# The Web CMS Buyer's Guide

INGENIUX

Content beyond boundaries

# Contents

The Web Content Management system has become a foundational technology in your organization's digital experience platform. It empowers you to engage and connect with audiences on multiple devices, channels and venues. It also enables you to cost-effectively manage all these digital channels while maximizing results from digital programs and communications.

Web CMS software is as business critical for most organizations as a customer relationship management system (CRM), enterprise resource planning system (ERP), or any other key business application. Yet most organizations do not have an effective way to evaluate a Web CMS. This is complicated by the sheer number of Web CMS solutions on the market, the industry marketing hype, and the wide range of use cases Web CMS software supports.

**This buyer's guide is aimed at helping organizations understand how to evaluate and buy a Web CMS.** It outlines the key use cases, features, types of systems, and success factors in selecting a Web CMS. While not an end-all guide to the Web CMS market, it does provide a starting point in understanding Web CMS software and a framework for evaluating the merits of different types of systems.

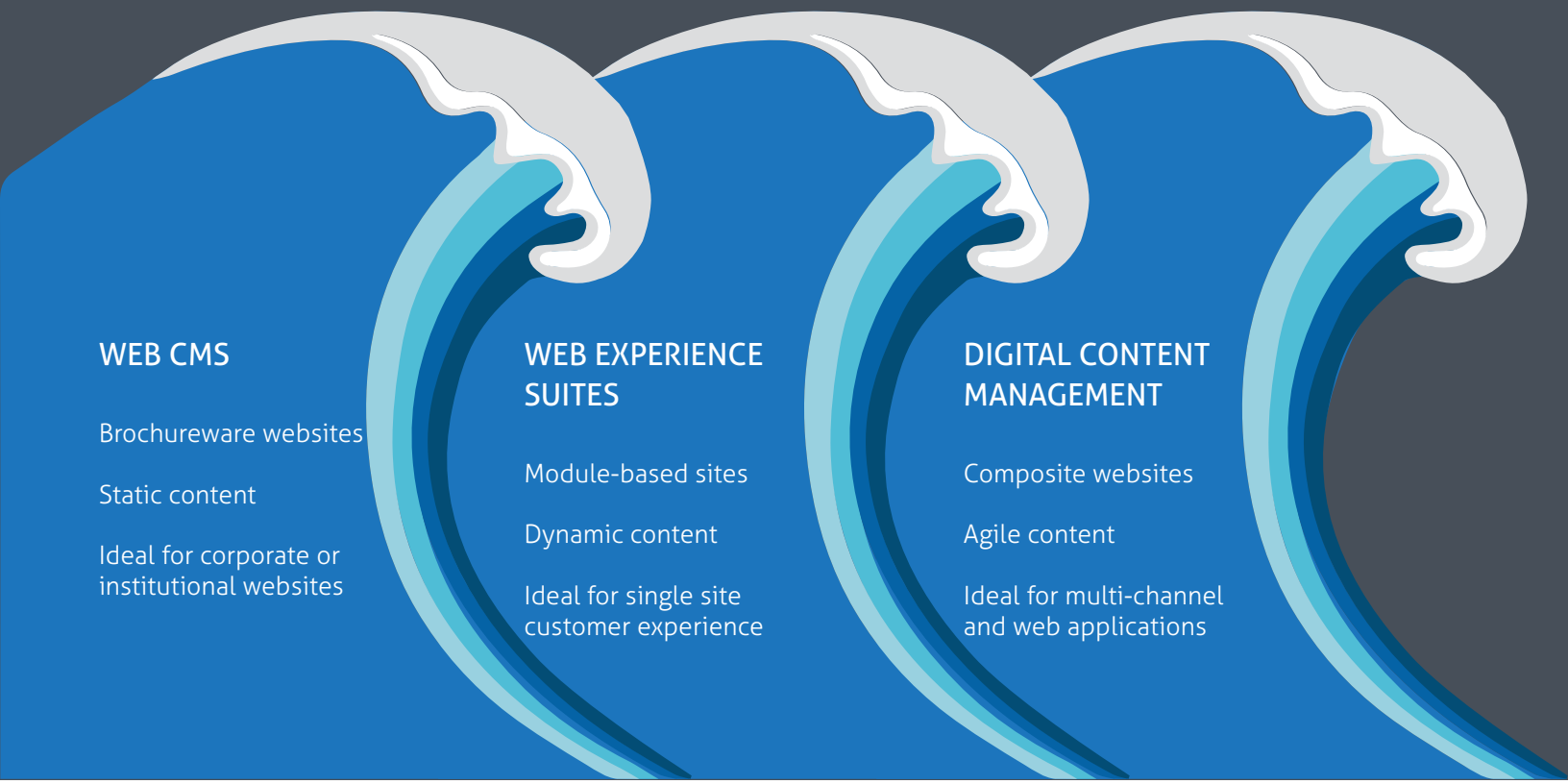
# A Brief History of Web Content Management

The web content management market has come a long way since the days of static brochure ware websites that used basic html and a lot of pretty pictures. This first wave of CMS was heavily product focused with no thought to the engaging with the customer or prospect. It was a good first start, but it was only the beginning.

The second wave of web content management introduced the idea of web experience management and building dynamic websites. During this second wave, large complex web experience management platforms rose to the forefront providing a wide range of capabilities that tied the presentation layer to the backend CMS administration. While this approach worked for some, for many it became complex and difficult to manage. New publishing channels started to appear – like mobile and email – and it was increasingly difficult to support multiple publishing channels on a single platform.

**The CMS must become the platform for content creation, governance and orchestration, but not always the platform for presentation delivery. The key is to not get locked into a platform that can't support current and future requirements.**

# The Third Wave




We are now in the third wave of web content management and the website is one of many publishing channels the Web CMS supports. **The third wave is about building smarter content that can adapt to different channels and audiences.** It's about new ways to deliver content without the overhead of a traditional Web CMS. Content is mobile first, intelligent and readily available for any website, campaign, app or device.

This shift to the third wave of content management changes how you define your requirements for web content management. The CMS must become the platform for content creation, governance and orchestration, but not always the platform for presentation delivery. The key is to not get locked into a platform that can't support current and future requirements.

# Digital Content Management in the Third Wave

When you approach your next digital project, look at it using this lens:

1. **Know every experience is different:** You can't deliver every experience the same way, using the same approach. A flexible CMS platform can adjust how you deliver content, whether it's delivering secure content via an API to your custom-built websites and web applications, or it's dynamic content delivery to multiple front-end websites or communities. You need the ability to apply the right delivery approach for each project.

2. **Think broader integration using agile methods.** The experiences you create pull information from a variety of best of breed marketing and business applications. The ability to connect these disparate systems and integrate their information to create the right experiences is critical. Look for solutions that are going to support connecting your key systems and data.

3. **Demand less complexity.** Designing engaging experiences is hard enough; your CMS platform shouldn't add to that complexity. It must be easy to use, and not get in the way of delivering the user experience you are seeking.

4. **Expect your information is secure.** Your CMS must deliver the customer experience you want, without putting your data and business at risk. Understand where your customer data is stored. Ask where the database is located, how the cloud architecture is secured, and what steps are being taken to ensure your website and data is protected.

# What is a Web CMS?

A definition of Web CMS may be summarized as, “Software that allows non-technical business users to manage content throughout its entire life cycle.”

This definition is about the only thing that many Web CMS solutions hold in common. As this Buyer’s Guide illustrates, there are a wide range of Web CMS solutions and use cases. There is also a wide range of terms and acronyms used to describe a Web CMS – from customer experience management (CXM), to web experience management (WEM), to digital experience management (DX) and content management systems (CMS). This Buyer’s Guide uses the term Web CMS to encompass all of the solutions named above.

# How to Buy a Web CMS

First and foremost, buyers should understand that there is no best Web CMS. There is only a best Web CMS relative to buyers' requirements.

It is important that buyers thoroughly understand what their requirements are, based on every user contingency within the company. That understanding should then be mapped to an equally well-defined understanding of currently available Web CMS platforms.

This mapping is more easily said than done because companies often have difficulty understanding exactly what their users need, as well as what the different Web CMS platforms actually offer. **The guiding principle in the purchase decision should be as close as possible to a 1:1 match between what a buyer needs and what a product offers.**

## Use Cases

A practical approach to testing the match between requirements and solutions is to define use cases. That is, how do those involved in the content life cycle interact with the Web CMS? How do they use it? What are their goals? Defining your key use cases is an essential first step in evaluating a Web CMS.

## Budget

Before making a decision on what Web CMS solutions are feasible, buyers must know what purchase resources are available. It makes no sense to test a $200,000 Web CMS license if only $75,000 is available. Similarly, even when funds are available, it makes no sense to overspend. Once the Web CMS budget is known, buyers should build the business case both to justify the expense and to ensure that money is not being wasted. In this activity, it will be important to get an idea of what the total cost of ownership (TCO) of any potential Web CMS will be over a three to five-year period.

## Total Cost of Ownership (TCO)

TCO is determined based on the deployment model of the CMS. Web CMS software is available in several different pricing models. The traditional model is based on purchasing licenses of the software and installing on premises or in the organization's private cloud. Along with on premises installations, many Web CMS also now offer either a Software as a Service (SaaS) version of their CMS or are SaaS-only. The approach to determine TCO is different for each deployment model.

## On Premise Hosting

CMS Licensing relates to the amount buyers pay to purchase the application for its particular configuration (number of servers, content editors, bandwidth, etc). Typically understood as the "price" of a Web CMS, the license cost over several years typically represents one quarter to one third the TCO. The cost of implementation is often equal to or slightly greater than the cost of the license. For a base license of $75,000, implementation costs typically range from $60,000 to $90,000 – or roughly 80 percent to 120 percent of the license cost. Buyers should be aware that implementation costs vary widely between Web CMS products.

## Annual Support and Maintenance

For technical support and maintenance, which includes product upgrades, buyers should expect to pay 20 percent of license costs annually. For a $75,000 license, $15,000 annually will be spent on support and maintenance.

## Administration and Support

IT resources will be required to administer the system over time. Buyers should attempt to estimate how many full-time employees will be required to do this, and they should calculate the burdened cost of such resources as part of the TCO of the Web CMS.

## Software as a Service (SaaS)

SaaS solutions are hosted outside the organization on the vendor's infrastructure. SaaS-based CMS pricing is generally a recurring subscription fee, either monthly or annually, that includes software licensing, support and maintenance, and software hosting. Because SaaS includes all of the application management and hosting services, the TCO of SaaS offerings tends to be 50 to 60 percent less than on premise costs. Some SaaS-based CMS solutions are multitenant, meaning customers share a common version of the software and common pool of resources.

**Most SaaS-based vendors, however, provide single tenant, managed instances of the Web CMS software providing customers the benefits of SaaS and cloud computing, while at the same time ensuring the security of the customer's environment.**

## Product Usability

While not commonly considered a factor in TCO, product usability correlates directly to user adoption rates. When products are not easily learned and used by content contributors, a common result is implementation abandonment. This can result in a complete loss of all resources invested in the project. A less dramatic impact of poor usability is the increased time – and therefore expense – required to create, manage, and re-use content.

# Success Factors

## Organization Buy-in

Critical to the success of any Web CMS purchase decision is buy-in throughout the organization. Especially important is top-down support, where the critical nature of managing content effectively is understood throughout the company. Yet it is equally important for content contributors to make their requirements known. In short, every level of the organization needs to buy into the critical nature of the business case for a Web CMS.

As an integral part of the business case, Web CMS buyers should strive to articulate as clearly as possible how the Web CMS will allow them to engage customers online. **The bottom line always comes down to how relationships with customers can be improved.** In the case of Web CMS, this is normally the online interaction. Improved branding, customer loyalty, content personalization, and multichannel management all contribute to a richer experience, and therefore to improved revenues – either directly or indirectly.

## Usability

The importance of Web CMS usability cannot be overstated. With it, implementations have a chance of success. Without it, they do not. Employees who depend on a Web CMS to do their jobs simply will not put up with applications that impede their efficiency. If the Web CMS slows them down or impedes their progress, they will simply work around the system. This inevitably leads to implementation failure.

## Customer Support

Excellent customer support is essential to the timely resolution of critical issues, which can dramatically impact company profitability. To take but one example, system downtime at the wrong time can dramatically compromise the web presence as a whole – think of the marketing campaign that can't be viewed for the first several hours after a press release, for example. If the Web CMS vendor does not offer 24/7 customer support, or if technical issues cannot be resolved quickly (within several hours), marketing, sales, and corporate communications overall can be

impacted. Competitors will be happy to take customers. Excellent customer support is essential to the timely resolution of critical issues, which can dramatically impact company profitability.

## Time-to-Market

The time-to-market, or speed of executing sales and marketing campaigns, is also critical to success. Gone are the days when one week was a satisfactory time-to-market for an online campaign. Now it's more like 24 hours. For that to be a reality, the Web CMS must perform well in a number of categories, usability and web engagement foremost among them.

## Vendor Roadmap

It's important to understand the plans the vendor has for developing its Web CMS system. While most of your use cases should represent existing and near future requirements, some of them may be future based.

**An established roadmap can demonstrate future capabilities that align well with your future use-cases.**

## Vendor Stability

Finally, all of the considerations above become moot points if the Web CMS vendor you choose isn't fundamentally stable. **Buyers should ask themselves how vendor strategies and market presence will contribute to the vendor's continued success in the market – or even existence – over the next few years.** However, it is important to note that traditionally the independent Web CMS vendors have been the most stable in the market, while larger portfolio software companies have waxed and waned on their support of Web CMS applications. A small company solely focused on Web CMS may have much more stability compared to a larger company that offers a Web CMS as a subset of a larger solution line.

# Web CMS Features

In our opinion, the distinguishing features of current Web CMS systems are in the value-added layers that some solutions provide.

Web CMS applications have been around for over 15 years. As such, the technology is relatively mature and has become the foundation for digital experience in most organizations. While Web CMS has not yet become a commodity, it is true that many of the features and functions within Web CMS's are common in offerings throughout the industry.  Such standard features include WYSIWYG authoring, visual workflow, and template creation. So, what should buyers of a new Web CMS look for? In our opinion, the distinguishing features of current Web CMS systems are in the value-added layers that some solutions provide, such as:

## Product Usability

The primary purpose of a Web CMS is to empower non-technical users to manage the Web, removing IT resources from common usage scenarios to reduce costs, maximize resources, and improve web content management operations. The user-friendliness of the Web CMS is therefore essential in the decision-making process. Consider the ease with which a marketer can create an online promotional campaign or a casual content contributor can create new, template-based web pages.

## Technical Foundation

The languages and technologies with which a Web CMS is developed is another important consideration in the evaluation process. This will define the toolset that can be used to customize a Web CMS to specific customer needs. It will also define the capabilities you will need to support the application, the type of staff you need to hire and manage, and, to some extent, the roadmap for your website and applications.

If your web team has a background in Java development, a .NET solution may not be a good fit. If you do not have in-house technical capabilities, you may be best served by a SaaS-based solution that includes application management.

From an operational perspective, scalability is important both as a Web CMS is integrated throughout the business and as seasonal spikes in user activity put bandwidth/hardware strains on the system (during holiday seasons, for example).

## Flexibility

Web CMS applications in the market vary widely in their architectural flexibility. Most were developed incrementally over the course of many years, and their development standards have not been consistent over time. Generally speaking, it is difficult to integrate a Web CMS that evolved in this manner with other enterprise applications, or even to extend their out-of-the-box feature functionality.

**The most flexible Web CMS systems are those that were conceived as a whole, developed purely within one language based on a well-articulated set of industry standards, and built in accordance with the principles of services-oriented architecture (SOA),** meaning the components of the platform are designed to operate as self-contained units that can be added and removed from the system at will.

## Web Experience Management

For many, the goals of Web CMS implementations focus on managing a customer's entire web experience. These goals commonly involve marshalling all channels of customer communication – particularly web and mobile – for the purpose of crafting a stream of engaging, interactive, user-specific, carefully coordinated customer communications.

**Web experience management (WEM) measures a Web CMS's ability to deliver these optimized online experiences based on the interests of particular website visitors.**

Web experience management (WEM) measures a Web CMS's ability to deliver these optimized online experiences based on the interests of particular website visitors. This is done by assembling the right content and delivering it at the right time, in the right context, based on the match between permutations of available content items and the website visitor's interests.

Over time, a good Web CMS helps to build detailed profiles of users, based on pre-existing explicit data and dynamically-

captured online behavioral patterns. Often these user profiles are managed in a separate system, such as a marketing automation system, or CRM, and then leveraged by the Web CMS in delivering targeted content. In this way, a Web CMS is good at managing not only the content and the user, but also the interaction between these two and, consequently, the experience itself.

## Headless CMS

Headless CMS, sometimes also called "content-as-a-service", or simply CaaS, is a strategy for delivering CMS managed content to Web applications and other channels. Many organizations develop business applications that don't live within the CMS environment but require some level of content management. Native mobile apps, and business applications such as a loan application are examples of business applications that need some content management.

With a headless CMS you manage and edit your content in the CMS. When the content is ready to be published, push it as resource file to an external application, or the application can request content using an API-based web service.

Headless is fast becoming a popular approach to content management, but there are also other approaches close to the pure headless model including. In all cases, the web presentation layer can easily support web applications and CMS content without complex integration.

## Digital Marketing

Organizations look for the availability of built-in marketing capabilities and/or tight integration with marketing tools such as CRM, marketing automation, analytics and so on. Landing pages, personalization, email marketing and social media marketing integration are digital marketing capabilities that require integration with the Web CMS system.

## Multilingual, Multi-site

For those conducting business across multiple geographies, the ability to manage multiple sites, in multiple languages, delivered via multiple channels will be a key success factor.

# Development and Administration

The robustness and quality of development tools (often an SDK) dramatically impact how easily and quickly IT can augment system functionality. A Web CMS's ease-of administration – the resource intensity required to maintain the software over time – dramatically affects the total cost of ownership. As an example, the ability to avoid adding even one additional IT person can amount to well over $500,000 in savings over three to five years.

Far more helpful than a feature-functional checklist comparison, serious reflection on these evaluation criteria, as applied to specific use cases, will help buyers distinguish between the various Web CMS offerings on the market.

**Far more helpful than a feature-functional checklist comparison, serious reflection on these evaluation criteria, as applied to specific use cases, will help buyers distinguish between the various Web CMS offerings on the market.**

# Web CMS Use Cases

To delve more deeply into use cases for Web CMS we will look at five of the most common use case scenarios: public websites, corporate intranets, extranets, community-based sites, and mobile sites.

## Public Websites

Public websites commonly encompass:

### Digital marketing (brand awareness and lead generation):

This feature of public websites is common to most companies offering online goods, services, and information. How does an airline get web visitors to purchase tickets from them versus a competitor? Web CMS's will commonly have digital marketing and lead-generation tools built in that allow marketers to increase the speed of creating marketing campaigns, incorporating search engine optimization, lead-generation capabilities, analytics capabilities to monitor user activity and determine what visitors are interested in, and so on.

### Product-based sites (e-commerce):

Critical to product-based sites is the ability to structure and categorize content so that inventory can be effectively merchandised and displayed in different sections of the website (for example, an air-conditioner sold in "Appliances" and "Heating and Cooling"), and across channels. Also important on this type of site is the ability to combine elements of product descriptions. Here, information such as product SKUs, bar codes, descriptions, user ratings, images, availability, pricing, manufacturer information, warranty information, related products, shipping information, availability in nearby stores, etc. become the critical functional building blocks. In addition, the ability to offer content that isn't product specific but demonstrates an understanding of the user's needs and challenges and the market is required (this is often referred to as "content and commerce").

### Member services-based sites:

This contains a fairly broad range of use cases, but generally requires users to log in for access to services based on tiered service level agreements. Members may have access to one part of the site but not others. Alternatively, they may be entitled to certain benefits based on their status level or tier level, and may not have access to certain information beyond that level. The key for this functional area of public websites is that services and access to content be based on users' membership levels.

### Federated websites:

Large organizations such as universities, government agencies, non-profits, and corporations often manage a significant number of loosely connected websites. These "federated" sites are most easily understood as multiple instances of the four use cases described above, with each department of the university or organization running one or more of the aforementioned scenarios.

## Corporate intranets

Corporate Intranets are the next use case for Web CMS implementations. While not necessarily different from public-facing websites in terms of features or functions, intranets do tend to share a common set of unique features and functions, which include:

### Human resources service portals:

The most common use case for the corporate intranet, the HR service portal disseminates any and all relevant corporate information to internal constituents. This information includes corporate policies, benefits, pay information, access to employee retirement accounts, stock options, payroll and tax information, medical benefits, etc.

### Employee enablement tools:

Another typical use for the corporate intranet is the provision of employee enablement tools, as demonstrated by an intranet sub-site for the sales force that includes training materials, marketing collateral, price lists, competitive analysis, SLA directives, etc. Generally speaking, the goal of the corporate intranet is to expose all corporate information in order to enable employees and disseminate that information.

## Extranets

Extranets are analogues to the corporate intranet that extend their functionality to partners and customers. In this scenario, people outside of the company have log-ins that gain them access to certain information within the content repository. Content access control is applied either at the repository or database level. Extranets also allow individuals outside the firewall to join internal workflows.

Taken together, public-facing sites, intranets, and extranets, are enabled by the core features of the web CMS. They are simply different permutations of the features exposed to different audiences.

## Digital Experience Portals

Digital Experience Portals are an essential platform for building and supporting key audiences and communities. A portal is simply a secure website that delivers content, applications or online services, collaboration and engagement.

You use a digital experience portal for any use case that requires secure access to information, collaboration, or explicit personalization based on the user profile. Common use cases are customer service portals, company extranets, partner or distributor portals, membership portals and communities, purchasing portals, student or academic portals, or employee portals.

Today's portals differ from traditional portals because they connect key information and systems to provide content-driven experiences within the portal directly. They not only offer the ability to integrate information from other repositories and applications, but they include complete content management capabilities that support a wide variety of needs.

A Digital Experience Portal is not a point solution, and it's more than a database of content. But it's also not an out-of-the-box solution that you can quickly plug and play. The very nature of unifying multiple sources of content, integrating multiple applications, modeling unique business processes, and engaging users with your brand pretty much excludes "out of the box."

Although each type of portal supports a unique purpose, most have a number of common features including dashboards and home pages, built-in content management, integration with backend systems such

as CRM, ERP and Marketing Automation, document storage, collaboration, social and community features, analytics, mobile access, member management, SSO, Permissions and Access.

## Mobile Websites

Mobile websites, including mobile and other devices, such as tablets and kiosks, are a distinct use case for many considering a Web CMS implementation. Mobile websites can support all of the use cases listed above, as well as native mobile applications. Generally, a mobile website will deliver a task-based user experience, support optimal layout on a wide range of devices, and feature multi-touch interactions for tapping and swiping elements of the websites. Mobile websites may also take advantage of device capabilities, such as local storage, cameras, GPS and other features.

There are two ways your CMS supports mobile websites: responsive design and adaptive design.

Responsive design is the most common approach to mobile web development and it's rare that a Web CMS doesn't support it. It refers to flexible web page layouts that automatically adapt to the size of the viewing device's browser.

Cascading style sheets (CSS) are the foundation for responsive design, empowering designers to manage the presentation layer independent of the content. Web developers can scale fonts, resize graphics, and even change layouts based on the size of the screen.

Adaptive design detects the device you are using and creates a unique visual experience for that device size. So instead of creating one design and rendering elements responsively based on CSS media queries, with adaptive you create a set of designs, each one supporting a different device, or set of devices.

Responsive design allows you to deliver one set of content with one design to a website whether it's viewed on a desktop, tablet, or smartphone; as a result, it is the predominant approach to mobile-enabling your web experience. However, if your web experience supports a range of devices such as different sized smartphones and tablets, wearables, kiosks, game consoles and more, an adaptive design approach may be a better option.

# Types of Web CMS

In considering the type of Web CMS that is right for your business, there are five key comparisons to keep in mind: publishing systems vs. development frameworks, database vs. file-based systems, software-as-a- service (SaaS) vs. on premise platforms, and open source vs. commercial systems.

## Publishing Systems vs. Development Frameworks

In making a decision for a particular Web CMS, perhaps the most important decision is whether you need a publishing system or a development framework. Publishing systems are characterized by the ease with which they allow for the creation, management and deployment of content. Ideally, publishing systems are ready to use out-of-the-box, have short implementation times, and make managing the content life cycle simple at each stage.

**Much of the heavy technical lifting in publishing systems is handled by the Web CMS itself.**

Much of the heavy technical lifting in publishing systems as we describe them here is handled by the Web CMS itself. Publishing systems are also characterized by the scope of requirements they address, which typically begin with content authorship and continue through workflow approvals, publishing, re-use, deployment and archiving.

**Development frameworks, on the other hand, provide a toolbox with which companies can craft their own solutions.** These frameworks place the emphasis on the ability to create highly specific customized solutions, which typically include a complex content staging topology, heavy involvement of development resources for system extensions and the associated need to manage the logic surrounding content at each stage of system extension or integration.

Simply put, publishing systems handle an organizations' need to create, develop and deploy websites, whereas development platforms provide an environment in which any number of solutions can be created, publishing among them. We find that most buyers seeking a Web CMS are seeking a publishing system rather than a development environment.

## Traditional Coupled vs Decoupled vs. Headless vs. Hybrid

There are several types of web content management systems available and the one you choose depends on your business requirements. A CMS typically falls into one of four categories: couple, decoupled, headless and hybrid.

In a tightly coupled CMS, both the backend CMS and the delivery tier exist on the same platform, often tightly integrated. In this model, there is typically a templating system that defines the structure of the website. With administration and delivery on the same platform, you are required to have components of your backend CMS installed on the web servers that deploy your website.

With a decoupled CMS, the delivery tier is separate from the backend content management system, sometimes written in a completely different language or technology. In this model, you can deliver content to any delivery format you want using via a content API. The decoupled model supports the creation of content in one place and publishing it to multiple locations: a website on the CMS, a website on another platform, a web-based application, a mobile app and so on.

Headless CMS is often interchanged with decoupled because it too separates the create of content from its delivery. It provides some type of content API that other applications can connect with to pull content from the CMS.

The different between decoupled and headless is simple. With a headless CMS, the delivery environment PULLS data from the CMS via an API when it needs it. With a decoupled CMS, data is PUSHED to another system, sometimes through resources files or HTML files and sometimes using an API.

With a decoupled CMS or a headless CMS, you can easily create and manage content for highly customized websites and rich web-based applications. These websites and

apps may take advantage of new client-side JavaScript frameworks such as Backbone, Ember or AngularJS.

Finally, a hybrid CMS supports more than one type of publishing model, tightly coupled, decoupled and or headless. Most companies have multiple publishing needs supporting different channels or digital experiences. A Web CMS with a hybrid publishing model can support a variety of needs including managing the corporate website within the CMS system, publishing content to a mobile apps via content API, and publishing help content to an online business application via resource files.

## Relational vs. File-based Database Systems

The next consideration in choosing a Web CMS is whether a system is relational database or file-based driven databased (NoSQL).

Relational database-driven Web CMS's require that content be stored in a highly structured way. Each piece of content is essentially stored in a matrix (rows and columns), where each row pertains to a piece of content, each cell (column) contains a "content chunk," and may also have related metadata. The content chunk becomes the most granular level at which content is processed. The feature-functionality of the matrix itself represents the limit of what can be done, not only with content within the matrix, but also with integration in the overall technology infrastructure.

In a JSON document store-based, NoSQL database, content is stored as documents. A document contains a set of key-value pairs, describing all the information that makes up a piece of content. Because all the content related to a record is stored in a single document, changes to the content can be made quickly. In addition, if the data model needs to change, only the affected documents need to be updated – there's no schema update required and no database downtime necessary to make the changes. Content managed by a NoSQL Web CMS

**File-based databases like NoSQL more accurately represent the way enterprises typically think about their content**

is highly structured. Structured content is content stored in a format that defines and describes it. This means you create content in such a way that there is no relationship to how it's delivered on a webpage, but it is structured with tags and categories and contains metadata about the content itself.

We find that file-based databases like NoSQL more accurately represent the way enterprises typically think about their content – that is, Word documents, PowerPoint presentations, Excel spreadsheets, PDFs, and the like.

## Software-as-a-Service (SaaS) vs. On-premise Platforms

The next key consideration in choosing a Web CMS is whether it is installed on premises or whether the application is offered as a service (SaaS).

Traditionally, buyers have thought of on-premises software as offering a richer set of features and functions than software offered as a service. However, this is no longer the case. The feature-functionality between these two types of Web CMS's has converged. The purchase decision now focuses more on the extent to which companies wish to customize their software, which would favor on premises solutions, or the extent to which they value automatic version upgrades and seamless system administration, as in a SaaS platform. These are generalities, and they will not exactly describe every prospective buyer's decision criteria, but as the need to control all of the minutiae within a Web CMS solution increases, the probability of favoring on premises software also increases.

The types of SaaS approaches have also changed. Traditionally, hosted applications have used a multi- tenancy model where multiple customers are hosted on the same servers and applications. This model, pioneered by companies like Salesforce.com, is a cost effective way for software vendors to add new customers and provide a robust platform at a lower

**With the advent of virtualization, Web CMS buyers now have an option of using a Web CMS managed in the cloud with the same extensibility as an on premise Web CMS application.**

cost of ownership. However, not all SaaS companies have the resources of Salesforce.com and not all applications fit the same model as a Web CMS.

With the advent of virtualization, Web CMS buyers now have an option of using a Web CMS managed in the cloud with the same extensibility as an on premise Web CMS application. Virtualization provides a complete standalone network that can run a Web CMS and integrate securely with other systems and applications. As the cost of virtualization has fallen dramatically with the rise of cloud computing, the costs between a dedicated virtualized environment and a traditional multi-tenancy Web CMS have become about equal. Virtualization also provides more flexibility. With a Web CMS application and website, you can move to other hosting locations or bring hosting in-house, have file-level access to servers, open secure ports for other applications, perform upgrades when you want to rather than when the vendor requires downtime, and scale the service to offer better performance.

## Open Source vs. Commercial Systems

While no one would argue there aren't some differences between open-source and commercial software, there is very little difference between the cost and quality of the software today. The cloud completely changed open-source and commercial Web CMS software. Open-source Web CMS providers are providing SaaS licenses, and commercial vendors are abandoning expensive upfront licenses, support and maintenance agreements.

Enterprise open source requires implementation, maintenance and operational costs and most businesses ultimately end up buying enterprise open-source licenses and support, the same as they would for commercial Web CMS.

Today "open" means more than open source. Open means that a platform can easily be extended and integrated with other applications. Web services, frameworks and APIs are application specific, not license-model specific. If you look at any commercial Web CMS application, there are many source components and libraries. Commercial

Web CMS companies have the resources to develop the code, provide quality assurance and deliver professional support and maintenance.

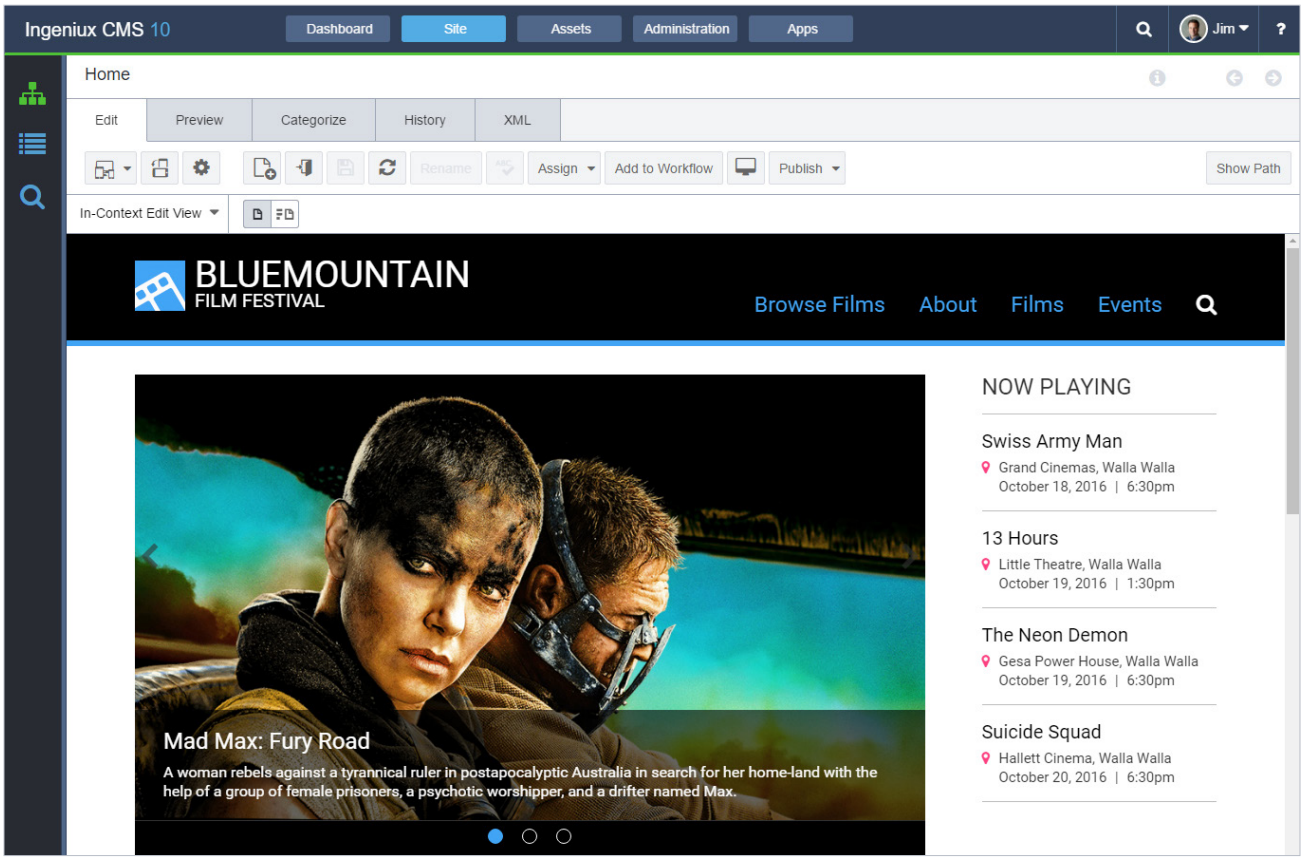
There has been a convergence between open source and commercial. **Open source is not a philosophy or a business model: open source is a software feature.** Whether the feature of modifying source code is important to you depends on your requirements.

Selecting Web CMS software requires balancing a lot of considerations: software features, viability and support model of the vendor, total cost of ownership, capabilities in your company and your business strategy and growth expectations. Success takes investment. You will pay for your software whether you use open-source or commercial applications.

# Key Web CMS Features and Capabilities

While the features and capabilities vary widely between Web CMS platforms, there is a core set of feature-functionality that all prospective Web CMS buyers should include in their requirements.

This section is not intended to be a complete list of Web CMS capabilities, but rather highlights many of the essential features of Web CMS applications, the absence of which would be a deal-breaker.



## Content Authoring

Web CMS systems should provide a content authoring environment that is easy for everyone to use, technical and non-technical. The Web CMS should provide a WYSIWYG content editor, which is essentially a word processing-like application that runs inside a standard browser. The user interface of WYSIWYG editors is very similar to word processing applications in which users can select spacing, alignment, text attributes, hypertext links, colors, images, spell checking, etc. The Web CMS should also provide the ability to edit content in-context of the page preview as well as in a forms-based view that supports additional structure and metadata.

## Browser-based Clients

Browser-based clients refer to an application running inside a browser, as opposed to an application that needs to be installed on the desktop. Using a Web CMS with support for browser-based clients is important in order to maximize the number of potential content contributors. Assuming that everyone has browser access, browser-based clients ensure that everyone can use the Web CMS.

Whether a client is browser-based or not, it is important to understand which web browsers the software can run in (Microsoft Internet Explorer, Apple Safari, Mozilla Firefox, Google Chrome, etc.) as well as the supported operating systems, such as Microsoft Windows or the current Apple OS.

## Content Re-use

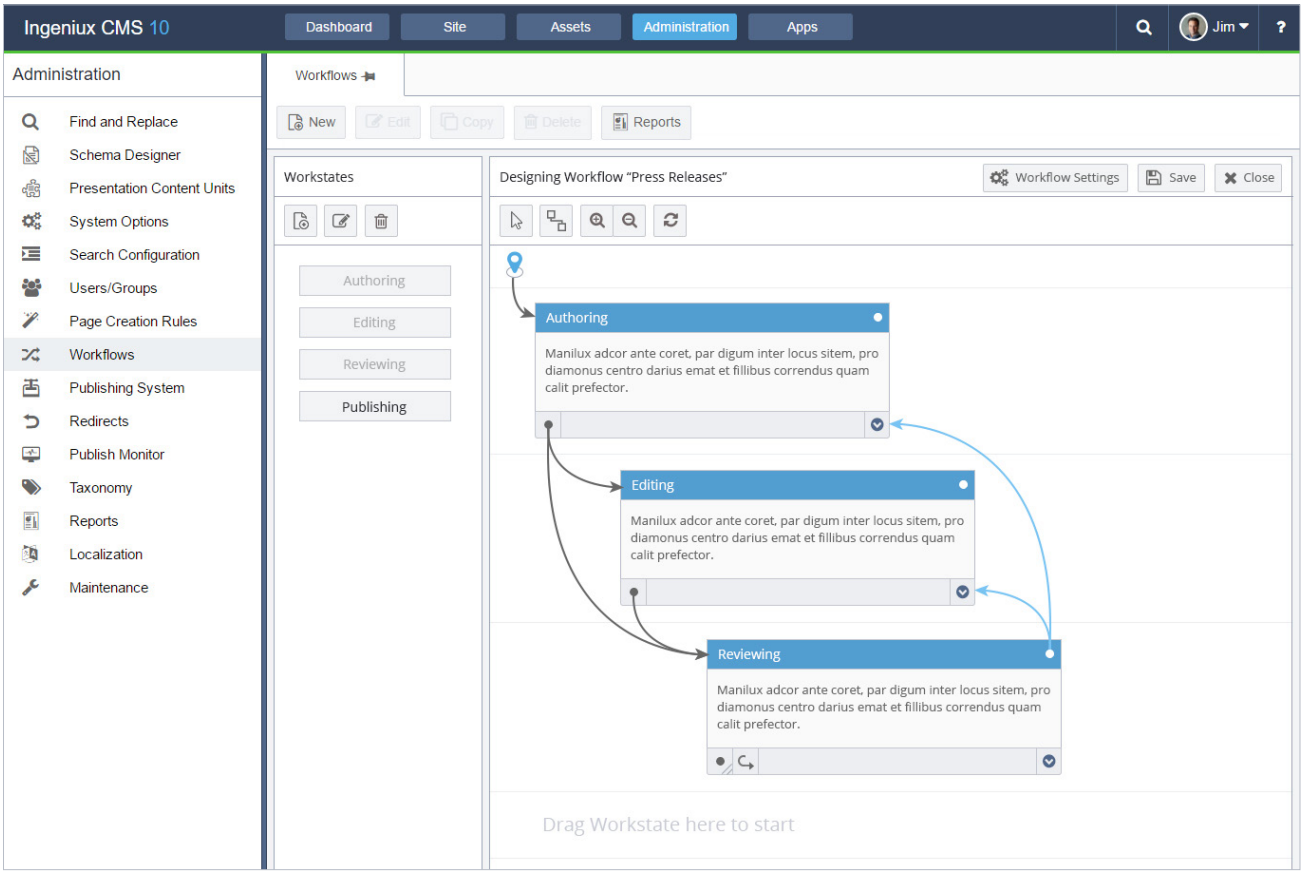
**Content re-use is often considered the number one factor for return on investment of a Web CMS.** Good Web CMS's ensure that source content is available centrally, and when changes are made to the source, updates are disseminated to every instance where the source content is used. For example, if a company changes the location of its headquarters, updating the corporate address inside the Web CMS would ensure that every time the address is published on the website – or in printed materials, or anywhere else – the updated address is used.

Some Web CMS's are based on XML, or similar technology, and are developed to meet a wide range of content re-use scenarios. This may include exports for elements such as titles, abstracts, and

thumbnail images in order to create content indexes and navigations, the ability to support multi-channel content deployment to desktop web, mobile, print, and kiosk “channels,”, and the ability to re-use content across multiple websites.

## Workflow

All Web CMS’s should provide for the assignment of content authoring by individual or role. That is, a supervisor may decide that Joe Smith should author an article, in which case a task is sent to Joe Smith’s email inbox. Alternatively, the same supervisor may decide that anyone with the title of marketing manager should author the article, in which case the workflow would send the assignment to anyone with the title marketing manager. Once the task is accepted and content is authored by the appropriate individual, the article is sent back to the supervisor for approval. Workflows may include any number of approval stages and, once complete, content is published to the appropriate



destination. Workflows may also include individuals outside of a company, such as business partners or content translators. Non-technical users should be able to set up workflows without the involvement of IT. This means that workflow tools should be visually-based and not require coding.

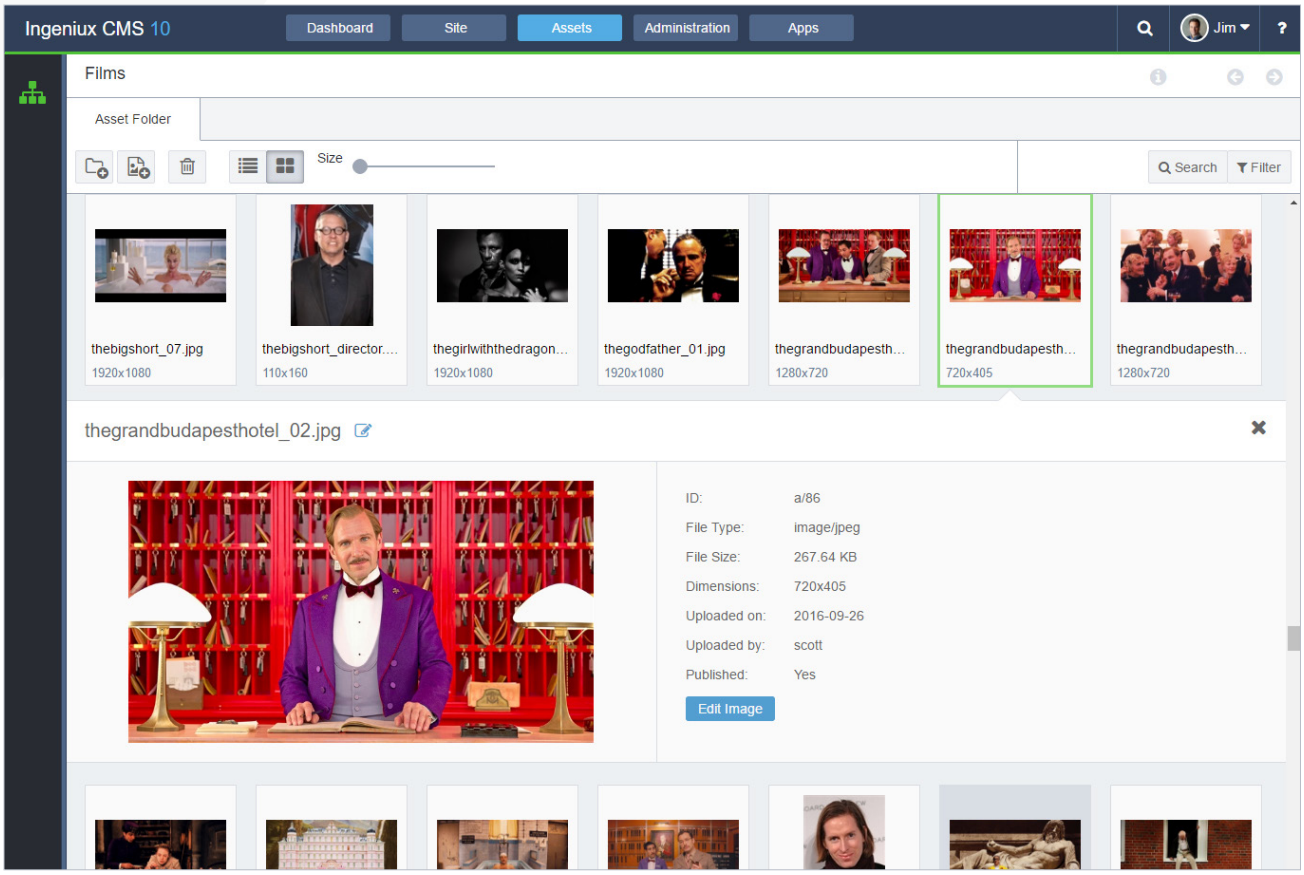
## Versioning

As documents, web pages, and other files are updated over time, the Web CMS should maintain all previous versions, and allow for version restoration, or "rollback," at any time. In addition to the simple maintenance of previous versions, the Web CMS should indicate who was responsible for every change. This ensures that previous versions are always available, and also allows for accountability. Many Web CMS solutions also provide red-lining or version comparison capabilities that highlight the changes to each version of the page. Version history is critical in industries such as financial services, health care, and life sciences that have compliance and reporting requirements. These organizations generally need more sophisticated audit trail capabilities that show who made content changes and track which versions of the content have been published.

## Taxonomy

A Web CMS should maintain a standardized taxonomy, which is essentially a lexicon that defines what terms mean and their degree of relevance to other terms. Taxonomies allow you to organize and deliver content by category and make it possible for users to search for specific content without knowing the exact term used by the content author. For example, if a user searches for "lawsuit" when the original author used "litigation," the taxonomy would establish relevance between the two terms that would return content for "litigation" based on a search for "lawsuit." Taxonomies are especially helpful in conjunction with metadata and can be critical in managing customer experience through segmentation and targeting of key audiences, guided or faceted navigation, and better search experiences.

**Taxonomies allow you to organize and deliver content by category and make it possible for users to search for specific content without knowing the exact term used by the content author.**




# Digital Asset Management

The Web CMS should manage all types of digital content, including images, video, and documents. In many cases, digital assets are the main element of an experience as seen particularly with videos and infographics, along with social media content for sites like Instagram and Facebook.

DAM capabilities within a web content management platform are the best approach to ensure digital assets are an integral part of the digital experience. Assets can be mapped to the Web CMS taxonomy and the appropriate metadata applied. Having digital assets managed within the same WCM platform should help reduce the number of redundant or duplicate assets because a single asset is used across web, mobile, and social experiences.

**A single platform to manage digital assets and text-based content improves marketing agility**, especially within the web page (or mobile) creation process.

Not all Web CMS manage digital assets directly within the CMS itself, but there are simple capabilities that all Web CMS should support including the ability to automatically resize images, or use an internal image editor. Another good feature is federated asset management where assets can be stored in different locations, but leveraged within the CMS.

**Having digital assets managed within the same WCM platform should help reduce the number of redundant or duplicate assets because a single asset is used across web, mobile, and social experiences.**

## Templates

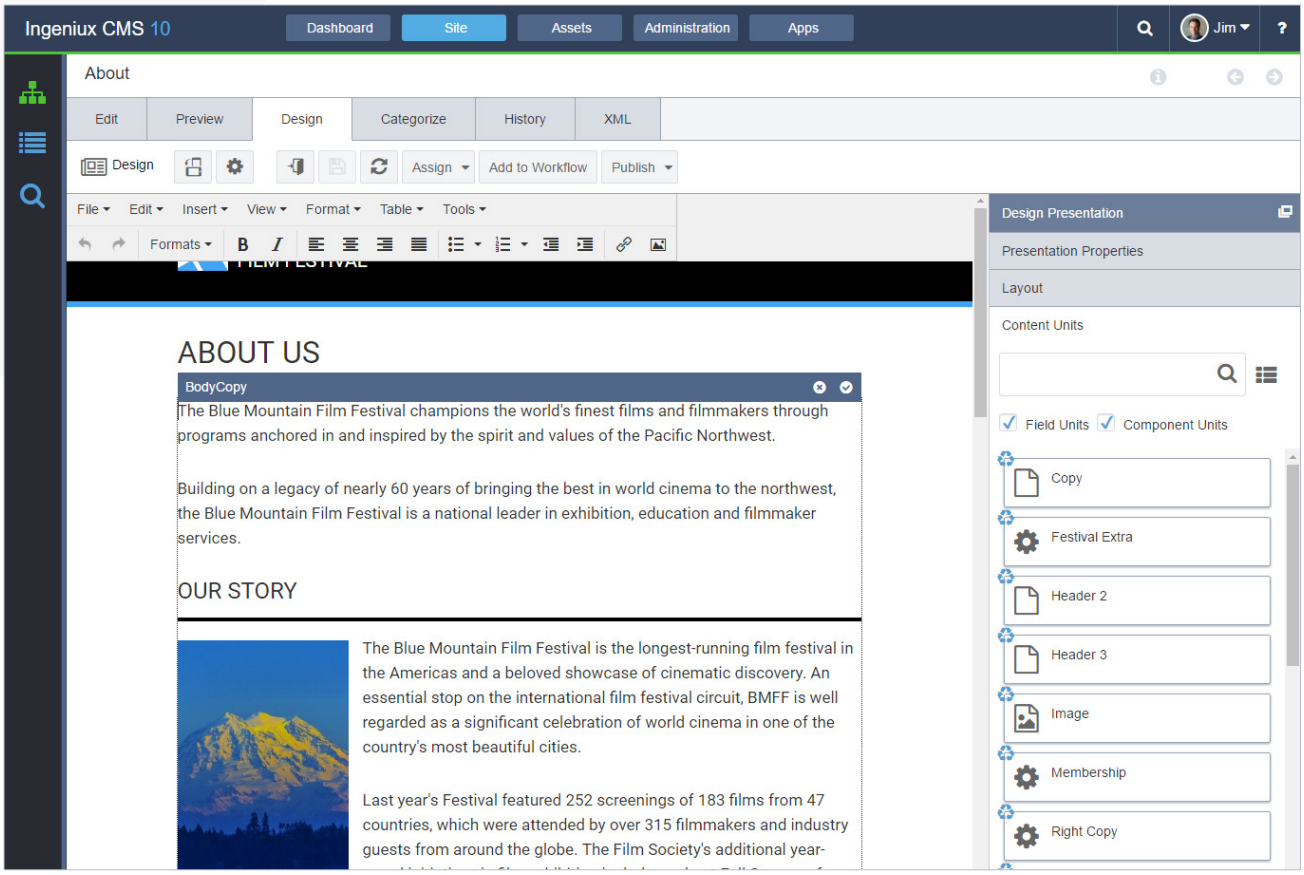
Since pages within a website typically follow similar structures, Web CMS's should provide the ability for users to create a new page based on an existing design. These designs are frequently called "templates" or "page types" and they should be readily accessible by non-technical content authors and editors. To learn more about templates, refer to the "Web CMS Template Approaches" section.

Some Web CMS provide built-in visual page builder capabilities which enable a non-technical user to quickly create a page from existing templates, widgets or components.

## Visual Page Builder

A Web CMS should provide built-in visual page builder capabilities which enable a non-technical user to quickly create a page from existing templates, widgets or components. A page builder supports quick modifications to existing templates, without the need to bring in web developers. The Page Builder should also be able to generate new templates and layouts from a blank canvas, and to generate a data model or "schema" for those templates based on the content and blocks or widgets that are applied.

When evaluating a Visual Page Builder make sure that is supports the latest mark-up standards for responsive design so your templates are optimized for mobile devices and tablets. The CMS should support multiple responsive frameworks, with the ability to easily update or customize the framework so you are not locked in to using legacy frameworks in the future.




# Analytics

A Web CMS should provide a means of tracking how website users interact with content – how many times they've accessed a page, the clickstreams used to navigate content, the ads that were most effective, etc. Analytics modules are the means of doing this.

Web CMS software generally supports analytics one of two ways: a built-in analytics module or support for a third-party analytics provider. While the integrated analytics of a built-in module is a nice feature, analytics is a software category of its own and has a much broader features set than can typically be supported by a Web CMS.

In our opinion, the better approach to supporting analytics is the provider model. With this approach, customers connect their existing analytics systems with Web

CMS dashboards that provide context to the content. Providers for analytics may include Google Analytics (usually free to use), Webtrends, Omniture, and other systems.

**The better approach to supporting analytics is the provider model.**

## Search

A Web CMS should provide search tools for the content repository. Search tools should offer options for content retrieval, relevance, and any number of other criteria appropriate for faceted search ("facets" referring to criteria such as file type, file size, color, date, region, price, brand, etc.). A Web CMS generally uses different solutions for an internal content or repository search and an external public web search, but this doesn't have to be the case. A public web search is often served by a third-party search application, such as Apache Lucene, Microsoft Enterprise Search Server, Google Search Appliance or the like.

However, some Web CMS's provide a built-in search application that supports both internal content management and external public website use. Tightly integrated search, or a native search-based architecture, means the search engine can quickly index all content in your CMS — structured and unstructured — thus improving the discoverability of relevant content in a single experience. It can also leverage CMSs built-in taxonomy and metadata functionality to further improve the quality of the search results.

## Reporting

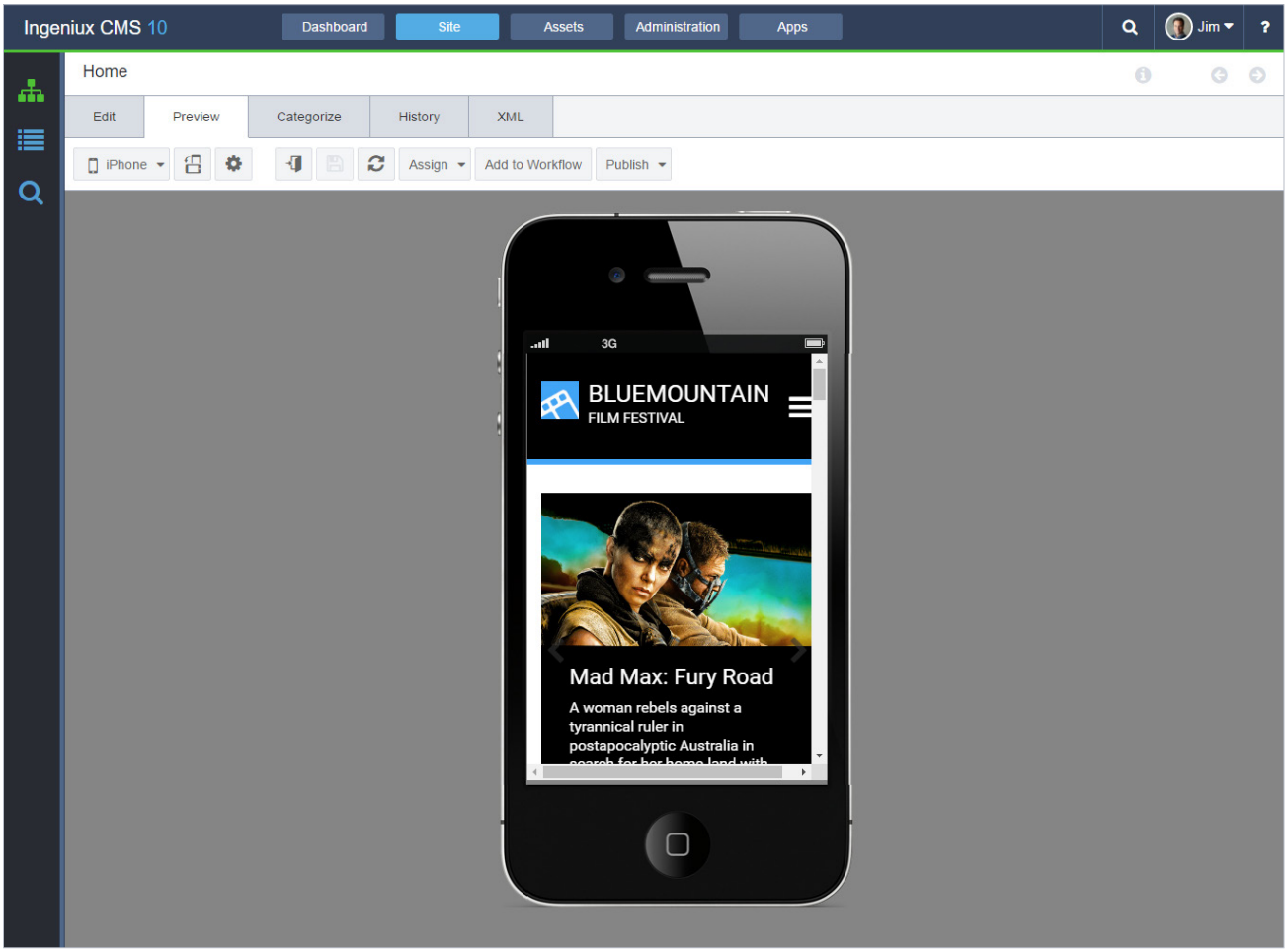
A Web CMS should provide reporting tools that make understanding content consumption quick and easy. These are typically dashboards that display content usage summaries.

Beyond a set of standard reports, such as audit trails, pages in workflow, etc., the Web CMS should provide the ability to create custom reports, such as where a digital asset is used within the website and in what context, or which components or widgets on a particular page are performing best. The ability to export a report as a Microsoft Excel or CSV file is also important.

# Mobile Device Optimization

**Mobile may be the most important feature or set of capabilities in evaluating a Web CMS today as it is outpacing desktop Internet growth by a wide margin.**

Since content will be delivered to a wide range of mobile devices, the Web CMS should optimize content for delivery to those devices. In order to do this, the Web CMS must be able to detect which device is making a request and deliver the right amount of content in the right screen format, checking for possible content incompatibilities, such as Adobe Flash content delivered to an iPad.



When planning for mobile it is also important to consider whether you need to develop a separate website or native app from your desktop website, or add another layer of presentation to your existing website to support mobile users. The decision often comes down to whether your website will support responsive or adaptive capabilities.

With responsive design, the web page is automatically resized based on the width of the browser used by a device. This minimizes the need for scrolling, panning or manually resizing the screen. Typically, you design web pages for the desktop and the responsive framework does the work of resizing and reframing the content for smaller form factors.

**Responsive is a popular approach, but doesn't necessarily address all the different mobile use cases.**

Responsive is a popular approach, but doesn't necessarily address all the different mobile use cases.

The second approach is design for mobile first (adaptive design). Mobile first means that you start your design thinking about the mobile user. You outline the content priorities, the core areas of the web page and how your visitors will engage with the web page using a mobile device. You can do this using responsive framework, but the best approach is to leverage your CMS's device detection capabilities to intelligently deliver content based on the device.

Another consideration is how the Web CMS can support multi-touch user interactions to enhance the customer experience on smartphones and tablets. Web CMS solutions with strong mobile capabilities support app-like mobile experiences and deliver touch-based interactions using HTML5 and frameworks such as jQuery Mobile.

## Applications

Since websites typically include a number of common elements such as calendars, newsletters, blogs, multimedia, user forums, etc. a Web CMS should provide each of these application types as a part of the basic product. Beyond the set of modules, Web CMS software should provide the ability to integrate content with external applications.

There are two different aspects of application integration to consider. The first is the ability of the Web CMS to integrate with other business applications such as a CRM, marketing automation, ticketing support system and so on. API-based Web services support, using SOAP or REST-based protocols, as well as connectors to external databases, are critical in integrating an organization's line of business applications and legacy systems. **Generally, a Web CMS's ability to easily integrate with applications and data is referred to as the "extensibility" of the CMS platform.** While extensibility is often one of the most difficult aspects of a Web CMS to evaluate, it is a critical consideration in gaining the business efficiencies and capabilities required in a website or project.

The second approach is the ability to manage content within the CMS for external applications and websites. This approach, known as Content-as-a-Service, or simply CaaS, is a strategy for delivering CMS managed content to Web applications and other channels. With CaaS you can manage and edit your content in a CMS and when the content is ready, it can be pushed as a resource file into an external application or the application can request content using an API-based web service. Not all Web CMS support content-as-a-service, so if this is a capability you require, investigate your CMS options carefully.

## SEO (Search Engine Optimization)

SEO is a mix of various features in the Web CMS that support best practices to help elevate search engine rankings for websites. Support for SEO should include:

**Search engine friendly URLs:** Web addresses for each page that describe the topic of the page, show where the page is organized in the website and do not use any special characters such as question marks, to denote dynamic content.

**SEO metadata:** The Web CMS should make it easy to add descriptive metadata for titles and page descriptions. It should also provide the ability to auto-populate metadata by default. Additional "bot" instructions, such as "no index" and "no crawl," should also be available.

**XML site maps:** Site maps are connected directly to leading search engines, such as Google and Microsoft Bing, and notify search engine crawlers of new or updated content and its priority in terms of indexing frequency.

Automatic redirecting and forwarding: When pages are renamed, moved or deleted, the Web CMS should automatically issue a permanent redirect so people using the old links can find the new content or be directed into other pages.

**Canonical URLs:** Every web page should only have one authoritative address. Canonical URLs redirects users from every variation of a web address to a single address. This may include adding (or removing) the www. before the domain and redirecting for various extensions (such as .htm, .aspx or .html).

## Multilingual

The ability to publish web content in different languages and for different geographic regions is critical for many organizations. A Web CMS should streamline the multilingual publishing process, making it easy to define a locale – a combination of language and region, for example French (Canadian) – for different sets of content.

Typically, a Web CMS can support multilingual in two different ways: as a version of each page, or as a clone of each page. Versions make it easy to track each translation for a page; however, the source page and each translated version of that page are bound together, so it becomes difficult to organize a different set of navigations and site structure for each language version of the site.

In the cloning approach, content is branched from the source page, but also remains linked to that source page. The cloned page may be organized into a different site structure. When one version of a page is updated, the owners of each of the other clones are notified of the changes. We find that for most organizations cloning

is a better approach, as companies often offer different products and services in different countries. The ability to support variation is essential.

A Web CMS should support translation, either side-by-side in the CMS, or through an external translation service bureau using an import/export system. Web CMS clients utilizing multilingual features need to support foreign character sets, such as double-byte characters and Unicode, as well as bidirectional (BIDI) text for Arabic and other languages. Lastly, it is often essential for Web CMS software to provide localized versions of the software so users in different countries can work in their own language.

## Multisite Publishing

A Web CMS should provide the ability to manage and deliver content for multiple sites. Again, like SEO and multilingual, multi-site publishing is not as much a feature as a collection of features in support of best practices.

Publishing targets provide the ability to choose which sets of content are deployed to which location and site, as well as determining the format the content is delivered in. For instance, a news story may be published to the public website using one look and feel and to the company intranet using another.

Similarly, a Web CMS should support managing content in different environments, such as test, staging and production. Content may be promoted between these environments to ensure that it is properly tested and approved before it goes live.

Finally, the CMS should offer a Content API your external applications can connect with to pull content from the CMS. This API is read-only so you can't change content in the CMS, only access what is already approved for publishing.

# Web CMS Template Approaches

To make clear just one technology decision, we will discuss briefly several approaches to site templating. Four common options are HTML, application server technologies (.NET, PHP...), XSLT, and Apache-based approaches.

## HTML Templates

HTML is the language used to create and define templating standards. While this approach has its pros and cons, it usually works well for primarily static sites ("early binding" sites).

## Application Server Technologies

This option tends to be the best templating approach for sites on the "late binding" side. This approach is often complex, but it offers the highest level of personalization. The details of this approach will vary depending on which application servers are used and can range from industry standards to completely proprietary. Application server technologies may include Microsoft .NET, PHP, Cold Fusion and Java.

## XSLT

This approach is based on transforming content within XML documents based on definitions in a stylesheet. XSLT is widely used, and is among the most extensible and flexible of all templating methods.

## Apache-specific Options

This includes two examples: Tiles and Velocity. Tiles is based on the Struts framework, and, very generally, allows for the definition of page fragments that can be dynamically assembled into a page at the time it is requested. Velocity is a Java-based open source templating framework, based on the MVC model, which emphasizes the best practices-focused, independent development of application code and page design.

# Understanding Deployment

For the purpose of explaining deployment options, let's assume that the optimal set of content for a website exists within the content repository. There are a number of ways that content can be delivered.

## Early vs Late Content Binding

The first consideration is that of binding. That is, when does the content on a web page actually bind with the page? When a web page always consists of the same content – such as a press release, for example – that page is said to be static. The content on the page never changes (unless updates are made to the content itself). When a website visitor views the web page, the press release will always be the same. In this case, the content has "bound" with the page at a very early stage (known as early binding).

In other instances, website content does not bind with the page until the website visitor requests the page. Suppose that a visitor clicks on current events. Within the repository are events that happened last year, this week, and events that will happen next month. Therefore, the current events page will have content that changes on a daily basis; in this case, the page and the content bind only when a user requests the page.

This is called "late binding" content, and such pages are made possible by the dynamic capabilities of the Web CMS. Instances where late binding capabilities are important include those where companies will want to offer content based on user preferences. Retail enterprises may want to present a particular color or size of clothing when a shopper clicks on the current specials section. This may be dependent on product availability, which the Web CMS may also need to check prior to delivering the content.

When considering the purchase of a Web CMS platform, buyers will want to understand in detail what the capabilities for the early and late binding of content and pages are. In most cases, there will need to be some ability to combine early and late binding content, such as a press release as part of a page that contains several other content objects. The collection of objects on this page would be late binding.

## Using a Content Delivery Network (CDN)

Another website deployment consideration is whether to use a content delivery network (CDN). A CDN provides mirroring and caching of web content across multiple servers placed in data centers worldwide. CDNs provide very fast content delivery and are ideal for delivering images as well as streaming media files. With a CDN, you can easily scale content delivery without adding additional hardware by utilizing native cloud services. Because content is redundant, a CDN helps ensure maximum uptime for websites. The challenge of working with a CDN is that it is difficult to support dynamic content. Some CDNs support a short list of server technologies, but deploying dynamic or data-driven content on a CDN adds additional complexity and can be expensive. CDN deployment often requires using a static, decoupled deployment of HTML, or the "pull" approach, where the content is pulled in by the application pages from static content.

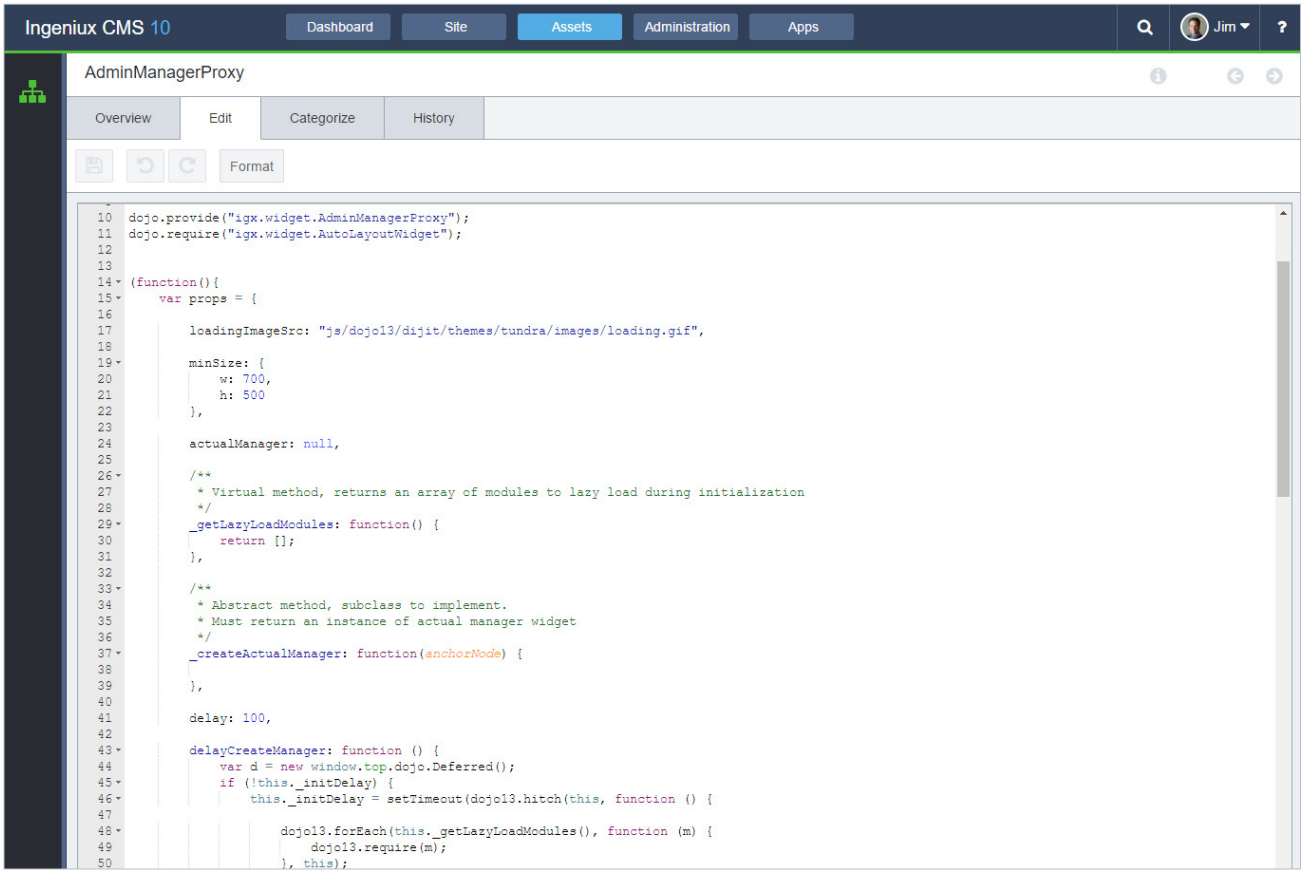
**With a CDN, you can easily scale content delivery without adding additional hardware by utilizing native cloud services.**

# Web CMS Implementation

The implementation of Web CMS systems commonly presents as many challenges as the selection of the Web CMS itself.

While various Web CMS's require different levels of proprietary product expertise, all products will require specialized knowledge of best practices for implementing that particular system.

Therefore, **implementation almost always involves some combination of internal and external expertise.** Some companies opt to perform the bulk of implementation themselves, taking a DIY approach, and only enlist the help of external experts



for particularly problematic areas. Other customers hire third- party system integrator firms to perform implementation. This is often a company with close ties to the selected software vendor, such as a certified partner within the vendor's services network. In other cases, buyers opt to let the vendor's own professional services team perform the implementation.

In each scenario, there are do's and don'ts, such as the following:

## DIY

Not knowing what they don't know, Web CMS buyers often embark upon an implementation with no reliable idea of how many IT resources or how much time will be required to implement their project. They typically commence such projects with estimates of three to six months, and typically conclude projects within three to 18 months. Sometimes their estimates are accurate, and sometimes they aren't.

**A little insight into the implementation do's and don'ts can go a long way.**

And because the devil really does live in the details in Web CMS land, it is not uncommon for a DIY implementation to take two to three times as long as planned.

It is advisable in such projects for Web CMS buyers to enlist the aid of an external implementation expert, at least for the initial planning phase. A little insight into the implementation do's and don'ts can go a long way.

## Third-party Systems Integrators

Third-party integrators often represent a good choice for combining specific Web CMS product expertise with industry-wide best practices. While the vendor's professional services will certainly offer guaranteed product knowledge, they may not be as competitive as third-party integrators at offering bleeding-edge implementation practices relevant to specific business use cases or objectives. However, buyers must be thorough in their assessment of a third-party integrator's product knowledge of the Web CMS they have chosen.

Buyers must also be cautious when allowing a systems integrator to make the Web CMS purchase decision. Because integrators integrate a limited number of Web CMS's (typically two to four), they will advise clients to buy one of these systems even if another product would be a better fit.

## Vendor Professional Services

As mentioned above, vendor professional services are a guaranteed way of getting up-to-date product expertise. They understand intimately how the CMS works and can often provide the best judgement in terms of how long it will take to implement the Web CMS based on your specific business and functionality requirements.  In specific industries, however, third-party integrators may have more up-to-date implementation expertise than vendors.

# Buying a Web CMS (Again)

Today many organizations buying a new Web CMS are upgrading from a legacy system.

Organizations change CMS solutions for a wide range of reasons: the user experience may be too complex, the software may have been implemented incorrectly and requires additional investment, the old CMS may not offer the current features required to deliver a compelling web experience, the cost of upgrading to a new version warrants an evaluation of other solutions, or the system has become too expensive to support, to name only a few.

For buyers upgrading from legacy systems the outlook is good. Web CMS software has improved dramatically over the last few years, while costs have fallen. You can get more from your Web CMS investment than ever before.

While upgrading does not fundamentally change the process of buying a Web CMS, it does introduce additional criteria. Foremost is how the existing content and applications may be migrated into the new Web CMS. While many vendors provide migration or site import utilities, the fact is that migration always requires some investment and level of effort.

**While upgrading does not fundamentally change the process of buying a Web CMS, it does introduce additional criteria.**

A content inventory is generally a best practice when migrating to a new Web CMS. In a content inventory the content owners define which pages need to be created, refreshed, maintained, or retired and prioritize those updates. Integration of existing and planned applications requires understanding the Web CMS technology platform, extensibility and deployment model.

Second-time Web CMS buyers are often guided by the pain of their existing system, rather than the opportunities of the new system. If the legacy Web CMS was slow to publish, or made it difficult to support multiple environments like staging and production servers, evaluators tend to focus on these issues above all other criteria. While we would not recommend purchasing a Web CMS that does not meet key requirements, it is also important not to miss the forest for the trees. The focus should be on the features and support that can help you execute your business strategy.

Lastly, **every Web CMS is different. The savvy buyer knows it's not what the CMS can do, but how the CMS supports each feature and use case.** Even experienced Web CMS buyers can fall into "check- box" evaluations where the emphasis is on the breadth of features, rather than the overall value of the solution based on the quality and depth of the features, the usability, and key criteria such as performance and scalability.

# Our Best Advice

We encourage Web CMS buyers to remember that choosing a Web CMS is a practical matter.

While it may sometimes be useful to discuss the academic niceties of various content management-related topics, the business use cases that mandate a solution cannot be postponed. Prospective Web CMS buyers are generally very good at articulating their Web CMS problems, and we encourage them to make use of these dilemmas as a starting point toward choosing a content management solution.

It's about information agility and delivery agility. Content is mobile first, intelligent and readily available for any website, campaign, app or device. The Web CMS is the platform for content creation, governance and orchestration — but not necessarily the platform for presentation delivery.

**Choosing a Web CMS is a practical matter. It's important to not find the best CMS, but find the one that is "best for you."**

As anyone who has ever tried to tackle the Web CMS problem can tell you, **effective content management is an ongoing project**, and it is about much more than simply managing websites. Businesses need agile content solutions that enable delivery of relevant content where the customer is, regardless of channel or device, exactly when they need it.

It must support content creation so that organizations can easily reuse it in different formats and quickly deploy it to any channel necessary.

To help approach a Web CMS purchase, we advise prospective buyers to:

- Assemble a list of current challenges that stem from content-technology, content-people, and content-process issues.
- Define issues that are likely to arise in the next few years.
- Define overall web strategy and tactics, as well as the metrics that will be used to measure success.

Use these criteria as the basis for evaluating content management systems and vendors.

If a Web CMS can successfully address these scenarios, within budget – and you as the buyer believe in (a) the vendor's viability, (b) the potential for positive long-term return on investment, and (c) the likely satisfaction of your users with the Web CMS platform – we believe you have found the right solution for your business.

# Get started!

# About Ingeniux

Ingeniux is the leading provider of web content management and digital experience software. We enable organizations to orchestrate the entire customer experience from acquisition through to sales to support and service, across any device, application, or website.

We build content management software with an unparalleled focus on the content itself. The Ingeniux CMS is designed to manage and deliver modern websites, customer support portals, online communities, and other customer touchpoints.

We believe in intelligent “structured” content. We design our software to enable content reuse, enable true mobile and multi-channel content delivery, and insightful content discovery. Our unique content-as-a-service capabilities deliver content into web and mobile applications, and other key channels.

Ingeniux software is available as a fully managed software service or an on premise application. Ingeniux delivers unparalleled service and support to customers worldwide.

To learn more, visit us at http://www.ingeniux.com.

INGENIUX

PO Box 21466
Seattle, WA 98111-3466

info@ingeniux.com
877 445 8228