Intelligent Content Management: A Primer

INGENIUX

Content beyond boundaries

Contents

The demand for consistent, engaging content across all channels is changing the way we create and manage content. Gone are the days when we could create multiple versions of our content for each channel. There are too many channels with diverse criteria for this to be feasible.

Instead, we need a way to create content once and reuse it across all channels and devices.

We need intelligent content management.

What Is Intelligent Content Management?

Intelligent content (also referred to as 'structured content') is content you create and manage separately from how you publish it. When the content is created, metadata and other descriptive information is included, making the content reusable across all channels – websites, web applications, portals, mobile apps, and more – in different formats and views.

Intelligent content is content that is:

- Structurally rich
- Semantically categorized
- Reusable
- Reconfigurable
- Adaptable

An intelligent content management strategy is an approach that defines how you will create, manage and deliver "intelligent content" across the organization and how you'll leverage that content using a content management system. This approach allows you to manage and deliver content more effectively – ultimately saving your organization time and money.

In this primer on intelligent content management, we'll walk you through the four key elements of preparing for an intelligent content strategy:

1. Content Strategy
2. Content Modeling
3. Content Storage & Repository
4. Content Delivery and Access

Let's get started!

Content Strategy

Every organization creates and manages content - but not every organization understands how to create and manage that content effectively. Every great user experience, or successful online customer engagement, has behind it a solid content strategy.

In this section, we'll explore how organizations can apply intelligence and metadata to content to manage and distribute that content effectively.

Defining Content Strategy

The simplest and best definition of content strategy:

> *"Content strategy guides the creation, delivery, and governance of useful, usable content."*

When you break it down, a content strategy provides a few different things. It:

1. Identifies the topics and types of content you create.
2. Defines how the content is organized, formatted, and published (structure of the content).
3. Describes the processes, tools, and people involved in creating and managing content through its full lifecycle.
4. Identifies who is involved in defining and adapting the content strategy; including key decision-makers, stakeholders and influencers.

Some people confuse a content strategy with a content marketing strategy, but the two are not synonymous.

A content marketing strategy focuses on identifying the personas or audiences your organization wants as customers and outlining the content to create that supports the customer journey (from research to decision, to purchase, and post-purchase).

A content strategy isn't only about marketing to customers. It about understanding all the content you create in your organization, how to organize it in a way that is understood and reusable by many channels and departments. It's about marketing content, support content, sales content, and technical content. And it's defined outside of any technology implementation.

> *"It's a simple matter of knowing what our future should look like before we step into that future. We create with our strategy and our technology follows."*
>
> – Cruce Saunders, [A]

Before we go any further, it's important to understand that you can't create a content strategy that supports the entire organization if it isn't built on an intelligent content framework. Intelligent content is content created with a structure that enables you to reuse it for different purposes, channels and in different formats. It allows you to mix and match content from different areas of the company to create a customer experience that works for each customer in their context.

How to Create a Content Strategy

There is a lot that goes into a content strategy; we'll outline the key points here.

If a content strategy is about the planning, development, and management of all content in your organization, then you first need to understand what content you are creating now. Two things you can do to understand current content is:

- Create a content ecosystem map.
- Complete a full content inventory/audit.

The Content Ecosystem Map

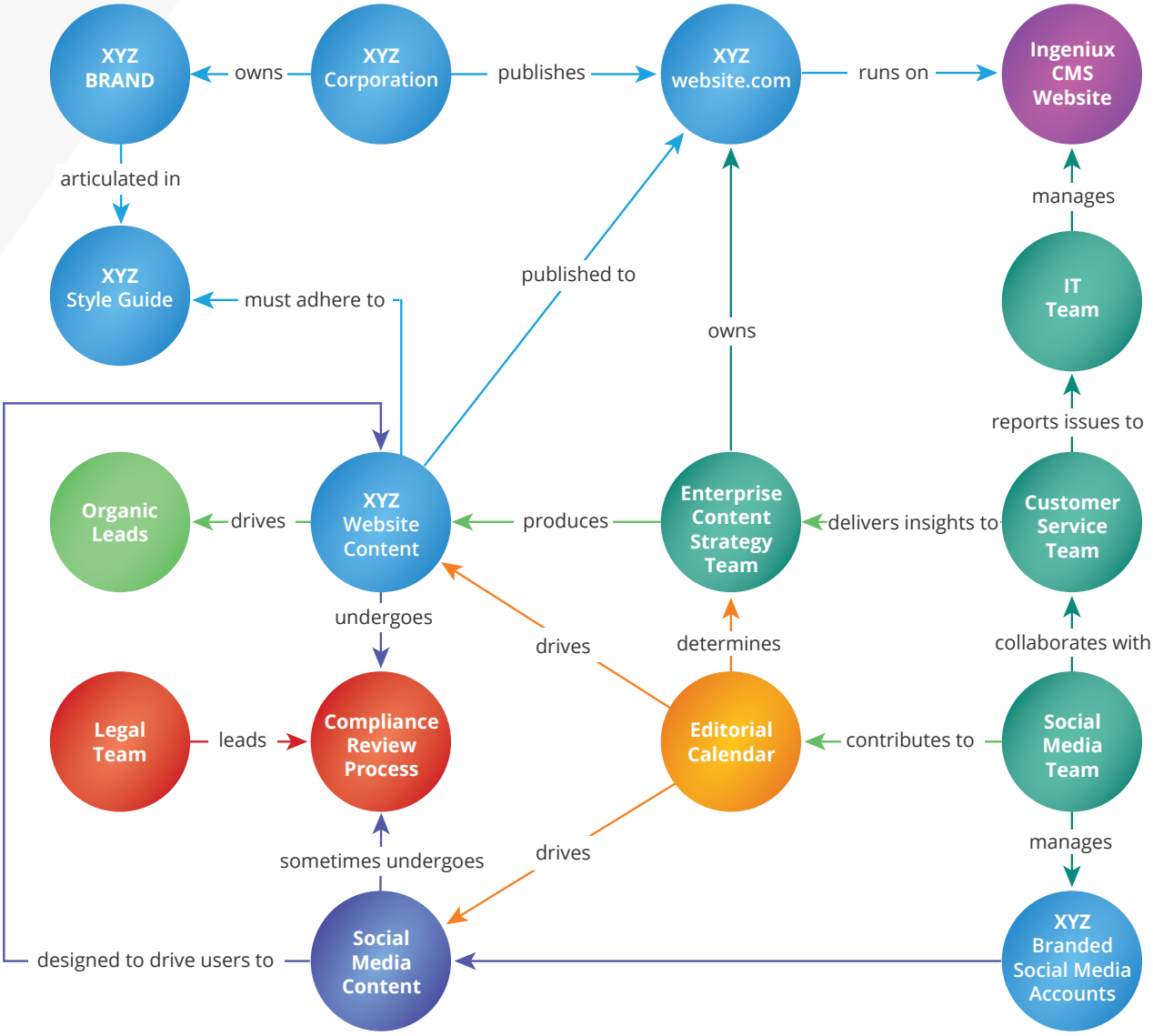
A content ecosystem map is an idea from Scott Kubie of Brain Traffic. It's a visual representation of the content in your organization.

> *"A content ecosystem map is a picture of your content ecosystem. Your content ecosystem includes all of your products, brands, content types, teams, technologies, and/or channels. Finding the boundaries of what you think of as your content ecosystem is part of the point of creating the map. What's in? What's out? What matters in your content world?"*

Kubie says that a content ecosystem map is the current state of your content. It's not a content model or a content inventory. It simply helps you understand what content you currently have in your organization and the processes around the creation, management, and publishing of that content.

If you are creating a broader content strategy than what you need for marketing and digital experiences, you could build a series of maps that interconnect to give you a complete understanding of all the content in your organization and how it relates and overlaps.

Here's an example of a content ecosystem map, based on one from Brain Traffic:



The Content Inventory and Audit

This is a high-level view of your content, however. The next step is to go deeper and create a content inventory identifying all the content in your organization. Typically done in a spreadsheet format, each content asset would be a row with all the information about that asset in a series of columns. For example, you would include columns for:

- Content Asset
- Type of content
- URL/location
- Where it's found in the organization

- Owner
- Users
- Metadata
- Taxonomy (Topics/tags)
- Workflow rules

Once you have a complete inventory, you should spend some time auditing that content to ensure it's still used in the organization. A content audit allows you to reduce ROT (redundant, obsolete, and trivial) content that is wasting space and management efforts. When you reduce ROT, you can improve the relevancy and findability of the good content.

As you work on your inventory and audit, take into consideration digital asset management and brand guidelines.

Building an Intelligent Content Framework

With a current state defined, you can now take the next step and plan your go-forward content strategy. The plan will include information such as:

- The topics, themes, and areas of focus for your content
- Who owns each
- The taxonomy your content will map to
- The metadata your content will include
- SEO guidelines
- The workflow processes involved in creating, managing and publishing content types
- Voice and brand guidelines for public-facing content
- The content governance model
- How you will ensure the quality of your content
- The channels you will deliver content to, including the format of the content for each channel

A content strategy must be communicated and shared with everyone in the organization. It should include tools and information that help others follow the strategy in their work.

Sauders made a very important point on content strategy:

> *"...organizations need to build a regular muscle that not only builds a strategy once but can then optimize it based on the constant inputs they get from customers, the market, competitors and the industry."*

Content Modeling

The main component of intelligent content management is the content model. A content model defines the type of content you create, how to structure that content, as well as the metadata to associate with the content.

When you build your content model following this structured approach, you create content that is reusable across channels. With an intelligent content model, content is created once and used in many different formats and publishing channels including your website, a mobile app, a web application, a chatbot and more.

Content Model: Define Content Types

To create an intelligent (or structured) content model, you start by defining content types such as an article, image, news story, press release, knowledge base article, and so on.

Next, for each content type you define a set of elements, also called attributes, such as title, description, summary, media file (image, video), author, main body. You also define the format of that element (date, text – 50 characters, number, image, etc.). The number of elements you have will depend on the content type and how you envision using it across the organization. The key to creating structure is to break down the content into its key elements, but you don't necessarily want to break it down into too many discrete elements that you will not use.

For example, when you create a press release content type, you might define elements including title, subtitle, summary, main body, but you don't need to break down the main body into separate elements for each paragraph because you wouldn't use the individual paragraphs. However, you might want to create a separate element for a quote so that you can use it for different purposes.

The decision on how to break down a content type into its elements depends on several factors: how you want to display the content in different channels and how you want to reuse the content in different channels.

After you define the content type elements, you then define the metadata associated with each content type. Metadata are content type elements that you don't see and include things like Create Date, Publish Date, SEO description, author. Metadata comes in several flavors:

- Descriptive Metadata: title, abstract, author, keywords
- Administrative Metadata: date created, file type, access/permissions
- Structural Metadata: version, relationship to other content types

Content Model: Define Relationships

Once you have defined all your content types, you need to define the relationships between content types. For example, the Content Type 'Author,' is related to the Author element in the New Article content type.

CMS and the Content Model

Every content management system supports a different approach to creating content, but only a few support a structured, intelligent content model. With Ingeniux CMS, once you have your model defined, you can set it up in the CMS using the Schema Designer. Without a schema designer like the one in Ingeniux, you may be required to modify your content model to fit the capabilities of the CMS.

The Ingeniux Schema Designer creates XML schemas that are then natively available as XML or JSON. It supports over 25 standard element types, such as Text (string), XHTML (WYSIWYG editing), image, media file, database query, navigation, etc. The Schema Designer allows content models to change and web administrators to selectively update existing content with those changes through versioning as well as syncing and replication to propagate changes across existing content items.

We've looked at modeling your content, now let's look at how you can store it for easy access and management.

Storage and Repository

As part of your intelligent content management strategy, you need to think about how and where you store your content.

There are three different ways to store content in an intelligent content management system:

1. Traditional Relational Database
2. XML files
3. NoSQL Database

Managing Intelligent Content in a Traditional Relational Database

It is possible to store structured content in a relational database, but it comes with potential challenges. Relational databases store content in tables with rows and columns. Once you have the structure of your content types defined, you can create tables that match those content types, creating columns for each element in the content type. For free-form fields – like the main body of a blog or a press release, you can use the BLOB field type.

There are several challenges to using a relational database for structured content:

- Using BLOBs makes displaying content for different channels harder. It also impacts the quality of your search queries.
- Tables can get very large and unwieldy if you have many elements in your content type. This can affect the performance of your database.
- When you need to make changes to your content type, you must update the table. Updates may apply to current element fields or add new element fields. These changes may require a transformation of content if existing fields are changed. In both cases, you will have to rebuild your database index and potentially drop or recreate related stored procedures your CMS application works with.

Storing Content in XML Files

You can store your content in XML files instead of a database. An older method of leveraging XML for structured content, this approach isn't applied so much to

managing content in an intelligent content management system as it is a way to store content for a web application that you manage in the CMS.

An intelligent content management system can replicate XML files to a publishing target or remote location. Embedded Help and labeling are two examples where content replicated to XML files works well. When content is stored in files using XML, the application is updated to load the files like a supporting resource.

When delivering content via an XML file-based method, you have to pay particular attention to timing and caching, as well as setting up the proper connections and enabling the delivery of files across the firewall.

Leveraging the Schema-Free Nature of a NoSQL Database

The best approach to managing content in an intelligent content management system is a NoSQL database. A NoSQL database is schema-free, so content can have any structure you define, and you can change that structure over time without development or database changes. A NoSQL repository supports millions of topics and documents. Automatic clustering and sharding allow repository clustering without DB administration or development costs.

You store content in a NoSQL database as documents or JSON objects. Content has rich metadata, and you can easily search and retrieve it using that metadata. All the content in the repository is indexed and cached in memory, enabling information to be quickly processed and delivered without any performance impact, database IO, or SQL injection risk.

NoSQL is less expensive than a relational database and easier to administrate, making it a good choice for cloud-based deployments.

Storing Content Efficiently and Effectively

While there are several options for storing structured content, it's important to carefully consider the advantages and disadvantages of each approach before you select the best option. The one you choose also depends on the CMS you select as not all content management systems support all three approaches.

Up next, delivering content to multiple channels and devices.

Delivery and Access

The hybrid approach to content management supports the delivery of content in an intelligent content management system to multiple channels and devices.

An intelligent CMS solution offers multiple deployment options to fit how the organization needs to deliver its content now or in the future. These may include:

- Dynamic delivery using a server technology like ASP.NET or Java
- Multi-format delivery using mixed or different server technologies
- Web services delivery using a REST or SOAP-based API
- Device-targeted delivery using a mobile detection system
- Push-based delivery such as XML, JSON or into an external database so a remote application can consume the content
- Plain old HTML delivery for static Web content

Let's examine a couple of these in more detail:

Delivery Via Dynamic Delivery Server

In an intelligent content management system, your backend content management administration is separate from content delivery. One way to deliver content is using a loosely coupled delivery server that acts as the delivery tier. In this case, you have a content delivery framework or application used to render dynamic content and other services.

When you have a website or microsites that you want to deliver content to, this approach works well. The dynamic delivery server can not only deliver content to a website; it can apply business rules for personalization or device and channel optimization.

Web Services via Headless Content API

One of the benefits of intelligent content management is the ability to deliver content to channels separate from your CMS. This is done through the delivery of content using a headless content API.

Any application, including native mobile apps, that want content managed within the intelligent CMS, pulls that content from the CMS using a content API. The application or website then formats the presentation of that content anyway it wants.

The Content API approach works for organizations building applications using modern client-side JavaScript frameworks like AngularJS and Ember or companies building highly customized web experiences not supported through the CMS.

Device-Targeted Delivery Using Mobile Detection

Not every mobile experience should look that same. An intelligent content management system can detect the type of mobile device requesting content and deliver that content in the appropriate format.

Mobile detection requires management of a device library that is regularly updated (mobile device form factors and capabilities change often).

Push-Based Delivery Using XML or JSON

Instead of an application or website "pulling" content from the CMS using a content API, the CMS can automatically "push" content to an application or website in XML or as JSON objects the website then loads.

This approach works well for managing help content in the CMS, and other content that doesn't change often but you want to manage in the CMS.

4 Approaches to Content Delivery: One Intelligent CMS

There is no one way you will want to deliver content. Every website or application may have different requirements, and potentially be built using different technologies. An intelligent content management system will provide several ways to deliver content including a content API and a dynamic delivery server. You get one place to manage your content for all your publishing channels and devices.

Putting It All Together

Ann Rockley said the key to creating intelligent content is to think about what the content is (e.g., defining elements such as teasers, title, call to action) and not how it looks. The channel will define how it is to look. It's about understanding the content and how you can break it down into its key elements, describing it as accurately as possible so that any application that requests it will understand what it is and how it's structured.

Intelligent content management focuses on identifying the right content strategy, mapping the content model, and then implementing that model within a content management system. The CMS then stores stores it and makes it available for multiple channels regardless of where that channel resides.

As you embark on your intelligent content management strategy, planning for these key elements will help you define the best content strategy for your organization and select the right CMS to support that strategy.

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.

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

To learn more about Ingeniux portal and web content management solutions, e-mail info@ingeniux.com.

INGENIUX

PO Box 21466
Seattle, WA 98111

info@ingeniux.com
877 445 8228