

Hearst Magazines Magnus Platform Content API

Jim Mortko

VP, Engineering

Hearst Magazines

February, 2013

Quotes

"What we have to learn to do, we learn by doing."

- [Aristotle](#)

"We learn by example and by direct experience because there are real limits to the adequacy of verbal instruction."

— [Malcolm Gladwell](#), [Blink: The Power of Thinking without Thinking](#)

"If you want to set off and go develop some grand new thing, you don't need millions of dollars of capitalization. You need enough pizza and soda to stick in your refrigerator, a cheap PC to work on and the dedication to go through with it."

- [John Carmack](#)

History

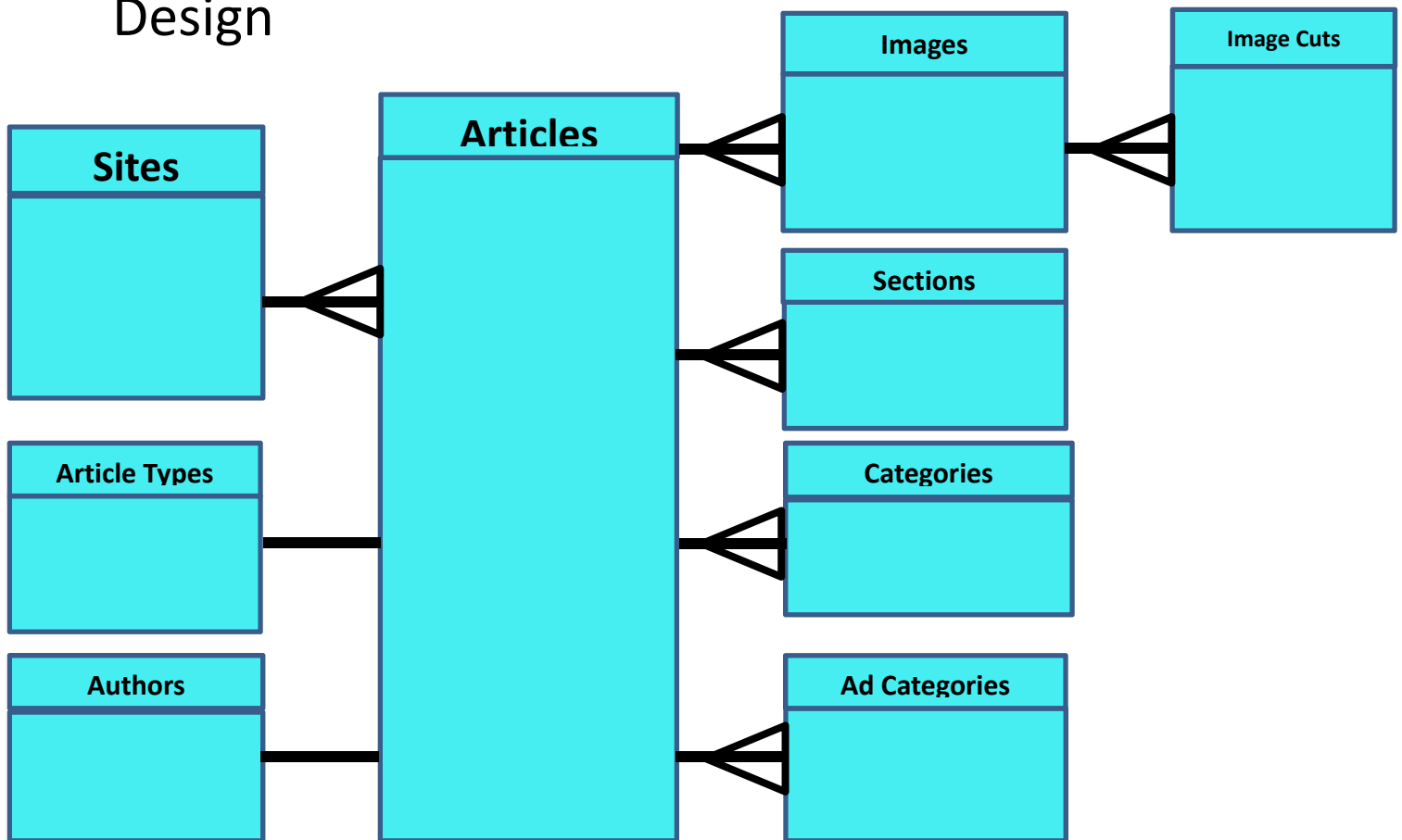
Magnus is Hearst Magazine's internally developed Content Management System (CMS). Prior to 2006, Hearst had all of its website content in the iVillage platform, a woman's network that was part of the Hearst portfolio of holdings. In early 2006, NBC purchased iVillage and the arrangement stipulated Hearst have all of its content out of the iVillage platform in 12 months. There was instantly a need for a new system to serve the Hearst magazines websites. A new system had to be built, all content ported over and each site had to be completely redesigned. Additional strategic requirements stipulated a tight integration with various other systems, namely User Registration, Ad Serving and Circulation Marketing/eCommerce. The choice to build an in-house system was made for us.

The build for Version 1.0 began in Q3 2006. The first site launched in Q1 2007. The system currently houses and serves content for 30+ websites for both the US and UK. Since the beginning, over 1,000 different people have used the Magnus system to publish content from over 11,000 different authors.

The system has grown over time as Hearst has added new sites to the platform either by new projects or via acquisition. At present, the system is responsible for serving 35 million page views per day.

Last year, in anticipation of moving towards a model where we make greater use of web services, the Magnus Content API was built. Just in time for this Hackathon!

Design



This is a simplified version of the Entity Relationship Diagram (ERD) that represents the major database tables in the Magnus CMS.

Best practices like normalization were adhered to during the design. We make use of a good deal of Object Relational Mappers on the development back-end, so it's a decision that has worked out well in that regard.

This is a simplified look at a segment of the database use by the Magnus platform.

The main database table is "Articles". Most other tables are either attributes of the Articles table or one-to-many child tables to the Articles table.

Design (con't)

Sites have one to many articles.

Articles have one Article Type.

Articles have one Author.

Articles have one to many images.

Articles are in one to many Sections.

Articles have one to many Categories.

Articles have one to many Ad Categories.

Sites

The majority of Hearst Magazines serve their content out of the Magnus system. At present there are 30+ sites in the system.

Content comes into the system in several ways:

- 1) Imported from some other CMS system (Vingette Story Server, EZ Publish, Wordpress)
- 2) All print content ends up in Magnus (in “draft” format, publishing this content on the web is up to the discretion of the web editorial team)
- 3) Imported from some 3rd party via business partnership (e.g. recipes imported from Martha Stewart)
- 4) Organically created in the system by the web edit teams (accounts for the majority of the most recent content)

Note: There are some inactive titles in the database for properties we no longer publish.

There is a great deal of fashion content in the system across all titles. The magazine titles with the most fashion content:

- Elle
- Harper’s Bazaar
- Marie Claire
- Cosmopolitan
- Esquire

The list of sites in the system can be access by calling the “Site Search API”.

Articles

Magnus currently contains close to 700,000 articles.

“Articles” actually represent a variety of different content types (e.g. Article, Blog Entry, Flipbook, Recipe, Q&A Interview, Video Clip, etc.)

Articles have a one to one relationship with Article Types. The Article Type is not only a label, but in many cases defines which extra data points need to be captured that are not in the base set of fields held by an Article (e.g. Title, URL, teaser, publish_date, etc.) Article Types may contain an XML DTD that informs the system :

- (1) how to prompt the user for input fields in the CMS Admin Tool
- (2) how to store the data in the “body” data field in the database and
- (3) how to decode the XML in the “body” field and deliver it to the template during rendering time on the web.

For the purposes of using the API, you’ll get a structured array called “content_loop” if the if the type does have a complex XML definition (like a flipbook or a recipe), otherwise “body” will be returned as a string (as in the case of a blog post or a plain article).

Flipbooks (aka “slideshows”) are by far the most popular viewed (and thus created) content type. “Flipbook 1” consists of a single article page with multiple images associated to it. “Flipbook 2” and “Flipbook 3” consist of a multi-page article, each with a single image attached to it. These later versions were done for SEO purposes in order to have each image have its own distinct URL for indexing.

Other notes :

There are over 11,000 authors in the system.

There have been a lot of hands on the content - it has been managed / published in this system by over 1,000 different people over the years.

Use “Article Search” method in “Magnus Content API”.

Images

Images are one of the greatest assets of the Magnus CMS. Images can be associated in a many-to-one relationship with a Magnus Article or article page.

Different Magnus sites use different image sizes. The CMS allows for a large master cut to be uploaded or imported and all required smaller cuts are automatically created, maintaining aspect ratio.

Images have their own set of meta data, including:

- Name
- Caption
- Description
- Search keywords (SEO)
- Internal keywords
- Photo credits
- Image source
- Rights information (tie in to our syndication system, “Hermes”)

Note: Some sites have different image sizes in the same named “cut” fields.

Use “Article Image Search” method in “Magnus Content API”.

Sections

Sections represent the physical location of the content as it exists on a site.

Sections are hierarchical in nature and should have a direct relationship to the navigation bar on the site. Sections have a site as a single attribute.

Sections have “prefix” values that are part of the URL of the article. Each article *should* only exist in one section. Given the rules of SEO, sites that display the same exact content in more than one site/section of a site are penalized in their Google search rankings.

One exception about Sections to note are the Elle Runway sections. Articles here are actually assigned to 4 sections, in order to allow navigation on the site by (1) Runway Type (2) Season (3) Designer (4) type of flipbook. Any article in these sections can be accessed using the “canonical_url” field.

Use “Article Section Search” method in “Magnus Content API”.

Categories

Categories are an editorial created taxonomy of contextual classifications for articles. The intention is to be able to access articles that are in similar groupings based on their actual content.

The taxonomy is by nature hierarchical. Categories, like sections, also belong to a site – so each site is able to define its own taxonomy. There is no limit to the number of categories an article can have associated with it, but typically we see articles tagged with one or two.

Categories can be used for pulling like content together for purposes of syndication or ad targeting for a given site.

Use “Article Category Search” method in “Magnus Content API”

Ad Categories

Ad Categories are an Ad Sales created taxonomy of broad marketing classifications for articles. The intention is to be able to group articles into buckets based on how the sales teams would like to execute on targeted ad campaigns that span multiple sites. For example, a cosmetics advertiser might want to be able to only have their ads displayed on “beauty” content across our entire platform.

The taxonomy is by nature hierarchical. Ad Categories do not belong to a single site. There is no limit to the number of ad categories an article can have associated with it, however the practice has been always to tag every article with two.

Use “Article Ad Category Search” method in “Magnus Content API”.

Good Luck!

Instructions should be available on the site to gain access to the APIs and generate a developer key.

<http://developer.hearst.com>

Sign up for the “Hearst Hackathon Bronze Package”