Web CMS Migration Handbook

If your Web CMS is not serving your needs, moving your site to a new CMS may be essential. But a CMS migration is often underestimated, under-budgeted and not well planned. The Web CMS Migration Handbook is a guide you can use to identify the high-level methodology for moving from your current CMS to a new one.

BY DEANE BARKER

blend

Successful CMS Migration

Of all the tasks that make up a content management system (CMS) implementation, there's one that's invariably underestimated, under-budgeted, unscheduled, or even outright forgotten: content migration.

If you're implementing a new CMS, then you often have content—text and files—in your existing CMS system which has to be moved before you can launch.

When someone builds a new house, they never forget that they have to move all their furniture from their existing house. However, when an organization builds a new website, it's somehow all too easy to think that the content magically makes its way to the new CMS. Sadly, it's never that simple.

When planning for a migration, the content manager has to answer three fundamental questions:

1. **The Editorial Question** — what content needs to migrate?

2. **The Functional Question** — how will content in the existing system be managed in the new system?

3. **The Technical Question** — how will the actual bytes of content be exported from the existing system and imported to the new system?

Answering these three questions are the core of any successful migration.

What Content Needs to Migrate?

Fundamental to any migration project is to determine what content is migrating. Remember, the easiest content to migrate is content you discard.

That may seem flippant, but there's almost invariably a chunk of content in any CMS that's no longer necessary. This is content that's redundant, outdated, or trivial.

Any content migration should start with a comprehensive audit of what content exists on the current site. A list of URLs needs to be produced, either by exporting a content manifest from the current CMS, or by crawling the website externally.

Each URL should be evaluated for the following questions:

1. **Is the content still relevant to the overall mission of the website?**
   Web redesign projects usually come from a desire to narrow and sharpen the focus. Does this content still contribute to that focus? Perhaps employee recruitment was extremely important during a high growth period in your company history but is less important now that you've leveled off, or perhaps you've transitioned to using an external recruitment service. Is there a section of pages that are no longer needed?

2. **Is the content still receiving inbound traffic?**
   How many page views has the content gotten in last 30 days? The last year? Expand this focus to the entire section—is a group of content not receiving much traffic. Does it need to? Or is this an indication that this content is less relevant or necessary than you might have thought and could simply be discarded? If you have no analytics to find these answers, it's not too late—even

installing a Google Analytics right now and reviewing data in 30 days would give you enormous insights into what content is used what isn't.

3. **What negative result will occur if the content is simply deleted?**
Sometimes, you might consider asking for forgiveness rather than permission. If your company has 426 news releases stretching back eight years, what would happen if half of them just didn't make it to the new site? Would anyone notice? Would anyone care? Can you just make a command decision that all news releases more than three years old are not getting migrated? If so, would anyone object? Don't migrate purely out of perceived obligation. There are times when content can simply be abandoned.

Clearly, questions like these can be the source of huge disagreement in your organization. Political turf wars are common, as the content represented on the website often defines the very mission of the organization to the public. This content is viewed as a proxy to various organization units— everyone thinks their content is the most important content, and any threat to it is a threat to the humans behind it.

For this reason, executive backing is critical. When two equally well-established personalities in your organization disagree on whether Content X should be migrated to the new website, who gets to make the final decision? To whom can the decision be appealed? Who has the final say?

Be prepared for this process to take more time than you expect. In addition to the occasional political catfight, understand that some decisions will just take time. Some organizations can't do anything without consensus, and plan on waiting for meetings, vacations, and mismatched priorities.

Recordkeeping is important here. Avoid the monolithic spreadsheet. Find a way for the entire team to collaborate on the same data, via a Google Document or some other centralized record.

Better yet, modify your existing CMS to allow recordkeeping alongside the content itself— something as simple as a "Do Not Migrate" checkbox and accompanying filter can help enormously in producing the final manifest of content which needs to be accounted for.

The intended result of this process is a final manifest—a list of content that will be migrated to the new website. Throughout the process of answering this question,

seek to keep this list as small as is reasonably possible. Keeping in mind the ultimate role of the website in serving the mission of the organization, be critical when evaluating content: is it really still needed, and does it justify the migration expense?

"The intended result of this process is a final manifest — a list of content that will be migrated to the new website."

How to Move from One CMS to Another?

Different content management systems simply do not work the same way. Two different systems might have two wildly different paradigms for how content is modeled, aggregated, controlled, and displayed. There is no "Unified Theory of Content Management."

To return to our furniture analogy, not every house is laid out the same way. Rooms are different sizes, power outlets are in different locations, and windows are on different walls. It's naive to think that furniture from your existing house will simply arrange itself in a new house the same way.

So, how will your existing content "fit" in your new CMS? Not only from an information architecture and navigation perspective, but from a functional perspective? Even assuming you can get the content looking the same way to the public, can you replicate administrative functionality like workflows, permissions, and approvals?

This question needs to be answered with the help of your development team. Hopefully, they completed a technical specification for your new implementation which included information on:

1. **Content types** — groups of fields and properties which combine to form content objects (examples: Blog Post, Employee Bio)

2. **Content aggregations** — collections of content objects, grouped together for a specific purpose (examples: Latest Blog Posts, Department Employee Listing)

3. **Contextual content** — areas on global templates with content driven from specific items being view as the main content (examples: Related Posts, localized navigation menus)

Your existing content needs to be evaluated for its ability to support these same structures. Sometimes these transfers are fairly direct if the two systems share architectural concepts. However, if the two systems differ markedly, it can be complicated to determine how the new system will support information structures currently serviced by the existing system.

For example, if both your existing and new CMS platforms support structured content, then it's easier to map properties from one content type to another. However, some CMS operate on a more page-like structure, inside which component is embedded and which comprise the actual content and functionality. Determining how this paradigm will map to or from a highly relational CMS can be difficult.

Navigation can be another problem area. Some systems have explicit navigation structures like menus, collections, and categories, while others use implicit content structure (such as a global content tree) to form navigation. If the two systems differ considerably in this regard, changes might be required in how content is organized, or new structures might have to be created to support the requirements of the new system.

"If both your existing and new CMS platforms support structured content, then it's easier to map properties from one content type to another."

Some Questions to Ask Your Development Team

- **How do the content models of the two systems compare?**
  Can all properties in the existing CMS be supported in the new CMS?

- **How is the content model changing?**
  Are properties being consolidated? Are they being added or dropped?

- **How does the new CMS handle navigation and aggregations of content?**
  Does it use explicit menus, or is the navigation driven by the content structure? How is this different than the existing CMS?

- **How does the new CMS handle global content—things like headers and footers?**
  Are they stored as actual content, or are they stored in configuration files?

- **How is contextual content in the layout handled?**
  If we have items in the headers, footers, and sidebars which change their behavior or content based on the main content being viewed, how does the new CMS recognize this linkage?

- **Can the URL structure of our current site be duplicated in the new CMS?**
  URL structure is important to SEO. If you can maintain the same URL structure in the new CMS, you'll spend far less time creating and managing HTTP redirect rules.

Too many migrations start with an assumption that the new CMS will simply work the same way as the old CMS, which is hardly ever true. Some systems can be very similar in the way content is organized and structured, but no two systems are identical, and changes and compromises almost always have to be made. In some cases, massive restructuring is required to "fit" content into a new CMS.

What are the Logistics for Migration?

Assuming you have solved the first two questions, you now have both (1) a list of content that has to be migrated, and (2) some understanding of the amount of transformation required from that content for it to function properly in the new CMS.

Now, you're left with a question both mundane and deep: how will you actually move content from the existing system to the new system?

This is the "bytes on disk" problem. Beyond logical, domain-related notions of content types and systems, at some point actual bytes on disk have to move from System A to System B. This seems quite pedestrian, but there are dozens of strategies, and hundreds of potential pitfalls.

The first decision is whether to perform a scripted or manual migration. While the idea of people copying and pasting content from one system to another seems low-tech, it can often be the right answer. The answer normally comes down to content volume: a manual migration is fine for a few hundred pages at most, beyond that point, the sheer amount of content will overwhelm a team, and errors will start to creep in.

Here are some decision factors for choosing which path to follow:

- **How many URLs are you migrating?**
  Beyond a few hundred URLs becomes a long haul for people copying and pasting.

- **How much access do you have to cost-effective labor?**
  If you decide to migrate manually, do you have staff available for the work that won't blow the budget? Highly-paid professionals moving content manually is rarely a good idea, but some organizations have access to work study or intern labor, which can keep them on the right side of the budget.
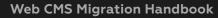
- **How much is content going to change on the way over?**
  If there's considerable transformation and reorganization, then this is easier to do manually because a human can make real-time decisions. If the content is largely going over unchanged, this can be scripted more easily.

- **How much developer support do you have?**
  Do you have a group of developers available to script the import? Or are you without any development resources available? Without developers, scripting a migration is not possible.

- **How code-friendly are both your existing CMS and your new CMS?**
  Scripting requires a way to export via code and import the same way. Do both CMSs have the ability to do this?

If you decide to attempt a scripted migration, it invariably ends up as an iterative process of trial and error. A developer will make tentative attempts to extract, transform, and import content. Content is imported, reviewed, then discarded, and the process repeats, hopefully with increasing accuracy after each attempt.

Additional pitfalls to be aware of:

- **Rich text (HTML) content will often have to be dissected and transformed**, which means parsing the HTML and deleting, changing, or rearranging tag structures to comply with new guidelines and remove obsolete markup.

- **URLs often change with a new CMS, so links between content objects will need to be "re-connected,"** which means finding them all, deriving the target, and changing the link. (This necessarily cannot be performed until all content is in the new system, as the link target must exist to have a new URL.) A redirection strategy needs to be determined, as links to existing content invariably exist outside your organization. When a request arrives a URL that existed in the old system, a lookup and redirection must be performed to deliver the new content.

During the migration process, quality assurance and workflow need to be tracked carefully. Content entering the new system should progress through stages of finality, with various tasks checked off: links corrected, images loaded, text quality checked, metadata correct, etc.

Tracking these activities can often be done in the new CMS itself. Checkboxes and text fields can temporarily be added to new content to allow editors to record the activities performed on that content. The URL of the content in the old system should also be imported and stored as content, both for URL redirection, and to allow easy reference to the old content during QA (it's often even helpful to temporarily load the page from the old CMS in an IFRAME below the new content, for real-time comparison).

Searching for a CMS?

Deane authored a book covering the systems, features, and best practices of web content management. To learn more, visit flyingsquirrelbook.com.

At some point, there invariably needs to be a "content freeze." Once all content is migrated from the old system, changes need to cease while content goes through final QA en-route to launch. This freeze can be stressful as it feels like the organization has one foot in the existing CMS and one foot in the new. Editors feel handicapped while they can't change content, and the QA is working furiously to clear all content and allow the new site to launch.

With luck and planning, all content all clear QA with only minor issues to be cleaned up post-launch. Note that a certain amount of tension is natural, and seemingly no migration ever ends calmly.

In Summary

There are three questions at the core of any successful migration:

1. **The Editorial Question** — what content needs to migrate?

2. **The Functional Question** — how will content in the existing system be managed in the new system?

3. **The Technical Question** — how will the actual bytes of content be exported from the existing system and imported to the new system?

By answering these questions fully and realistically, you can avoid the all-to-common script of a new CMS sitting for months, or even years, waiting for content to migrate.

Understand that migrations are one of the riskiest elements of any CMS implementation projects, for the simple reason that they're overlooked and/or underestimated. Selecting, building, and implementing a new CMS is exciting—it offers a glimpse into the new and possible.

The existing content, however, is often perceived as an anchor from the past. Dealing with it is too often viewed as an unglamorous anchor, which has to be dealt with. Consequently, project planning activities tend to focus on building something new, without regard to the content that has to inhabit the new system once it's launched.

When planning your implementation, be proactive about your existing content and migration. Every decision and planning step should be evaluated in light of what it means for the existing content and how it needs to eventually be migrated. Never underestimate, always provide generous allowances, and make sure every member of your team understands the magnitude of the migration task.

DEANE BARKER

Deane Barker is a veteran of hundreds of implementations, ranging from small marketing websites to massive publishing operations, across nearly every programming architecture and dozens of different CMS platforms. He writes extensively on the subject at gadgetopia.com.

blend

Planning a new project? We can help.

We've got more than a decade of experience helping higher education clients. We are always willing to talk to anyone, especially when it comes to the things we're passionate about. We'd love to talk to you about your project. Or, to have a discussion about content management in higher ed — no project needed.