scheduling of content publishing and archiving	Allows content providers to determine when content is presented on the site and when it is removed. Automating this process ensures that date-sensitive information is available only while relevant.
	Searching, indexing	Content management systems often provide support for integrated indexing and search functionality. By embedding categorised metadata it is possible to fine-tune desired search results and ensure that customers can easily find what they want. A indexing application "walks" a file system, reads every text file, and builds a mini-database of content elements (most typically content metadata).
	Custom tag definition	Can you define your own tags in the templating language?
	Business rules	A business rule may be referenced in conjunction with a personalisation engine. Business rules determine which users are delivered a specific type of content. However, simply targeting a few key groups (groupalisation?) offers many advantages at lower costs. See Personalisation.
	Syndication	Can you enrich content of your site by automatically importing materials from one or more other sources? Can you export your materials to other sites that may wish to use them?
	eCommerce	Including secure transactions.

	Personalisation	The ability to personalise the user's experience is dependent on linking dynamic page content to previous user actions. Personalisation also requires a membership database to store customer attributes and purchasing history.
	Session analysis	The process of observing a customer's behavior as s/he clicks through the Website and storing that click-through information in a user profile database. Behavior tracking enables business managers to target content with tailored business rules. By logging user activities you can produce reports that clearly show popular areas, average visit time, and other important information.
Deployment	Caching	Content caching of dynamic pages, the ability to mark Web elements as cached in server memory, influences site performance.
	Flexible output	Soon Web browsing will take many forms, from PDAs to cell phones. You will want to ensure that your site looks right regardless of how a user accesses it. You will continue to want to re-purpose content for paper, CD-ROM and voice.
	Replication	After staging and testing Web content must be transferred to the production server. Some systems only replicate common Web elements such as Web pages, graphics, multimedia files. However, to be complete, the deployment system should also include application components, applets, registry settings, access control lists; in short, it should provide a mirror image of the staging server environment. Replication typically involves specific rules about what gets deployed where, and often represents a complete synchronisation of two independent bodies of content.

Table A2: CMS Product List

Product	Vendor	Web site
4i Content Management	Documentum	www.documentum.com
ACS (CM module)	ArsDigita	www.arsdigita.com
Affina	emojo	www.emojo.com
ArticleBase	Running Start	www.running-start.com
Asset House V5	Asset House	www.assethouse.com
Astoria	Chrystal	www.chrystal.com
Bladerunner	Interleaf	www.xmlecontent.com/products/im.htm
ChangingPages	ChangingPages	www.changingpages.org
CMDS	Backstream	www.backstream.com
Cocoon	Apache.org	www.java.apache.org/cocoon/
Cofax	Cofax.org	www.cofax.org
CommonSpot	PaperThin	www.paperthin.com
Communique	Daynetwork	www.daynetwork.com
CONTENS	Contens Software	www.contens.de
Content@	Xyenterprise	www.xyenterprise.com
Content Centre & Content Server 3.1	Open Market	www.openmarket.com
Content Manager	IBM	www-4.ibm.com/software/data/cm/
Content Server / Content Engine	BridgeMedia	www.bridgedns.com/ce/
ContentStudio	Rational Software	www.rational.com/products/contman.jsp
ContentWare	OpenPages	www.openpages.com
Conversant	MacroByte Resources	www.macrobyteresources.com/Conversant/
CoreMedia	CoreMedia	www.coremedia.de
Corenet	Core Technology	www.corenet.ch
Cybersword	Cybersword	www.cybersword.co.uk/home.htm
DialogServer	Tridion	www.tridion.com
Dispatch	DesertNet	www.dispatch.net
Drillpress	Drillbit	www.drillbitonline.com/Products/Drillpress/
DVP.HTMPlates Pro	Digital Vantage Point	www.dvp.net/htm_pro.htm
Dynamic Web System	SmartSite	www.smartsite.nl
eBT	eBT	www.ebt.com
eBusiness Express	Webridge	www.webridge.com
eContent 1.31	Javacorporate	www.javacorporate.com
eGrail 2.5	eGrail	www.egrail.com
eKeeper	eKeeper	www.ekeeper.com
eMPower 2.0	Ektron	www.ektron.com
entrepid, engenda, DynaBase	E-Business Technologies	www.ebt.com/products/
Erc Engine	Erc	www.erc-engine.com
Excosoft Content Manager	Excosoft	www.excosoft.se
Expressroom	Starbase	www.starbase.com
Frontier	Userland	frontier.userland.com
Fusion	NetObjects	www.netobjects.com

Generic Information Server Toolkit	JRC	gist.jrc.it
iNes Content Management	Syntagma Rendezvous AB	ines.syntagma.se/page.asp?document=product_wcm
Insertrix	Top Floor Software	www.topfloorsoftware.com
interact!	COIN Corporate Interactive	www.getinteract.com
IO CMS	IO	www.io-pt.com
Kosmos	Kadius	www.kadius.com
Lychee	NetCentric Systems	www.netcentricsystems.net
MasterIT/WebCM	Computer Associates	www.cai.com/products/masterit_content.htm
MediaDepot	e Media-IT	www.mediadepot.co
Mediasurface 3	Mediasurface Ltd.	www.mediasurface.com
Midgard 1.4	Midgard	www.midgard-project.org
MMBase	MMBase	www.mmbase.org
NoCode	NoCode Inc	www.nocodeinc.com
NUA Publish 2.0	NUA	www.nua.ie/products/nuapublish/index.shtml
Obtree C3	Obtree	www.obtree.com
OmegaCMS	OmegaCMS	www.omegacms.com
One to One Publisher	BroadVision	www.broadvision.com
Open Share 2.1	Info Square	www.infosquare.com
OpenCMS	OpenCMS	www.opencms.com
OS Content	Open Symphony	www.opensymphony.com
Panagon	FileNet GmbH	www.filenet.com
Participant Server 2.0	Eprise Corp.	www.eprise.com
passage	Passage	www.passage.nl
Polopoly Content Manager	Polopoly	www.polopoly.com
PVCS Content Manager	Merant	www.merant.com
Q-Web	SYSDATA	www.sysdata.es
Rational	Content Management Studio	www.rational.com/products/contman.jsp
RedDot	RedDotSolutions	www.reddot-solutions.co.uk/products.html
RedSnapper - Builder	RedSnapper	www.redsnapper.net/en/1/atucms.html
Reef	Reef	www.reef.com
Resolution	Microsoft (NCompass)	www.ncompasslabs.com
Revize	Idetix	www.idetix.com/index_revize.htm
Rhythmyx Content Manager	Percussion	www.percussion.com
Roxenplatform 2.1	Roxen	www.roxen.com
Site Station 4.6	NST Systems	www.sitestation.com
SiteBuilder	SohoNet	www.sohonet.com/home/products
Sitecontentmanager	Sitecontentmanager.com	www.sitecontentmanager.com
SiteGenesis 4.5	Cassini Division	www.cassini-division.com
SixCMS	Six Offene Systeme	www.six.de
Spectra	Macromedia	www.allaire.com
StoryPage	TW3W	www.tw3w.com
SyCOMAX	SyCOMAX	www.sycomax.com
TeamSite	Interwoven	www.interwoven.com
Toutatis	Toutatis	www.toutatis.com
Update Engine 5	FatWire	www.fatwire.com
Update This	UpdateThis.com	www.updatethis.com

V/5 Content Management Server (fka StoryServer 5)	Vignette	www.vignette.com
VIOMA Content Manager	VIOMA	www.vioma.com
VIP's Platform	Magellan/Gauss interprise AG	www.gaussinterprise.com
WebCAS	Elinear	www.elinear.com
WebCast Pro	Astound.com	www.astound.com/products2/wcpro/wcpro.html
WebGate 5	InnovationGate	www.innovationgate.de
WebSite Director	Cyberteams	www.cyberteams.com/products/wsd/index.html
WEBdirector	Xbot	www.xbot.com
websiteMAX internet	Maxium	www.websitemax.com
Xobix	Xobix	www.xobix.com
Xpedio 4.0	Intranet Solutions	www.intranetsol.com
Zope	Digital Creations	www.zope.org

This is a list of stand-alone CMS software products or suites, supporting most or all the features we define as core. This list does not include software web application development or e-commerce platforms. Such software can often be used to build CMS solutions.