

- [Digital Asset Management News](#)
- [About](#)
- [DAM Vendors](#)
 - [Vendor Submission Form](#)
- [Resources](#)
 - [Digital Asset Management Glossary](#)
- [Features](#)
 - [Whitepapers](#)
- [Contact](#)
- -
- -
- -

[Digital Asset Management News](#)

Digital Asset Management News, Reviews, Trends and Opinion

You are here: [Home](#) » [Special Features](#) » **Digital Asset Management And The Politics Of Metadata Integration**



Digital Asset Management And The Politics Of Metadata Integration

Last week in [Why Your DAM Needs Purpose And The Reason It Still Lacks It Now](#), I wrote about how the metadata schema employed in many DAM implementations is not always as relevant as it should be (from the perspective of DAM users). I used [David Diamond's CMSWire article](#) as inspiration for this piece.

We cover more discussion oriented topics on our [DAM News LinkedIn group](#) with the intention of sparking debate about articles and the subjects associated with them. Some get more responses than others, but this one provoked a number of interesting contributions from several group members. The subject forked off into a topic relating to integration of Digital Asset Management solutions with other applications. Since that is somewhat different to the original item, I am going to summarise the key points here with the objective of initiating a new discussion.

Background To DAM Integration And Its Relationship With DAM Metadata

One issue that many involved in DAM implementations will have faced already (or be about to in the

near future) is where data from an external third party system has to be integrated. To be clear about the frame of reference, I am not referring to other DAM solutions and would usually exclude other close relations like ECM or Document Management etc. Some suitable examples would be CRM, HR, specialist applications such as CMS (Collections Management System) and licensing/IPR databases. There are many others and you could include any system where the key entity is not an asset (and where that terminology would be considered unconventional by stakeholders).

In these scenarios, the external entity which contains the data of interest has an adjacent or perpendicular relationship with a digital asset. In other words, it is not above or below it in terms of the metadata schema hierarchy and needs to be treated independently (i.e. linked by association rather than part of the same record).

Example Use Cases

To give some examples: a DAM solution with photos of employees could be integrated with an HR system. This would enable more sophisticated aggregate functionality, such as automatically archiving or deactivating an employee photo asset if the associated member of staff leaves the firm, or retrieving current photos from the DAM system for display to users of the HR application. The staff HR record and the employee photo are independent of each other and different users have to work on them separately from each other to fulfil independent business functions.

To take a more complex use case: a CRM application that stores sales leads and interactions with prospective and existing customers. RFPs or other document assets submitted might be held in the DAM system so they can be searched for along with other sales and marketing collateral. Each of these are further composed of other assets, such as boiler plate copy, photos or videos of related projects. Having integrated data could yield some potentially useful business information, for example, a schedule of the assets used in a given response document to reduce the time required for sales personnel to source suitable assets or to avoid over-using a given image.

History Repeating Itself – Lessons From Digital Preservation and Collections Management

These type of intra-entity relationships are quite common in preservation and Collections Management Systems (and have been for decades). The methods for handling management of them, however, have not always been optimal, especially where different groups of stakeholders are involved.

Apart from the fact that Museum CMS projects are usually sponsored by public sector organisations and solution implementation funds are typically hard-won anyway, I suspect one of the other reasons this situation persists is that the pain of re-developing many of these applications properly from scratch is so great that few are willing to face it.

Having worked on a few migrations away from legacy preservation CMS in the past, I can appreciate that perspective. However, there appears to be a risk that the same mistakes will be made again but this time in the context of modern commercial DAM solutions. In the rest of this article, I will examine what can go wrong with DAM integrations and examine some possible methods to avoid the pitfalls.

What Can Go Wrong With DAM Integration?

One point that all the participants in the LinkedIn discussion were in full agreement on was that DAM solutions should be kept independent and focussed on what they do best: i.e. managing digital assets.

Anyone who has even a modest amount of experience of systems design instinctively knows that mixing functionality with different purposes in the same application is almost always a bad idea. The bugs, usability issues, training and all the other numerous issues that IT systems are subject to is made n times more complex when unrelated objectives get bundled together (where n=the number of different business functions the system in question is designed to offer). Our examination of the [DAM Value Chain](#) last year and exhortations to the DAM industry to focus on best of breed as the preferred delivery model for providing features to users all points to this same core theme.

With integration, the scope of responsibility widens to encompass not only both the technical teams involved in delivery of each separate solution, but also the people who briefed them. As such, there are

both technical and internal political reasons which can get in the way of integration best practice. Below are some reasons why DAM integration requirements can go awry:

- Monolithic applications that try to meet numerous separate business objectives.
- Bundling irrelevant functionality into a new system to escape an incumbent application which stakeholders do not want to use any longer.
- Differences of opinion about functional boundaries.
- Excessive duplication of data from related systems.
- Application hosting decisions and security restrictions.
- Misunderstandings and arguments between stakeholders about the nature of each other's data and how to cross-reference key entities successfully.

I have seen several of these in the aforementioned preservation systems and I am also observing similar patterns in corporate DAM integration projects now too.

Monolithic Applications

This refers to products that aim to address multiple business objectives and functional requirements within a single solution. These are not always as easy to spot as the term suggests. To start with, everyone knows that 'monolithic' is a pejorative term, so even if an application has got much more bloated than it ever should have done, it won't have that description applied, other than perhaps by someone who is completely new to it and unencumbered by the need to be polite towards the internal sponsors or vendors involved.

In the case of integration, monolithic applications become more likely when there is a suggestion that a single purpose Digital Asset Management system could or should be replaced by a multi-faceted suite that will seek to address many different requirements. Note that the design or architecture of suites is significant, if the elements are independent from each other (i.e. you can replace one part without ditching the whole thing) then that might be more tenable from a systems design perspective.

Alarm bells should ring if there are suggestions by stakeholders that aggregating requirements into some mega-system is a good idea. Some advice I was offered early in my career by an older (and wiser) colleague was that software applications never really get completed, people just decide to stop working on them for varying lengths of time. Bloated systems acquire an unstoppable momentum of their own which causes them to remain in use far longer than they ever should and you should always be aware if you are sleep walking into implementing one.

The Escape Pod Migration Project In Disguise

With a number of DAM integration projects I have been involved with, the obvious answer is for the users to use the services of the DAM system from within another application. For example, in the case of CRM, the user finds the sales lead then the system goes off to the DAM to find assets that relate to it.

This is simple and represents a clear distinction between both products, but the stakeholders are less keen. Why? It sometimes transpires that hardly anyone uses one of the two applications to be integrated and in reality, all they want is the data from it. Their method of addressing this is to transplant the functionality from the unofficial legacy into the other system (which is usually more modern and easier to use etc). Corporate DAM systems are often hosts for this kind of 'escape pod' migration because they are typically newer and less over-burdened with a sediment of legacy mistakes. This is one of many reasons why enterprise applications end up becoming bloated and monolithic and the best advice to stop it is to nip the problem in the bud early by having single purpose systems where the use case is easily defined and understood. If the application to be integrated with has problems, fix those first (or at least at the same time) rather than mixing up the purpose of what each should be used for.

Differences Of Opinion About Functional Boundaries

While it might appear obvious that the role of a given system should be kept tight, well-defined and not cross functional boundaries, you will find reasons which are plausible and well argued for doing the opposite. Consider the following by Jane Zupan of ECM vendor, Nuxeo, writing in CMSWire last year:

“Many different use cases for DAM technology have emerged in recent years, highlighting the need to track video, audio and image content...For a simple example, my car was recently rear-ended, and an insurance adjuster came to evaluate the damage. She took pictures and filled out an evaluation form on her tablet. Insurance claims in the past were managed on paper and stored in physical file folders, along with photos of the damage. Now, the paperwork, photos and videos are all digital content to be cataloged and stored for processing. In this context, it wouldn't make sense to store the photos in a DAM application and manage the case in a separate application.” [\[Read More\]](#)

I am not sure I would necessarily 100% agree with Jane, since having a separate DAM application to manage the photos would enable this aspect to be upgraded, scaled and extended independently of the case management function (including allowing it to be used for other purposes). However, I have based my critique on a three sentence description of the usage characteristics and she also qualifies her analysis with ‘in this context’. Whoever developed the application in question originally might have had compelling reasons for a single integrated application, or maybe they just did not consider it as an option.

I do have to note, however, that Nuxeo are pursuing a platform product strategy and their marketing output obviously will reflect that. Whether that is in your interests or not depends to a greater extent on the context (as Jane points out) but there will be a trade-off between the simplicity of having a single application which handles multiple tasks and the complexity required to modify the behaviour of it later on should your needs change.

Excessive Duplication of Data

One of the complications with DAM and integrating third party systems is the need to index these data sources, for example, with an HR system, you might need to store the department of the employee so it can be searched for. There are federated search systems available which claim to be able to index many different data sources (including enterprise application). I have not yet personally seen an example of one that works well enough to be usable for Digital Asset Management, however and the most common method to address this issue is to simply grab the data and index it within the DAM. The belt and braces method is serviceable, but it now means you have two copies, one in the original system, the other in DAM. If the updating between the two is comprehensively managed and diligently monitored then this may not pose a problem. If there is a glitch, however, then the potential for discrepancies between the two increases.

The ideal method for two integrated systems to exchange data is using some kind of mutually compatible service interface. Those with a technical background will have heard of REST and SOAP etc. For anyone who is less familiar, this means that each application connects to its integration partner to collect or send it data in much the same way that real users do, except without a graphical display (because the systems obviously don't require one). Although this method is still imperfect, there are common themes which make it more of a straightforward software engineering task rather than the proverbial rocket science.

In far too many integration projects that I have observed, the web services method either cannot be used (see the next point) or is discounted too quickly because the two implementation teams cannot agree with each other about how to handle it. There are other non-technical factors also, for example, if there is no funding for one half to implement alterations (or no one available to do it), they may be compelled to seek quicker and dirtier methods rather than doing the job properly.

The fallback old-fashioned way to do this (which probably hasn't changed for about 30-40 years or more) is to use delimited data files, such as CSV (Comma Separated Variable) which are imported on a scheduled basis (e.g. once per day). This method can go wrong in numerous different ways from simple failure of one party to deliver any data to the integrity of the data being compromised during export or transmission. Unlike dedicated integration services as described above, it is the responsibility of each partner application to verify that the data it is sending or receiving is safe to use.

Considering that we're now in 2014, this approach still seems to be rolled out on an alarming number of occasions. Like that phrase about old jokes being the best, they usually aren't and neither is using CSV data feeds to integrate two enterprise applications.

Application Hosting Decisions and Security Restrictions

This is a highly topical issue which I see a lot, especially where one system is considered fundamental or mission critical as compared with the DAM solution. It has some relationship with the previous point also, as I will explain.

Many DAM solutions are now externally hosted either using Cloud providers or just held on a server at some remote data centre utilised by the vendor. By contrast, applications like HR, finance etc tend to be held behind the corporate firewall on internal servers that the IT department manage. If the partner application is one such example, gaining access to the data it holds from an external system can be quite a challenge as IT usually will not want to take the risk of opening up external access just to service some niche application that is not considered business-critical.

This already starts to limit the range of options available for integration. The internal application can make requests to services provided by the externally hosted DAM (as most vendors now offer APIs) but that means the responsibility for transmitting and receiving data is vested wholly with the internal system. This implies that either internal development staff have to do the work or the application vendor's professional services team have to be paid to develop the integration, deploy and test it, along with whoever provides the DAM. Either of those options means time, money or both and the motivation to pursue the integration drops away as a result. This is usually where a less sophisticated option (like the delimited files referred to in the previous section) get suggested and they trade short-term off engineering complexity with increased long-term maintenance overhead.

DAM integration and hosting choices is a multi-faceted topic which I have covered in the past on [this guest blog post for DAM vendor, ADAM](http://blogs.adamsoftware.net/2013/08/06/damhostingenterpriseapplicationintegration/) and anyone who is interested may wish to read that item:

<http://blogs.adamsoftware.net/2013/08/06/damhostingenterpriseapplicationintegration/>

Misunderstandings And Arguments Between Stakeholders

One of the more politically charged aspects of integration is when it comes to arguments between groups of stakeholders about the impact of their requested integration with another application upon other users.

There are more straightforward problems like semantic misunderstandings where the stakeholders in one system do not realise how the metadata schema in the other fits together and therefore how best to relate to it (whether in respect of digital assets or other entities). But a more complex issue is where the functionality and associated UI (User Interface) has to be changed to incorporate an expanded scope. This appears to affect DAM systems more than the alternative scenario where digital assets get used in some other application since most of the time, users will now want to search for assets using metadata entities that are employed in the corresponding application. Existing users who may have little or no interest in metadata integrated from the third party application can become disoriented by the changes made. A further source of disputes is where the new stakeholders successfully persuade the implementation team to remove any incumbent search features they do not consider relevant to their needs (with the resulting flying sparks from those who do still rely on them).

There are methods to avoid this, for example, user profiling and showing/hiding search features or fields based on each user's characteristics. In the LinkedIn discussion referred to at the top of this article, some points were made about adapting or modifying metadata in a context-sensitive manner and as well as fields, search features are equally eligible for that kind of treatment.

To resolve disagreements between users, there are two essential requirements. Firstly, the DAM solution needs to be flexible enough to cope with multiple metadata schemas and be able to modify search interfaces accordingly also. Secondly (and in my view more importantly) someone on the implementation side has to have thought about the potential implications of the integration beforehand, asked the users in question and developed both strategies and tactics to address them to at least attempt to keep everyone happy.

Conclusion

Integration between DAM solutions and other enterprise applications is an essential evolutionary step in

ensuring Digital Asset Management becomes ever more widely understood and utilised across organisations – in particular in larger enterprises where user volumes are greater and the ROI opportunity is too good to pass up.

While there are some widely understood methods for successfully achieving this, if the delivery team is not strong-willed and fully aware of both the technical and political challenges which face them, the best laid plans can get blown off course and cost a substantial amount to correct. Having a clear grasp of not only the technical best practices but also the impact on (and reaction of) users and key stakeholders is essential to ensuring integration is implemented successfully and with the minimum of internal conflict.

\$49 LLC Formation

Form an LLC - 3 Easy Steps. Free 1st Year Registered Agent!



Leave a Comment

 Name * E-mail * Website

-
-
- **DAM Books**
 - [DAM Books](#)
- **DAM Glossary**
 - [DAM Glossary](#)
- **DAM Projects**
 - [DAM Projects](#)
- **DAM Vendors**
 - [DAM Vendors](#)

- **DAM whitepapers**

- [DAM Whitepapers](#)

- **LinkedIn**

- [DAM News LinkedIn Group](#)

- **Open Source DAM**

- [Open Source Digital Asset Management](#)

- **Twitter**

- [Follow us on Twitter](#)

- **Contents**

- [Archival and Storage](#) (7)
 - [Big Data](#) (2)
 - [Brand Asset Management](#) (2)
 - [Cloud Computing](#) (29)
 - [Copyright And IPR](#) (17)
 - [DAM Education](#) (12)
 - [DAM For IT](#) (2)
 - [DAM For Librarians](#) (2)
 - [DAM For Marketing](#) (10)
 - [DAM Interoperability](#) (3)
 - [DAM Kit](#) (3)
 - [DAM Projects](#) (1)
 - [DAM Skills](#) (7)
 - [Data Management](#) (4)
 - [Digital Asset Management](#) (53)
 - [Digital Asset Management Books](#) (19)
 - [Digital Asset Management Definitions](#) (5)
 - [Digital Asset Management Value Chains](#) (14)
 - [Document Management](#) (10)
 - [Emerging DAM Technologies](#) (17)
 - [Enterprise Content Management](#) (32)
 - [Enterprise Search](#) (16)
 - [General](#) (2)
 - [Industry Events](#) (112)
 - [Industry News](#) (101)
 - [Industry Standards](#) (24)
 - [Open Source](#) (41)
 - [Opinion](#) (101)
 - [Preservation](#) (13)
 - [Print On Demand](#) (2)
 - [ROI And Business Cases For DAM](#) (20)
 - [SaaS](#) (4)
 - [Semantic Web](#) (38)
 - [Sharepoint](#) (32)
 - [Social Media and DAM](#) (8)
 - [Special Features](#) (9)
 - [Streaming](#) (11)

- [Taxonomy And Metadata](#) (99)
- [Vendors](#) (205)
- [Video Asset Management](#) (40)
- [Workflow](#) (4)

• Recent Posts

- [Widen Release Media Collective Version 8.0](#)
- [DAM Innovation – Who Hit The Pause Button? \(Part 2\)](#)
- [DAM Foundation Ten Core Characteristics Update – Vendor Compliance Reviews](#)
- [DAM Innovation: Who Hit The Pause Button?](#)
- [Digital Photographs And Trustworthiness Of Social Media Platforms: Survey Results](#)

© [Digital Asset Management News](#) 2015 - All Rights Reserved.