



Top 5 Reasons Content Management Projects Fail

And How to Avoid Them

Written by:

Janus Boye, Managing Director, Boye IT

*More white papers
are available at:*
www.fatwire.com

Introduction	2
Content Management Failure Defined	2
Why Content Management Projects Fail	2
#5 Organizational Acceptance Fails	2
How to Avoid Reason #5	2
Involve the different project stakeholders upfront to define requirements	2
Include business representatives throughout the process	3
Balance individual requirements with the need for overall project success	3
#4 Ownership is Unclear	3
How to Avoid Reason #4	4
Anchor ownership in business units	4
Identify business and technology costs	4
Create a competency center	4
#3 Lack of Measurable Success Criteria	4
How to Avoid Reason #3	5
Define business objectives with quantifiable success factors	5
Use Key Performance Indicators (KPI)	5
#2 CMS is Too Complex and Difficult to Use	5
How to Avoid Reason #2	6
Think about usability for the entire team	6
#1 What is Content Management?	7
How to Avoid Reason #1	7
Define what content management means to your organization	7
Create a basic project glossary	8
FatWire Software Helps Ensure Project Success	8

Introduction

The explosion of content created for the Web has led most organizations to investigate more efficient ways to produce, store, and deliver electronic content intended for use on intranets, extranets, and Internet Web sites. A content management system (CMS) is a primary tool employed to improve the speed of content publishing, increase content re-use, expand the group of users who can easily create and change content, and decrease overall expense. However, despite the benefits of content management (CM), too often content management projects don't offer anticipated benefits or fail to reach their full potential.

This report shares insights the author has gained by working with dozens of organizations on the implementation of CM projects. It describes the most common pitfalls companies experience with CM, and provides practical tips for avoiding them.

Content Management Failure Defined

Content management project failure is defined as an inability to meet pre-defined business goals with a quantifiable return-on-investment (ROI). This can mean that the cost of the CMS implementation exceeds the savings or increased revenue derived from the project. It might mean that content publishing processes aren't improved, or even worse, the new CMS is underutilized because it is more difficult to use than the previous solution. In some cases, it means that the CMS is never fully deployed due to implementation difficulties or lack of resources.

Why Content Management Projects Fail

Here are the top 5 reasons that content management projects fail, in reverse countdown order:

#5 Organizational Acceptance Fails

A successfully deployed CMS will likely have a far reach within an organization. Ideally, diverse constituents who own various aspects of a business, and therefore its content, will make use of a CMS and embrace the changes that new content management processes will inevitably bring. However, when a content management project is initiated without large-scale buy-in from relevant business groups and stakeholders, the acceptance of a new CMS will often be met with roadblocks. Groups or individual users may refuse to use the new system or may actively work against its successful implementation. There may also turn out to be conflicting projects underway in other groups.

An individual project champion, such as an HR person, a marketer, or someone in the information technology (IT) department, typically drives a new content management project. Whether it is initiated by a business or technical group, a project is at risk if it doesn't cross organizational boundaries.

How to Avoid Reason #5

Involve the different project stakeholders upfront to define requirements

It is crucial to develop a cross-functional team at the outset of a content management project to define requirements. It is also important for this team to evaluate existing formal and informal processes involved in managing content to determine how these will

translate in the new system. A CMS often highlights ill-defined processes by making publishing tools available to more users and enabling the creation of digital workflows. If all content previously went through IT and the CMS eliminates this bottleneck, what business processes are now required? Only a cross-functional, cross-organizational team can effectively define these.

Include business representatives throughout the process

To ensure that the chosen CMS meets diverse business requirements, don't limit participation in the selection process to IT. IT has a role to play in ensuring that its needs are met by the new system for requirements like enterprise architecture, user-friendly administration, scalability, and developer tools, but IT shouldn't define the requirements of the business users. The content contributors, editors, and managers who will be using the system should represent their own needs. It is vital to include business representatives not only in the product selection process, but also throughout the implementation, deployment, and maintenance phases of the content management program.

Balance individual requirements with the need for overall project success

While it's important to ensure that all business requirements and goals are gathered, it's also necessary to get a system implemented in a reasonable amount of time. There is a danger of defining an 'ideal world' set of requirements that may require many customizations to any available packaged product. At the same time, make sure requirements are based on business needs and not just available product features. Don't let product features define how your business processes work. Look to balance unique business requirements with standard functionality and implement what makes sense and can be completed in an acceptable timeframe. Keeping project cycles below three months is an ambitious but possible target, even at large enterprises. Many organizations start with a smaller, less visible site in order to show quick success, before tackling mission-critical sites.

#4 Ownership is Unclear

While large-scale acceptance of content management is a requirement for project success, clearly defined ownership is also an obvious necessity that is too often missing. And most likely, a project will run more smoothly with more long-term success if it is sponsored by a business champion outside of IT.

Management and the responsible business units (e.g. marketing, communications, sales, HR, etc.) may excel at creating content, but may be unlikely to claim ownership of a content management project due to a lack of experience and knowledge. As a result, project direction often defaults to an IT group, which can create a chasm between those who own the business requirements and those who own the project. The IT department, while essential to project success, will invariably have a different set of goals and incentives than the various customer-focused business units that have content management needs.

Similarly, a lack of business management responsibility for a project can lead to a lack of ownership altogether. While IT owns the technical implementation, they may not control the resources and budget necessary to make the project a success and therefore can't really own the project. Without clear project ownership, it's difficult to track milestones, meet deadlines and keep participants motivated.

How to Avoid Reason #4

Anchor ownership in business units

Content management is not just a technology project. While the project might be technically focused initially, it is really about business processes such as building and maintaining strong editorial teams and continually creating quality content. Having a project owner grounded in these requirements can help ensure alignment. Also, as acceptance by business users is critical to success, involving them early and empowering them with ownership will invariably make them less hostile towards the changes that a new CMS will bring.

Senior management needs to have final say on both resources and overall priorities. Therefore, in the end, it will be up to management to commit the proper amount of resources and budget and to provide essential prioritization of requirements. Even if IT owns CMS funding, that doesn't mean that IT should own the CMS project. Business managers should work through IT procurement processes to fund content management as a business project to ensure buy-in of business users and alignment of requirements.

Identify business and technology costs

As part of this process, evaluate the total-cost-of-ownership (TCO) for a CM initiative. This can provide valuable insights for estimating annual budgets. More importantly, it can help identify the true overall cost split between IT and the business. How much of the current and expected cost is actual technology cost (e.g. software, hardware, hosting, development, system administration, and integration) and how much is related to business processes (e.g. editorial processes, content audits and migration, workshops, project planning and training)? To the extent that the latter expenses will almost surely exceed the former, especially when calculated over a number of years, the project becomes even more clearly a business undertaking than an IT project.

Create a competency center

One practical approach to moving project ownership into the business is to create a cross-functional competency center that contains team members from all relevant business units, including IT. This content management competency center is a centralized project management and development organization that is dedicated to owning, maintaining, developing, and supporting CM technologies, processes, Web sites, and applications. The team becomes a knowledge base, utilizing established best practices, documentation, consistent communications, and standardization. All knowledge is shared and work is re-usable. The competency center may report to a steering group or a department head. In many organizations, the director of the competency center is the owner of the content management program.

#3 Lack of Measurable Success Criteria

It is possible with new and interesting technologies such as content management for projects to take on a life of their own, with stakeholders believing that they "must have" a content management project underway. However, as with other projects, success will be elusive if it's not tied to clearly defined business objectives. Without quantifiable success criteria, it is difficult to define requirements and meet goals.

Projects that lack clear objectives with milestones can drag on, as there's no consistent definition of success. Many projects lose funding or resources because it's not possible for the owners to show progress. Worse, an implemented CMS may not be used or it

may be underused as the project was never tied to sound goals and as a result, doesn't meet user needs.

How to Avoid Reason #3

Define business objectives with quantifiable success factors

Success factors in many CMS projects are too vague, such as "complete project within budget," "meet production deadline," or "migrate to a new CMS." A successful content management implementation needs to be tied to real business goals, such as reducing quantifiable costs, redeploying technical personnel, or increasing revenue. Then, define what constitutes "success" for each goal. Most projects will have some goals that may not be easily quantifiable (e.g. "save employees time" or "make the organization more flexible"). These objectives shouldn't be ignored but they also shouldn't be the project's key success factors as they're difficult to track.

Use Key Performance Indicators (KPIs)

Key Performance Indicators (KPIs) can help define and measure success. KPIs must be quantifiable and should be tied to key success factors. KPIs can be used for performance management and as a way to quickly show management the benefits of the CM project. Week by week, month by month, progress can be measured. Consistent tracking of KPIs ensures potential problems are identified early so that necessary changes can be made to systems or processes.

The following are examples of quantifiable KPIs:

- Goal: Reduce publishing and marketing costs
 - Example KPI: Cost of printing and distributing sales materials to partners.
 - Example KPI: Cost of creating and distributing product manuals.
- Goal: Create faster content production cycles
 - Example KPI: Average time it takes to move new content from creation to publishing.
 - Example KPI: Average time required to make an urgent change to the site.
- Goal: Reduce call center costs
 - Example KPI: Average time agent spends handling a call
 - Example KPI: Number of service requests / orders filled via the Web.
- Goal: Increase product sales
 - Example KPI: Average order size via Web.
 - Example KPI: Percentage of sales through the Web.

Ideally, identify KPIs that can be rolled up into organizational KPIs. This can make the project easier for senior management to understand and helps ensure management support.

#2 CMS is Too Complex and Difficult to Use

First generation Web sites were often entirely implemented by IT and comprised of static HTML pages that had to be manually created by technical staff. Removing this "IT

Bottleneck” to Web publishing is often a goal of a CMS implementation. Enabling business owners to manage their own content can make content more relevant and more up-to-date. And IT can redeploy valuable resources to technical projects, while business users manage their own content.

However, if a CMS is too complex and difficult to use, non-technical users will continue to email content to IT staff for Web publishing. The IT people may then use the CMS to publish content. In this situation, even though the CMS is in place, the IT bottleneck remains and therefore, the CM project has not met its goal.

The technical complexity of a CMS can also bog down a project in other ways. Products can be too difficult to implement, resulting in a long implementation time or projects that are never finished. The team might decide to switch to a new product, wasting time and money. A CMS can also be difficult for administrators or not meet corporate application development standards.

How to Avoid Reason #2

Think about usability for the entire team

The first step in ensuring that a CMS meets the needs of all users is to define user roles. Using personas and scenarios specific to the organization can help identify key roles and tasks.

Common roles and requirements are:

Roles	Requirements
Editors: Also called content contributors, authors, or content owners. These users are focused on easy daily creation, editing and publishing of content and are very frequent users, with little or no technical skill.	Editors want to be able to make changes “in context” on the site, to author content in tools they already know like Microsoft Word, and to use intuitive, browser-based interfaces.
Managers: Approve content for publishing. May be the ultimate content owners for a business unit, but aren’t likely to actually be writing the content.	Managers need to complete tasks very quickly without having to complete many complicated steps or to access separate systems (e.g. they want to approve content in a tool they use every day such as Microsoft Outlook or Lotus Notes).
Administrators: Focused on administrating the CMS, managing users and roles, creating and editing workflows. Also charged with ongoing system maintenance, performance, and archiving.	Administrators want easy-to-use administrative interfaces (they’re generally not developers) and a system that scales to handle traffic peaks and high usage.
Developers: Focused on creating templates, designing user interfaces, and writing business logic for content applications.	Developers require that a CMS support common programming languages and standards and integration with common development environments.
Occasional users: May be internal or external to the organization and only use the system once in awhile to manage a small slice of content.	Occasional users need a CMS that is so intuitive that there is nothing to learn or re-learn during each occasional use.

Role definition and requirements should be identified by the cross-functional team at the start of a project and should be top-of-mind as the project moves through its phases from vendor selection to system design to implementation.

#1 What is Content Management?

The term “Content Management” is vague. What does it mean in your organization? Each project participant probably has a different view. The immaturity of the industry adds to potential confusion, with a lack of standardized definitions for commonly used terms. Different vendors will use the same terminology to mean different things. This can lead to vastly different expectations on the part of team members about the overall scope of the project and about how individual features will work. Without proper definition upfront, projects can go wildly off course, at least in the eyes of some stakeholders.

How to Avoid Reason #1

Define what content management means to your organization

Figure 1 provides a high-level overview of the various aspects of content management. As indicated, the actual ‘management’ of content is only a small piece of a larger content management picture. This is not to say that your project must include all of these components to be a content management project. The important thing is to determine what the requirements are for your organization and to define your components, processes, and terminology. That said, it can be valuable to think about the big picture to make sure there aren’t other problem areas that could possibly be addressed by one system. That will help ensure key requirements aren’t missed.

Content Management

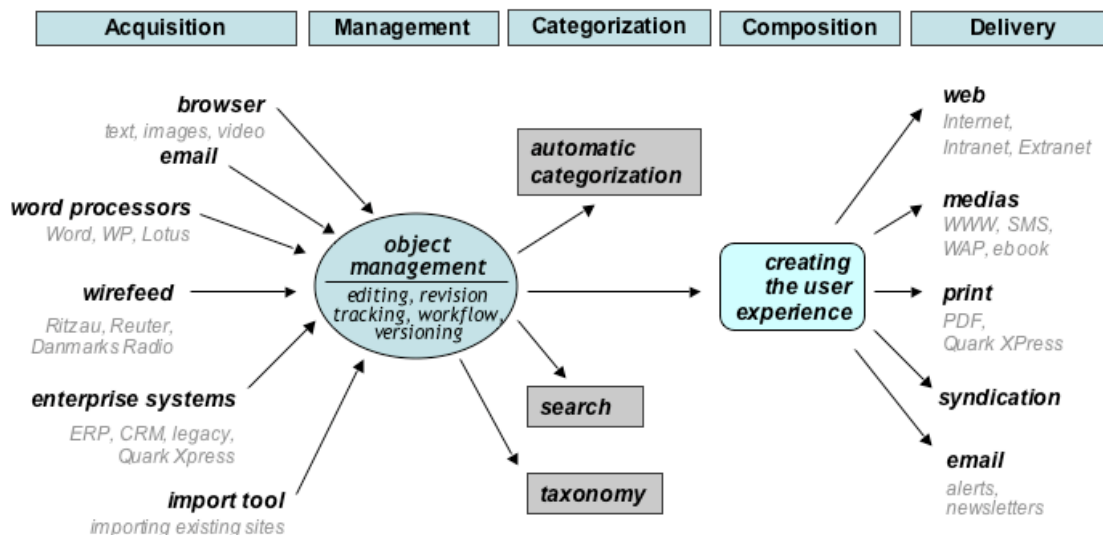


Figure 1 – The lifecycle of content from acquisition to delivery.

As the market for CMS products is still immature, there isn’t always a consistent definition of what a CMS includes. Some products only manage some aspects of the overall lifecycle depicted in figure 1, making it necessary to purchase several products to

complete the picture. It's important to have a crisp in-house definition of content management to ensure you understand what functionality you require in a CMS product.

Create a project glossary

Work to establish a common project vocabulary. It will save time and confusion. Think about terms like *CMS, workflow, template, document, content fragment, roles*, and so on. For this project, what do these terms really mean? Most likely they mean something different for each project player and it gets worse when external resources, like consultants and vendors, are included. A practical approach to defining terms is to ask a key project player to create a glossary with definitions of key terms early on in the process. This needs to be a living document, where terms can be added as the organization learns.

As part of the definition process, consider the differences between content management, CMS, and related terms: enterprise content management (ECM), Web content management (WCM), document management (DM), knowledge management (KM), digital asset management (DAM), and marketing asset management (MAM). It is beyond the scope of this paper to discuss the differences and overlap between each of these. As a convenient starting point, you can download a sample glossary by going to www.fatwire.com and clicking on "Download White Papers". Industry analysts, trade press, and other third-party sources can provide additional material to help define these terms for your organization's needs. Again, the important thing is to ensure team members and the larger organization are working from a common understanding and using these terms consistently.

FatWire Software Helps Ensure Project Success

FatWire Software has worked with more than 450 customers across industries with their successful CMS deployments. FatWire Content Server delivers capabilities that help drive project success.

With FatWire, you can:

Empower Business Users with Easy-to-Use Tools

Content Server includes a range of tools that make it easy for business users (i.e. editors and managers) to manage content. It offers tight integration with Microsoft Word so that non-technical users can publish content using the tools they use everyday. Similarly, they can use Windows Explorer to move content from the desktop to the managed environment. Content Server also offers browser-based authoring and editing that can be done within the context of a site. Easy-to-use tools let managers design workflows and easily complete approval tasks. Business managers can also create content targeting and personalization rules to drive content delivery without needing a programmer.

Meet Developer and Administrative Requirements

Content Server is built on an open and scalable J2EE platform that leverages common Java developer skills and adheres to corporate application platform standards. Administrative tools also utilize user-friendly interfaces. Content Server is built on a multi-site architecture that makes it quick and easy to deploy new sites that re-use or share common elements. And FatWire Satellite Server helps ensure system performance with sophisticated caching and delivery capabilities.

Support the Content Lifecycle

FatWire Content Server makes it easy for users across the organization to contribute, manage, and deliver all kinds of content to multiple channels. Content Server applies enterprise-class workflow to all managed content and leverages common repository and library services across content types. A strong multi-site architecture and standards-based management make Content Server the ideal system for managing content once and delivering it to multiple channels.

Leverage the Experience of Others

FatWire offers FirstSite, a quick-start program based on the experience of many successful customers, which lets you get content management projects off the ground quickly. It includes a collection of standard templates and site components that are common to most sites, combined with documentation, training, a rich developer community, and a best practices methodology. FirstSite helps to immunize CM projects from potential pitfalls.

Put Content to Work

At FatWire, the focus is on more than just managing content. FatWire's solutions enable organizations to put content to work by delivering highly targeted and persuasive experiences to customers, partners, and employees. Our customers are deploying content-centric applications to improve sales, customer service, partner effectiveness, and employee productivity, helping to ensure that CM projects meet company business goals.

About Boye IT / Janus Boye

Janus Boye is the managing director of Boye IT, a leading vendor-neutral content management consultancy based in Denmark. Janus regularly publishes articles in industry publications and is a frequent speaker at industry events. Read more on www.boyeit.dk and www.janusboye.dk

Boye IT
Mejlgade 73
DK-8000 Aarhus C
Phone: +45 86 18 77 30
Email: jb@boyeit.dk
Web: www.boyeit.dk

About FatWire Software

FatWire Software helps organizations put content to work by deploying content-centric applications such as intranets, extranets, portals, and corporate Web sites. Founded in 1996, FatWire has over 450 customers, including J.P. Morgan Chase, Sony, Radisson/SAS, European Space Agency, Nihon Seimei, and Bank of China. FatWire is headquartered in New York and operates offices throughout North America, Europe, and Asia-Pacific. For more information about FatWire Software's award-winning products and services, visit www.fatwire.com or info@fatwire.com