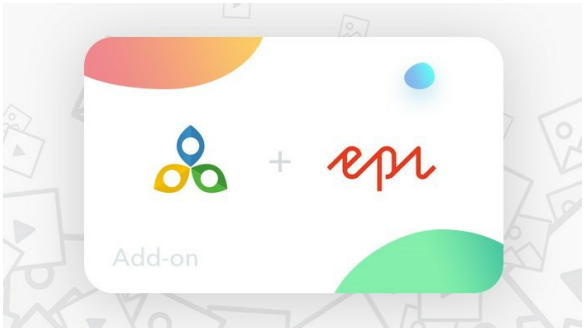
← Blog posts

Digizuite DAM for Episerver

DIGIZUITE, CMS, OPTIMIZELY (EPISERVER), INTEGRATIONS, ADDON DEVELOPMENT

Allan Thraen | 3 Years Ago | PDF |



Digizuite is a pretty serious DAM player in the enterprise market - and I have been lucky enough to be part of their DAM adventure in Episerver land. In this blog (and most likely several future posts) I will share some of the thoughts and approaches we have taken to make a good integration.

When I earlier this year announced that I would be going freelance, Digizuite were pretty quick at reaching out to me - and it turned out they had a really interesting project I could help them with as a freelancer. So, most of the time since I started out freelancing I have been helping them built an integration for Episerver.

What is a DAM and why do I need it?

Before digging in to all the fun tech, it might be worth to discuss the concept of a DAM for a moment. DAM (Digital Asset Management) systems are basically systems that keep track of all your digital assets across your channels. It's the one source that you can use to store product images, purchased stock photo, company videos, whitepapers, product descriptions, HR guidelines and everything you have, making it available on many, many different channels - the web site being just one. So - do you need it? Well, if your entire business is a single website running on Episerver and you are perfectly happy with the functionality provided there, then you certainly don't. But a lot of companies like the fact that they have one repository, with quality media they can use both on their website, their corporate presentations, in their print brochures and live on the sales guys phones when they are with customers, and in the corporate knowledgebase. All assets managed with access rights, meta-data, different media formats, streaming videos, standardized image crops and so on. And this is where a DAM can be very handy.

Digizuite

Digizuite is a Danish based DAM vendor that has been around for a number of years. They have from the start known that integrations and API's are key to success in a market where the DAM is but one of many components in an enterprise scenario that needs to be able to work with the other systems, and they have had great success with their DAM for Sitecore. Now, it's time to make them successful with Episerver.

Intended goals and features

The main goal is of course to make Episerver available as a valid channel for Digizuite assets - in a way so editors in Episerver easily can use assets from Digizuite in their web sites. But knowing how Episerver sites are built and maintained it's also essential to give implementing developers the right toolset to manage what is exposed to the editors, validate how editors are using it and customize how the assets are being shown to the visitors.

At the same time, we are aiming to improve the editorial experience in Episerver with some much needed features when working with assets, including:

- Easily finding the right assets - even when there are many, by efficient searching and filtering on free text, asset types, asset crops, and asset meta-data.
- Better navigable taxonomy - where an asset can be in multiple categories, that are easily browsed
- Automatically managing the media-format of the assets - and ensuring the right sizes are used in the right situations - either controlled by editors or developers implementing the assets
- Providing tools to let editors easily set focal points and create a predefined set of crops that are commonly used throughout the site
- Support efficient streaming of even quite large videos in multiple media formats, with automatic transcoding.
- Still allowing users to upload assets straight into Digizuite.
- Support many more asset types than those default available in Episerver - with automatic preview and handling of them
- Automatically added meta data - like Exif data and Geo data for images.
- Automatically convert and optimize images
- Maintaining relationships between original images and derived (Cropped) assets
- Multilingual image metadata.
- Much, much more.

The Integration

In future blog posts I plan to dig into the technical details of some of the architectural decisions we have made in building the integration as it is quite interesting and many of the concepts I think could be a good inspiration for others. For now, I'll just outline the main approaches.

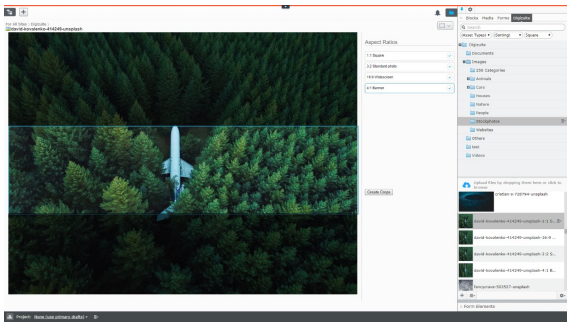
It has been essential for us to try to keep 'the Episerver way' of working with content - even if the content happens to be an asset in Digizuite. This is both to make everything recognizable for the editor - but just as much for the developers implementing the web sites. For example this means that developers are free to build their own strongly typed asset content models - just like they are

used to. The models and properties can then be mapped up to the Digizuite asset types and meta data using attributes.

The assets are essentially based off MediaData and implemented through a Content Provider that exposes both the assets and their taxonomy in the shape of a tree structure. The content provider is of course made somewhat resilient to connection issues to the Digizuite server - and it has a lot of additional caching to ensure a fast editorial experience. The actual binary assets themselves are exposed through a Blob provider - and so works pretty much as we are used to in Episerver - and both content provider and blob provider happily co-exists with others (pre-existing) content and blob providers in Episerver.

We're enhancing the UI in multiple ways - but again, using standardized Episerver extensions - like adding a browsing and search widget built in dojo, inheriting from the existing - and providing many more asset details than usual in custom views for those specific assets.

Todays blog post is mainly meant to be a teaser of more to come soon - but I will leave you with this screenshot (but still work in progress).



DIGIZUITE CMS OPTIMIZELY (EPISERVER) INTEGRATIONS ADDON DEVELOPMENT

Post Comments 0

Digizuite's announcement