



WHITE PAPER:

DAM 2.0: Collaboration & Project Management Control

Learn how digital asset management has evolved to address the needs of today's sophisticated workflows.

Digital asset management (DAM) today must support the needs of organizations that:

- Collaborate on project development—locally or globally,
- Rely on approvals and communication for quality control, and
- Base access and distribution on complex, variable conditions.

The Call for DAM 2.0

Some digital files are little more than random data, but others can hold significant value: They are the success of your current campaign, the confidentiality of your next strategy, the integrity of your brand, or more. The loss or erroneous release of even one of these files can be embarrassing, expensive—or worse.

This document details the needs of organizations that understand the business dangers of erroneous asset distribution, and aim to avoid the problem altogether. In addition, it outlines ways in which the Cumulus digital asset management system can help organizations control their assets at the project level in order to avoid mistakes.

It's easy to classify erroneous asset distribution into one of three categories:

- Content errors — Errors in content that were not caught during edits and approvals.
- Approval errors — The content might be technically correct and typo-free, but it's not appropriate for other reasons—sensitivities, timing, etc.
- Distribution errors — An error-free, approved asset was distributed before an embargo, or it was sent to the wrong recipients, etc.

Typically, organizations don't think of DAM solutions as being able to help them avoid errors like these. "DAM" to most organizations, means little more than "advanced search engine."

And though it's true that some DAM systems offer search capabilities far beyond those available at the operating system level, today's DAM systems must do more. What was once merely about *finding* files, is now also about defining, controlling and tracking an asset's content development—and its history and future.

Today's DAM systems must address these needs in order to remain viable, and they must be flexible enough to meet the needs of tomorrow. After

all, can anyone reasonably imagine the flow of digital information will become any less complex—or less important—in years to come?

This is the call for DAM 2.0.

Rethinking Digital Assets

Operating systems encourage us to think of files in terms of *file name*, *modification date*, *folder location*, etc. Computer users have found files based on simple metadata like these for so long, we've adapted to the limitations they impose.

But do metadata like these properly and adequately describe your digital assets?

Imagine "Twitter" for Assets

To put some perspective on the notion of adequately describing assets, think of someone close to you—a family member or a friend. Now, imagine describing that person using operating system metadata:

- Name: David
- Creation Date: 12/19/1963
- Size: 190 pounds
- Type: Male
- Location: Alameda, California, USA

To an operating system, this is a "complete" file description. Add a thumbnail, and that's pretty much all you have to work with.

But do these values offer any useful sense of who "David" really is? To really understand David, you need to know more: Where did he come from? What does he know? What can he do? Where has he been? Where does he need to be tomorrow? The breadth of information you could collect is infinite. And the more you know about David, the more "value" he can offer you—just like your digital assets.

Have you ever needed to open a file because you couldn't determine its contents based on a file name alone? Have you ever forgotten from whom or where you got some file in the first

place? What about the projects in which it's already been used? What about licensing restrictions? Or production status?

Now, consider the passage of time. Over time, David grows and changes. As his friend, you update what you know about him in each conversation you have. But if you don't keep in touch, the next time you talk, David might seem like a completely different person—nothing like you remember.

So what about updating the metadata you assign to your files?

Even the metadata for a file you *haven't* used in years needs to be occasionally updated to reflect, say, the fact that you haven't needed it in years. Should you pay to renew your usage license for the asset? How would you know, if you don't know when your organization last needed it? What about renewing your contract with a designer? How often was her work chosen over the work of another designer?

Metadata values enable you to "know" your assets just like you know your friends. Metadata offers the information on status, history and future you need to "keep in touch" and make sure your assets always remain your best friends.

Managing Ours vs. Mine

When a file has been created by you, you might know information about it that others wouldn't, such as what the file contains, and who has already seen it. You might even have sticky notes on your computer monitor that remind you about what you need to do to complete files that are still in development.

All the little tricks you use to help you get your work done define your personal workflow. And if you work alone, your personal workflow probably works pretty well for you—at least for files in recent memory. But when you collaborate on files with others, sticky notes don't tell the whole story—and they're tough to share.

A fully featured DAM system can be used as the “gatekeeper” that enables you (and others) to track the asset (and project) information you need to maintain control over your development pipeline. If you choose the right DAM system, and configure it to properly suit your needs, your system can save you considerable time and money.

DAM systems that don’t offer adequate metadata management options are as effective (and dangerous) as air travel with no air traffic control. You need to know where files are going; you need to know when they’ll arrive; you need to know who has control of them at any given moment—and so does everyone else working in the same collaborative “air space.”

Control is the goal; chaos is the consequence.

To establish and maintain control over your collaborative project and asset development, you need to start with two things:

- A defined process by which your files are developed, and
- A system that supports and enables that process, and helps you manage it.

The Power of Process

The process behind an organization’s asset development (or acquisition) is often taken for granted by smaller groups working together on a regular basis—everyone knows how everything is done, so no one gives the process itself much thought.

But when someone unfamiliar with the process comes into the fold, problems arise. A new employee requires some training—easy enough—but what about a new client? What about a new corporate partner? Or even a new print house?

A secure, efficient development process is only as strong as its weakest link—if one person makes an error, the entire process can fail.

An established process helps ensure content quality and timely delivery; but even better, it can significantly reduce (or eliminate) the errors described at the start of this document.

Defining Process

The most obvious first step to take when defining a DAM-assisted development process is to analyze your existing process. Each situation is unique, but most asset development follows the steps shown in figure 1.

Depending on your organization’s existing processes, other steps not shown in figure 1 might be needed. Most important to identify are the steps your organization takes between initially

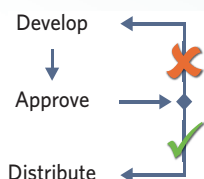


FIGURE 1: Digital asset development usually follows a series of edits and approvals before final distribution.

determining an asset is needed, all the way through to the asset’s distribution. This information will help you determine the metadata your organization will need to integrate DAM into your development pipeline.

The Value of Metadata Flexibility

Before you can identify all the metadata values you’ll need in place to keep track of things, it’s helpful to consider the various metadata options available, and how you can apply them to your processes.

Tags vs. Metadata

Tagging describes the process of assigning individual words or phrases to a given file. Collectively, tags “paint a picture” of the file’s content—*flower, yellow, rooftop garden*, etc.

And though a file might be a picture of a *flower* that is *yellow* and planted in an *rooftop garden*, the management of that file involves knowing more about the file than what it contains. Consider the following attributes:

- 10MB in size
- Ready for approval
- Pending license verification

Attributes like these in no way describe the content of a file. What’s more, these values will change throughout the asset’s lifetime. This is where proper metadata fields come into play. A metadata field provides a placeholder for information. Some of that information might never change, but some might change regularly or even constantly.

Tags are cumbersome for managing variable attributes, because tags are static text values. If you assign the value *Ready for approval* as a tag, when the status changes, you need to remove the old tag and add the new one. If you forget either step, you’ll have confusing situation on your hands. In other cases, such as file size, tags make no sense at all.

Even worse, searching for files based on tags doesn’t always yield expected results. Search for “apple” and what will you find: fruit or computers? And what happens when, in your part of the world, “apple” is *Apfel* or *pomme*? When using tags, *spelling counts!* (Figure 2)

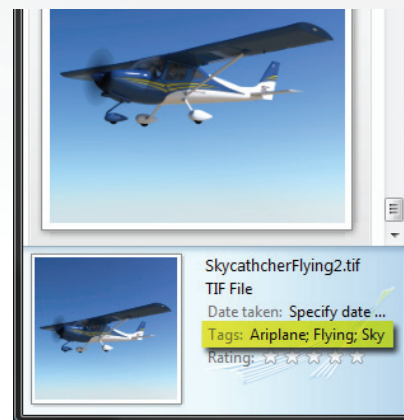


FIGURE 2: Tags offer a limited way to track files. Notice the misspelled “airplane” tag. This is one asset that might not be found when needed!

Using a tagging system is like having a digital asset management system that offers only one metadata field: *keywords*. Tags can be handy when sorting personal files in an operating system, but tagging is not a solution for digital asset management.

Projects Count Too

In addition to individual assets, projects can also be described by metadata. Examples of metadata that would be useful when describing a project include:

- Due tomorrow
- Managed by Uli
- In approval

These metadata values describe a project as a whole, not the individual assets associated with the project. The following table offers examples of metadata that make sense for assets, projects or both.

Metadata	Asset	Project
File Format	✓	✗
Status	✓	✓
Client Name	✗	✓
Publication Edition	✗	✓
Creator	✓	✗
Manager	✗	✓
Due Date	✓	✓

In order to manage projects using your DAM system, it must offer the following functionality:

- A way to define projects — There should be no limitation on the number of projects you can define and manage.
- A way to assign metadata values to projects for the purpose of tracking status, etc. — There should be no limitation on the number of metadata fields you can use.
- A way to assign assets to projects. — There should be no limitation on the number of projects an asset can be assigned to, and no limit on the number of assets a project can be assigned.
- A way to find projects (and the assets they contain) — Multiple methods, including browsing and searching should be available.

Some metadata value names, such as *status* or *due date*, are useful for describing individual assets and entire projects. Your DAM system must enable you to reuse metadata value names when it makes sense to do so, all the while ensuring those values remain unique. For example, when developing a book, you might have a *status* value for the book cover art and another *status* value for the book as a whole. Though named the same, the values of each *status* value will not remain in sync throughout the development, so they must be treated separately.

NOTE: In most DAM systems, a single metadata value is represented by a corresponding metadata field.

In addition, your DAM system must enable you to add as many metadata values (fields) as you need to manage your assets and your projects.

Thinking of (Metadata) Fields

All this in mind, take another look at figure 1. The steps depicted are simple, but the metadata values required to represent each step, and communicate progress between collaborators, are diverse. The following table lists metadata field examples that could be used to track the workflow figure 1 outlines.

Fields	Purpose
Asset Status	A <i>status</i> field would define development status. Users and managers could use this field value to determine “todo” lists, or to check development against schedules. <i>Minimal Functionality</i> The field should enable users to enter the words used to describe their status values.
Project Status	<i>Ideal Functionality</i> The field should enable users to choose status values from a menu or other UI widget that makes metadata changes easy and error free.

Fields	Purpose
Asset Creator	A <i>Creator</i> or <i>Manager</i> field would include the name of the person or department in charge of creating and editing the asset, or managing the project. This information is useful for knowing who to contact with questions, and it can also be used to serve notifications, such as when an approval has been completed and edits are required.
Project Manager	<i>Minimal Functionality</i> The field should enable users to enter the name of a person in charge of development or management. <i>Ideal Functionality</i> The field should provide an easy and error-free way for users to assign <i>one or more</i> users.
Asset Approver	As with the <i>Creator/Manager</i> field, this field identifies the person authorized to perform approvals for the asset. <i>Minimal Functionality</i> The field should enable users to enter the name of a person who will approve the asset or project.
Project Approver	<i>Ideal Functionality</i> The field should provide an easy and error-free way for users to assign <i>one or more</i> users.
Asset Approved	This field can serve as a “digital signature” that indicates when an asset or project has been given final approval. Approval could also be determined by a value in the <i>status</i> field, but using a separate field to indicate final approval offers some advantages, such as having a final “sign off” field that’s available only to an authorized manager.
Project Approved	<i>Minimal Functionality</i> The field should enable the person who has the authority of final approval to sign off on the asset. <i>Ideal Functionality</i> The field should provide a clear indication of approval (such as a check box), and access to the field should be restricted to only those authorized to sign off on the asset or project.

Metadata fields can also be used to convey details, such as why a project was delayed, or where an asset should be distributed once completed. They could even provide a checklist of the development process that can help users move the asset or project through the milestones of the development pipeline. (Figure 3)

In summary, metadata values (and the fields that store them) must be flexible enough to serve two purposes:

- Indicate an intention, association or state of being (size, status, value, etc.), and
- Serve as the foundation for automation that can help keep production on schedule, which is described next.

Metadata that Automates

Some DAM systems can use metadata values as the basis for automated tasks, such as emailing

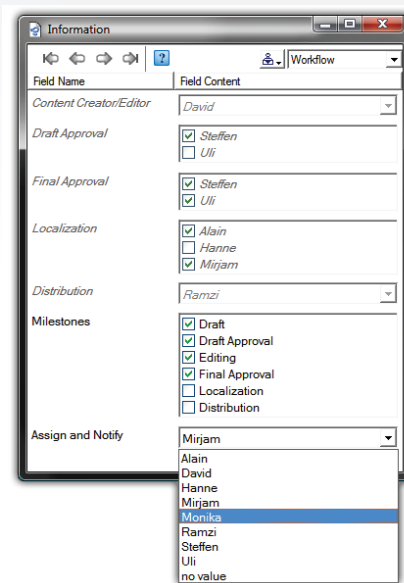


FIGURE 3: The fields at the top of this window define who is responsible for each step of the development pipeline. They are disabled in this view because the connected user lacks permission to edit them. The *Milestones* and *Assign and Notify* fields are editable to this user. Note how some of the fields permit more than one selection using check boxes, while others provide drop-down menus from which only one choice can be made. User interface widgets like these make metadata entry fast and accurate.

notifications, distributing files, removing access to expired assets, etc. DAM system automation is typically based on a trigger/action paradigm— a *trigger* occurs that causes an *action* to take place. Triggers are generated by users events, such as changing the value of a metadata field, or they can be based on recurrent time schedules.

Figure 3 shows a field that’s used for the assignment and notification of an asset or project. When a new value is chosen in this field (menu), the DAM system would send an email notification to the selected user.

In order to perform tasks like these, your DAM system needs the pertinent information, which means you’ll need to think about the metadata fields you’ll need to provide that information. To better anticipate the information you’ll need, think of your DAM system as a personal assistant.

When you assign your assistant a new task, what information do you need to provide?

The following table shows examples of questions an assistant would ask, based on tasks assigned.

Task	Assistant asks
Email notifications of status changes.	<ul style="list-style-type: none"> • How will I know when a notification is needed? • Who should I notify?
Change access permissions when needed.	<ul style="list-style-type: none"> • How will I know when an access permission change is needed? • What changes should I make?
Distribute a file when it's ready.	<ul style="list-style-type: none"> • How will I know when a file is ready for distribution? • Where should I distribute it?

Thinking automation through from the perspective of an assistant helps you better determine the metadata fields you'll need in order to control your production pipelines.

Rules for Access

One final consideration with regard to metadata field choices is *access*. Asset security is based on access permissions, so this is not a subject to overlook.

Some background: Computer operating systems restrict access to simple yes/no determinations—you either have access to a directory or file, or you don't. Some flexibility is added by providing permissions based on groups in addition to individual users.

This permissions model was revolutionary when it was originally introduced—*over 30 years ago!* And though this model is still in use by virtually all operating systems in use today, your DAM system should free you entirely from this limitation.

The problem with this model is that access is based on a static decision: *yes* or *no*. Not many permissions in life are granted this way. Consider the following permission request:

"Mommy, may I go out and play?"

Sometimes the answer to this question might be a simple yes or no; but most often, rules apply:

"You may go out only after you've finished your homework. And make sure you come back inside before it gets dark."

This is a *rule-based* permission, meaning the answer is conditional. If any condition is not met, permission is not granted. (Or the permissions is revoked, if previous granted.)

Operating systems don't support this advanced permissions model, which forces organizations that rely on operating systems for digital asset management to make access decisions manually.

Should we post the press release on the website?

Would the answer to this question ever be a simple yes or no? Even when the final "yes" comes in, it's the result of preconditions that have each been satisfied—the press release has been approved; the embargo has expired, etc.

Your DAM system should enable you to automatically limit access to assets for certain users or groups, based on rules you define. It should also be capable of revoking permission based on those same rules.

When you determine the rules you need, you'll also be able to determine which additional metadata fields will be required for your DAM system to make those decisions. Example rules and the fields required to define them are shown in the table below.

Rule	Metadata fields required
Distribute file after embargo expires, only if approved by manager and CEO.	<ul style="list-style-type: none"> • Manager Approval (check box) • CEO Approval (check box) • Embargo (date)
Limit access to file to certain individuals and groups only.	<ul style="list-style-type: none"> • Permitted individuals (check boxes) • Permitted groups (check boxes)
Show only those files rated as 3-stars or better.	<ul style="list-style-type: none"> • Rating (1 to 5 stars)

The examples given throughout this section show how metadata values can describe far more than just a file's contents.

Benefits Over Time

Most DAM system benefits are accumulative. For example, just after launch, a DAM system is little more than something new to learn. But within months, a good DAM system can become as important to your operation as the electricity you use to run it.

The immediate challenge lies in convincing skeptical users why DAM matters. Dentists know this struggle well: "Floss your teeth!" is a mantra many don't appreciate until later in life, when the benefits become clearer.

The need for digital asset management is similar in concept, and the conveyance of its benefits can be similarly difficult. Programs like Photoshop or PowerPoint demonstrate value within minutes of installation; but the first step after installing a DAM system is to catalog (index) your files—not too rewarding. (Though you are taking that important first step toward never losing another file!)

And while launch day for a new DAM system is typically neither remarkable nor typically memorable, soon comes next week and next year. In the

meantime, your organization has amassed thousands of new files, and you have scores of additional projects to archive—not to mention tracking the development of current ones.

The DAM system you installed to little or no fanfare a year prior, has since become your organization's single portal to absolutely everything. Gone are the days of wading through file server folders, or sifting through printouts of assets archived offline. There are no more complaints about file server logins or permissions, and no file has been lost, inadvertently edited or prematurely released. If you need the status of anything, you know exactly where to go—24/7—using your computer, someone else's computer, or even your smart phone.

"Digital dental floss" will change everything for your organization, and in far less time than you might expect.

ROI in Retrospect

Looking back over your first year using DAM, the benefits you'll most easily recognize are:

Increased production efficiency Designers and sales personnel spend far less time finding files for layouts and presentations. Files are no longer scattered across computers and locations, and no file is beyond reach from anywhere. Delays are reduced and quality improves, because "second choice" assets are never used in place of preferred assets that couldn't be found. Mistakes and "redos" are avoided, because users can easily tell when a file is approved, suitable and current. But, perhaps best of all, when your sales team saves time, they can use that time to generate additional revenue—they spend time finding leads, not files!

Reduced approval roadblocks Approvals require less time and come faster, because a single resource has become the communications and access hub for all development. Managers know where to go to check the status of anything, and they can grant approvals or submit revision instructions from home, a hotel, or even 35,000 feet.

Decreased time to market Projects get to market faster because productivity goes up, while delays go down. It's also easier to efficiently handle more projects simultaneously—and faster—thanks to the greater information and control your DAM system has made available to you.

Investment protection Time and money are saved because no asset needs to be recreated or re-licensed. Investment ROI is maximized across your entire organization, because one department doesn't waste money acquiring asset

licenses already purchased by another department.

Litigation avoidance It's easy for users to know when a restriction is placed on the use of an asset, so you avoid license violations that could lead to costly and embarrassing litigation. What's more, you know that the assets you acquired in 2008 will be properly and legally used by designers in 2010, because usage information remains intact and easy to find—even after everyone's long since forgotten where the assets came from.

Unforeseen Advantages

Additional benefits depend on how an organization uses its DAM system. Below are actual examples from organizations using DAM today:

- A book publisher claims a savings on erroneous shipping charges, because fulfillment personnel can visually confirm orders based on book cover thumbnail images, which are easier to verify than inventory numbers.
- A supermarket chain uses its DAM system to quickly confirm that product shots planned for its weekly flier are accurate, and not the same shots recently published by competitors.
- An advertising firm has developed an entirely new business model based on archiving the historical assets of clients. When clients need images from legacy ad campaigns, they know where to find them.
- A software publisher uses its DAM system to track the development of marketing and training materials in three different languages. English materials are produced in the United States, then localized in France and Germany. Employees and free-lancers are involved, and everyone has access to the same materials—without cumbersome VPN or unsecured remote access clients.

Metadata Standards 101

This section offers a basic overview of metadata sources and standards, which is important to understand in order to gage the strengths and weaknesses of any DAM system.

Metadata Sources and Standards

Asset metadata typically originates from one or more of the following sources:

- Hardware devices (digital cameras, scanners, etc.)
- Operating systems (file name, creation date, etc.)
- Software programs (Photoshop, PowerPoint, etc.)

In some cases, metadata is gathered and stored into fields that reflect *metadata standards*, which are agreed upon conventions used to describe assets. The following table lists the most popular metadata standards.

Metadata Standard	Description	Example Metadata
EXIF Exchangeable Image File Format	The standard used by virtually all modern digital cameras for metadata capture and storage. Most applications that deal with digital photos support EXIF.	<ul style="list-style-type: none"> • Shutter speed • Camera model • GPS shot location • Shot date
IPTC International Press Telecommunications Council	A metadata standard originally developed to describe photos used by news agencies. Most applications that deal with asset metadata support IPTC.	<ul style="list-style-type: none"> • Caption • Headline • Location • Keywords
XMP Extensible Metadata Platform	XMP is a framework designed by Adobe to, in part, encapsulate previous metadata standards and overcome the limitations of "closed" standards. Adobe Creative Suite and other applications support XMP.	The metadata accessible via XMP varies between applications, but most common fields are globally supported.

Metadata values created by operating systems, such as file name, size, creation date, etc., are always stored inside files, regardless of the file's format. These are the metadata values you see when you look at a file's properties or info window. (Figure 4)

Many file formats also support the embedding of additional metadata inside files. For example, Microsoft Word enables users to add metadata, such as an author's name, which is saved inside the file. QuarkXPress and InDesign would additionally store a page count and list of fonts used in the file. The information that can be embedded depends on the file type.

Some file formats support the storage of entire metadata standards. JPEG and TIF are two popular file formats that support all metadata standards listed in the previous table. One advantage of this is that you can "write back" IPTC/XMP metadata into your JPEG and TIF assets so that it travels along with those files no matter where

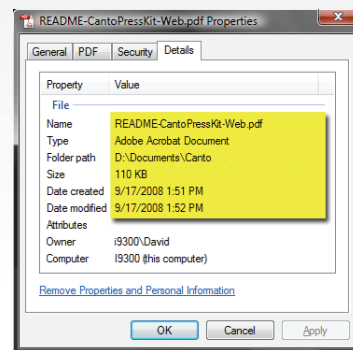


FIGURE 4: The Properties window of this file shows metadata that's embedded into the file by the operating system. DAM systems extract this information (and more) when the file is cataloged.

they go. (EXIF metadata describes conditions the moment an asset was created, such as shot conditions and camera information. In order to maintain the integrity of that metadata, this metadata should *not* be editable.)

It's not possible to write-back metadata into file formats that don't support it, but it's still possible for metadata to "travel" with assets through the use of external files that act as storage containers for the asset's metadata. (Figure 5)

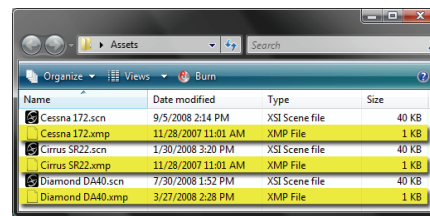


FIGURE 5: External files are used to store metadata for file formats that do not support metadata embedding. Here, "XMP Sidecar" files store XMP metadata used to describe Autodesk Softimage XSI scene (.scn) files, which do not natively support XMP.

Standards: Good, but Limiting

What if none of the metadata standards provide fields suitable for all the information your organization needs? For example, say you work in a research facility whose work is supported by government grants. You're required to track assets based on the grants that fund their development, but none of the popular metadata standards include fields for "Grant" or "Funding Resource."

Some organizations might just repurpose metadata standard fields they don't need. For example, they might use the XMP "Description Writer" field to track grant resources, because they don't have a need to track the writer of their asset descriptions. Though this seems like a

clever solution, it can lead to confusion—and can even data loss!—if an editing application assumes the field is used as intended by the metadata standard. (Figure 6)

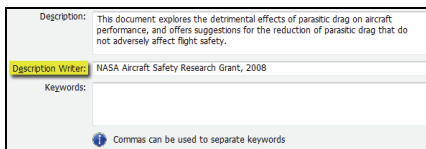


FIGURE 6: The Description Writer field has been used here to store the name of a funding resource. This is confusing, and it could lead to data loss if a “real” description writer value is ever added by an editing application.

When a metadata standard is used in a manner inconsistent with its intended purpose, you defeat the concept of “standard.”

Instead, your DAM system should be flexible enough to enable you to add any and all metadata you need, and to use those fields right alongside any “standard” fields in use.

NOTE: For ongoing tips and “best practice” information on DAM, regularly visit the Canto Community Blog: <http://blog.canto.com>

How Cumulus Helps

Now, with a more solid understanding of the requirements collaborative development places on a DAM system, it’s easier to appreciate the various capabilities Canto Cumulus offers to address those needs.

Cumulus Metadata Flexibility

Cumulus supports all popular metadata standards. It also enables you to add and track additional information *not* supported by any metadata standard. In other words, Cumulus offers the best of both worlds.

As a primary means of metadata storage, Cumulus uses *catalogs*. (Think of catalogs as metadata databases.) The important advantage of this design is that assets don’t need to be physically available in order to be found through searching. So, offline archives remain completely searchable, just as if they were online. This makes it easy, for example, to catalog a collection of CDs or DVDs, knowing the discs themselves will not remain online. When someone searches for a file that’s on one of those discs, all of the metadata will be available (and editable), and Cumulus will even indicate which disc the file is on.

In addition to catalog storage, Cumulus can write-back metadata into file formats that support it, and it can create “tag” files to locally store metadata for file formats that don’t support metadata write-backs. (Cumulus tag files are

text files, such as those shown in figure 5, that include asset metadata. Tag files are not specific to any metadata standard.)

So, if your organization has standardized on IPTC or XMP, but some of the file formats you use don’t support those standards, Cumulus solves the problem for you. Or, if you use custom (non-standard) metadata and you need to write those metadata values out to external files, Cumulus can do that too.

DAM systems that store metadata only inside asset files are of limited use, because you cannot edit a cataloged file’s metadata unless that file is available *and* not in use by another application. Plus, one of the main advantages of a DAM system is to help you find files. If your only hope for finding a file lies within the file itself, that’s one DAM benefit lost—along with the file.

Capturing Metadata

Metadata comes into Cumulus catalogs in one of two ways:

- Cataloging your files (indexing)
- User editing

During a cataloging operation, Cumulus extracts metadata from your assets and stores that information in your Cumulus catalogs. (Cumulus catalogs do not store your actual assets—those remain stored on the media of your choice, such as a file server, ftp site, etc.) The metadata Cumulus captures depends on each file’s format and your Cumulus configuration settings.

Cumulus can catalog any digital file, because it collects the “core” metadata about files that’s provided by the operating system. This is the file name, format, size, storage location, etc. you see inside a file’s properties or info window, as shown in figure 4. These values are considered “core” because they are applicable to all file formats.

In addition to these core metadata, Cumulus can extract additional information for hundreds of the most popular file formats. Canto refers to this as *enhanced metadata support*. When Cumulus offers “enhanced” support for a particular format, that means it can find and extract metadata beyond what the operating system provides. Examples of enhanced metadata values for various file formats follow.

File Format	Enhanced Metadata Examples
AVI, MOV, WMV	Duration, FPS (Frames per second)
TIF, JPEG, TARGA	Color Mode, Resolution, Bit Depth
AIF, WAV, MP3	Duration, Sample Rate

File Format	Enhanced Metadata Examples
InDesign, QuarkXPress, Microsoft Word	Page Count, Fonts Used

The “enhanced” metadata a given file format includes depends on the information required to describe assets of that format. Example: video and audio files have a *duration* metadata value, whereas desktop publishing and word processing formats include *page counts* and *font* information.

Cumulus captures this information and stores it into catalog metadata fields that have been allocated for it. For example, say a catalog contains a metadata field called *Duration* that’s mapped to the *duration* metadata inside your assets.

Canto calls this mapping capability *field linking*, and it’s an extremely useful feature unique to Cumulus. The real power of field linking comes not from serving as a metadata “traffic cop,” but from enabling users to easily change the default mappings, and even add new ones. So, for example, all the information stored in your various file formats as *notes*, *comments* and *information* can be copied into a single field inside Cumulus. This means, no matter what the file format, the file’s “about” information will always be found in the same place.

This is handy because different metadata standards and file formats often use different names for what is essentially the same metadata. What one format calls *duration*, another calls *length*. Cumulus field linking enables you to handle it all with a single metadata field.




Metadata Field Types

Each Cumulus catalog field has a *type*, and the numerous metadata field types Cumulus offers is the basis for much of the power the program offers.

Beyond the text and number field types you would expect, Cumulus offers a number of additional field types that make collaborative workflows work better, as seen in the table below.

Field Type	Description
Audio	<p>Communicate verbal ideas between users. Use your computer’s microphone to record input.</p> <p>Usage ideas:</p> <ul style="list-style-type: none"> • Audio captions for images used in information kiosks, or on websites. • Verbal approvals.

Field Type	Description						
Boolean	<p>Use Boolean fields for yes/no values. They're easy to use, easy to find and easy to understand.</p> <table border="1"> <thead> <tr> <th>Field Name</th> <th>Field Content</th> </tr> </thead> <tbody> <tr> <td>STOP DISTRIBUTION!</td> <td><input checked="" type="checkbox"/></td> </tr> </tbody> </table> <p>Usage ideas:</p> <ul style="list-style-type: none"> • Mark assets or projects "Approved." • Provide an emergency "kill" switch for asset distribution. • Use multiple Boolean fields as checklists for development stages or pre-flighting. 	Field Name	Field Content	STOP DISTRIBUTION!	<input checked="" type="checkbox"/>		
Field Name	Field Content						
STOP DISTRIBUTION!	<input checked="" type="checkbox"/>						
Date / Time	<p>Date/Time fields enable you to search and sort based on date and time references. Example: search for "7 days from now" (\$Today+7) or "one hour ago" (\$Now-1).</p> <table border="1"> <thead> <tr> <th>Field Name</th> <th>Field Content</th> </tr> </thead> <tbody> <tr> <td>Embargo</td> <td>12/19/2007 12:01:00 AM</td> </tr> </tbody> </table> <p>Usage ideas:</p> <ul style="list-style-type: none"> • Use embargo dates, release dates, due dates, etc. • Indicate forthcoming expirations that help users determine the "shelf life" of projects in development. • Find assets cataloged in the past x hours. 	Field Name	Field Content	Embargo	12/19/2007 12:01:00 AM		
Field Name	Field Content						
Embargo	12/19/2007 12:01:00 AM						
Label	<p>Label fields offer a visual indication of any status you need. The number of labels and the color of each is determined by you. Find and sort assets and projects based on label values.</p> <table border="1"> <thead> <tr> <th>Field Name</th> <th>Field Content</th> </tr> </thead> <tbody> <tr> <td>Priority</td> <td> </td> </tr> </tbody> </table> <p>Usage ideas:</p> <ul style="list-style-type: none"> • Determine development priorities. • Make indications using color key conventions standardized in your organization. 	Field Name	Field Content	Priority	 		
Field Name	Field Content						
Priority	 						
Rating	<p>Rating fields can be used for any "0 to 5 star" ratings you need. Find and sort assets and projects based on ratings. Use as many rating fields as you need.</p> <table border="1"> <thead> <tr> <th>Field Name</th> <th>Field Content</th> </tr> </thead> <tbody> <tr> <td>Art Director</td> <td>★★★★☆</td> </tr> <tr> <td>Editor</td> <td>★★★★☆</td> </tr> </tbody> </table> <p>Usage ideas:</p> <ul style="list-style-type: none"> • Assign different rating fields to different users to see the results of all. • Use rating values to limit the assets shown to clients. • Indicate the approximate value of assets in a manner that's easy to see and compare. 	Field Name	Field Content	Art Director	★★★★☆	Editor	★★★★☆
Field Name	Field Content						
Art Director	★★★★☆						
Editor	★★★★☆						

Field Type	Description																				
Picture	<p>Cumulus stores thumbnails for many file formats automatically, but you can add additional picture fields of your own. Use copy & paste to get your images into these custom fields from other programs.</p> <table border="1"> <thead> <tr> <th>Field Name</th> <th>Field Content</th> </tr> </thead> <tbody> <tr> <td>Book Cover</td> <td></td> </tr> </tbody> </table> <p>Usage ideas:</p> <ul style="list-style-type: none"> • Use a book's cover art to represent its project. • Store multiple thumbnail images to provide a scene-by-scene overview of a cataloged video. • Store a visual copy of a signed release or contract along with the asset or project it affects. 	Field Name	Field Content	Book Cover																	
Field Name	Field Content																				
Book Cover																					
String List	<p>String list fields contain a collection of values from which users make selections—just like the menus used in operating systems. String list fields can be single choice or multiple choice. You can use as many as you need, and you can search and sort based on any values they contain.</p> <table border="1"> <thead> <tr> <th>Suitable For</th> <th>Field Content</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>Children</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>Teenagers</td> </tr> <tr> <td><input checked="" type="checkbox"/></td> <td>Adults</td> </tr> </tbody> </table> <table border="1"> <thead> <tr> <th>Workflow status</th> <th>Field Content</th> </tr> </thead> <tbody> <tr> <td>To Do</td> <td><input type="checkbox"/></td> </tr> <tr> <td>In Progress</td> <td><input type="checkbox"/></td> </tr> <tr> <td>Ready for Review</td> <td><input type="checkbox"/></td> </tr> <tr> <td>In Review</td> <td><input type="checkbox"/></td> </tr> <tr> <td>Ready for Production</td> <td><input checked="" type="checkbox"/></td> </tr> </tbody> </table> <p>Usage ideas:</p> <ul style="list-style-type: none"> • Indicate audience segments suitable for a given asset. • Redirect assets to the attention of a coworker. • Provide drop-down, radio button or check-box access to any pool of values for quick, error-free edits. 	Suitable For	Field Content	<input type="checkbox"/>	Children	<input checked="" type="checkbox"/>	Teenagers	<input checked="" type="checkbox"/>	Adults	Workflow status	Field Content	To Do	<input type="checkbox"/>	In Progress	<input type="checkbox"/>	Ready for Review	<input type="checkbox"/>	In Review	<input type="checkbox"/>	Ready for Production	<input checked="" type="checkbox"/>
Suitable For	Field Content																				
<input type="checkbox"/>	Children																				
<input checked="" type="checkbox"/>	Teenagers																				
<input checked="" type="checkbox"/>	Adults																				
Workflow status	Field Content																				
To Do	<input type="checkbox"/>																				
In Progress	<input type="checkbox"/>																				
Ready for Review	<input type="checkbox"/>																				
In Review	<input type="checkbox"/>																				
Ready for Production	<input checked="" type="checkbox"/>																				

Collectively, these fields types (along with standard text and numeric options) enable you to define powerful and flexible project management and collaboration controls that reflect the way your organization works.

Process Flexibility and Forgiveness

The term "flexibility" when applied to Cumulus means more than just having options. Cumulus flexibility includes the following benefits:

Out-of-the-box readiness Cumulus installs in minutes, and any organization can start using it immediately in its default configuration.

Programming-free configuration Configuration changes made to Cumulus can be done without a single line of programming—ever! Configuration options are set via user-friendly user interface items, such as menus, buttons and check boxes. (Figure 7)

One-stop configuration You can configure a Windows-based Cumulus Server from a Mac, or vice versa. You can even configure a Cumulus Server running on Linux from a Mac or PC. Plus, all configuration changes you make are automatically honored by the Cumulus Web clients. So you never need to re-do work, or worry that your Web users won't see what your LAN users see.

Experimentation-safe operation No configuration option you choose is permanent. You can change things at any time, with no loss to the metadata you've already captured or entered. You will never configure yourself "into a corner" with Cumulus. No configuration choice is a "mistake," because revisions are always easy.

Configuration changes anytime Cumulus configurations can be updated even while your users remain connected and working. This is one of the most powerful and unique features of the Cumulus architecture. Add a new metadata field, change the order of metadata fields on layouts, create new layouts and more—all without the need to disconnect users or restart your Cumulus Server. (Note: Permissions changes do not take effect until the user's next login. In an emergency, the Cumulus administrator can disconnect any user.)

Fine-tune as you go Launch your Cumulus system immediately, and scale and customize it later. This enables you to start reaping the benefits of the system before you've mastered its ultimate potential.

Never stop learning! Study and experiment on your own, learn from the new tutorials and tips Canto publishes regularly, share ideas in the Cumulus user forums, or even take advantage of the professional services and training offered by Canto and its certified partners.

Most DAM systems on the market require considerable up-front planning and development before the systems can go online. In many cases, the costs of configuration can easily exceed the costs of purchase. And the time required for planning, programming and testing delays any ROI that might—or might not—ever come from those unnecessarily complicated systems.

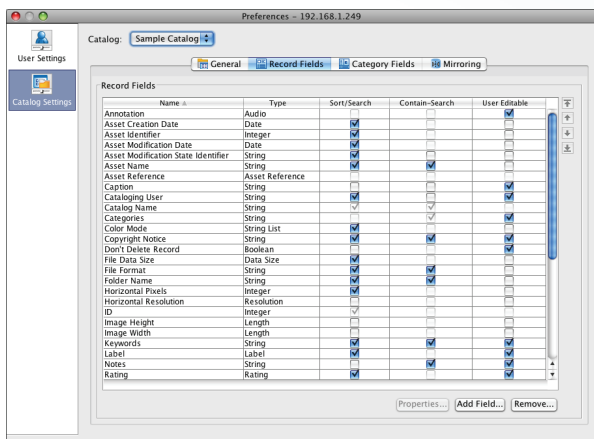
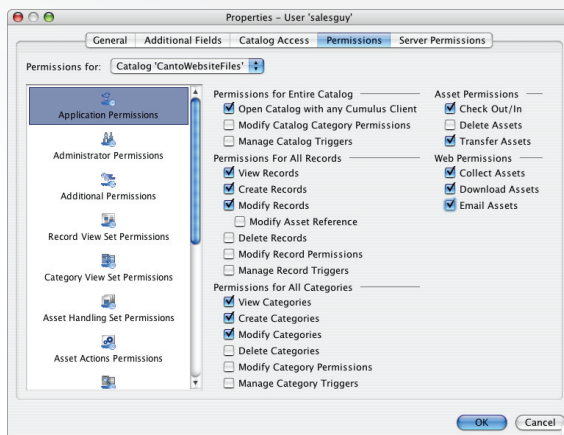


FIGURE 7: Cumulus configuration is done via user-friendly controls that are easy to use, even for new administrators. The image on the left shows the fields in a catalog. Adding a new field, or setting options for existing fields is intuitive. On the right is a window used to set access permissions for a user.

In either case, no command line typing is required—ever!



What's worse, when configuration changes are made, most DAM systems need to be taken offline, sometimes for days or weeks. This not only reduces the likeliness that an organization will ever fine-tune their system exactly how they want it, it also requires that changes be made only during the times the organization can afford to take their system offline—a time that rarely ever comes.

Some systems do promise out-of-the-box readiness, but none of those systems come close to Cumulus in terms of overall power or potential.

User Interface Flexibility & Power

The Cumulus user interface was designed to provide ready access to the features users need most, while making it easy for Cumulus adminis-

trators to hide features and metadata fields users don't need, or shouldn't see.

In addition, the concept of "process" was heavily weighed during UI design. For example, what a user sees at any given point in time is not necessarily based solely on what that user is permitted to see; it can be based on what that user needs to see at that moment.

The Cumulus "view set" is an example of this concept. A view set is a selection of metadata fields that are organized into a layout. The fields chosen for each view set are up to the Cumulus admin, and there is no limit on the number of view sets that can be used, so the potential is limitless.

Figure 8 shows two view set examples. Though this catalog might contain hundreds of metadata fields, view sets like these make it easy to provide users with small subsets that are task-specific and easy to use.

Access to view sets is controlled via user or group permissions, so it's easy, for example, to hide administrative fields from designers, or make sure the fields managers use for comments are visible only to other managers.

Figure 3 (back on page 3) shows an example of a view set used to set and edit metadata values associated with asset or project development.

Configuring the fields that appear on view sets is easy, as shown in figure 9.

Organization that Makes Sense

Cumulus categories take the best aspects of keywords and folder structures and combine them into a unique organization tool far more powerful than keywords or folders alone.

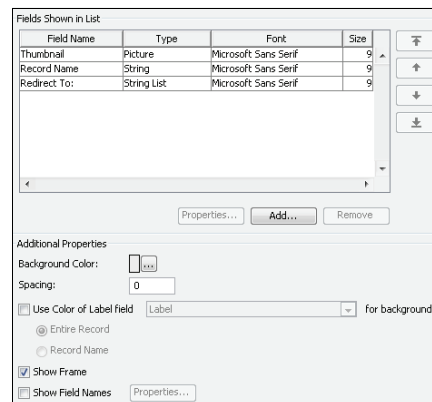


FIGURE 9: Adding fields to view sets is easy, and all changes made apply to the Mac, Windows and Web clients. Options exist for fine-tuning the appearance of individual fields and the layout.

Categories can be nested to create hierarchies, and assets can be assigned to any number of categories at a time. (Figure 10)

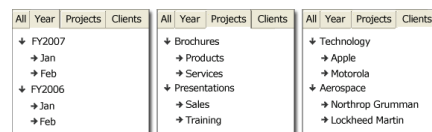


FIGURE 10: Different category trees are useful for organizing assets using different taxonomies. Here, structures are used to organize assets based on the fiscal year in which they were acquired (left), the projects in which they've been used (center), and the clients to whom they belong (right).

Double-clicking on a category finds all assets assigned to it, and category assignments (and category structure changes) are made via simple drag & drop.

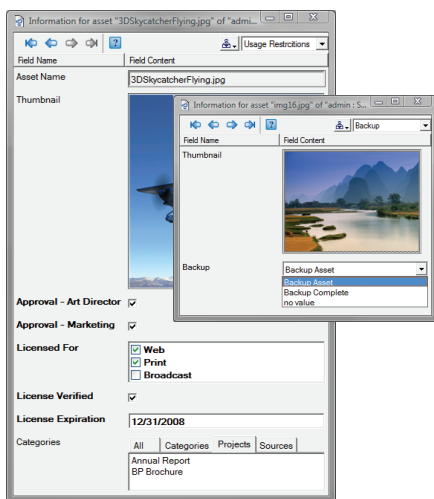


FIGURE 8: Here, a view set called "Backup" provides access to a thumbnail and a single field used to indicate when an asset should be backed up. The "Usage Restrictions" view set provides single-source access to all catalog fields that affect the legal use of an asset.

Access to categories or entire category trees can be controlled via user or group permissions.

The biggest benefits Cumulus categories offer over folder structures is that a single asset can belong to any number of categories, and it makes no difference where the asset is stored.

And, while category assignments are as easy to make as keywords or tags, the similarities end there. The hierarchical structure of categories makes more meaningful assignments easy, and categories can be renamed or moved in the hierarchy at any time without any worries—all associations are automatically updated throughout the entire catalog.

Why does this matter?

Say you have a million assets that include the text tag “U.S. President George W. Bush” Then, all of a sudden, those tags should read, “Former U.S. President George W. Bush.” Using a tag-based system, you suddenly have a million new things to do! But using Cumulus, it’s a single edit that automatically propagates to all affected assets.

Or, say you’re organization has cataloged a few hundred thousand PowerPoint slides that you’ve assigned to a category structure like:

Current Projects > Sales> Presentations

Once that project is no longer current, Cumulus would let you drag that Presentations category anywhere else in your category hierarchy you needed. So, it might end up under:

Archived Projects > Presentations

You can rename or move categories as often as you need. Cumulus always makes sure everything is updated for you.

Automation Options

The automation built into Cumulus enables the program to handle some of the repetitive tasks associated with collaborative development, including:

- Sending notifications of status changes, additions or deletions.
- Applying standardized metadata values to asset records (copyrights, initialization values, etc.)
- Modifying metadata values based on other metadata values and conditions.
- Setting and changing permissions for asset access based on default values, or variable metadata values, like embargo dates, status updates, etc.
- Creating variants of cataloged assets, such as format or colorspace conversions and Web-

ready downsampling, based on pre-defined configurations.

- Creating entirely new assets, such as making a new PowerPoint presentation from cataloged slides or images.

Time savings is the most obvious benefit of automation—and saved time is saved money!—but another important benefit is accuracy and protection. For example, having someone scour your catalogs daily for expired asset licenses is a good idea, but it’s a bad use of resources. Having Cumulus do it automatically makes far more sense. Not only do you free up your personnel for other tasks, but Cumulus is far less prone to errors when it comes to searching and making metadata changes—and it’s happy to work off hours and weekends!

Cumulus initiates all automated tasks based either on time intervals or user actions. When you need something to occur at repeating intervals, the Cumulus Scheduler makes it easy to define tasks and determine when they should be performed. (Figure 11)

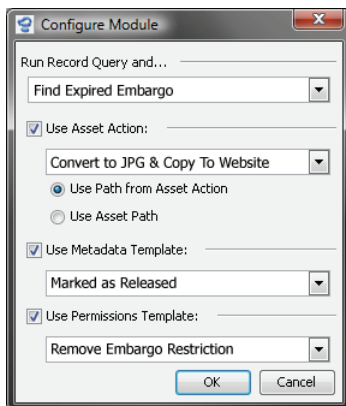


FIGURE 11: Using queries, actions and templates previously defined, this Cumulus Scheduler task searches for assets whose embargoes have expired. It creates a JPG copy of those assets and saves the file in a staging area for the website. It then updates the status of the asset to “Released” and it removes a permission restriction so that the asset can be viewed by others.

As simple in concept as the action shown in figure 11 is, the steps taken in the background by Cumulus are significant. Imagine having to do the following manually:

1. Create a query to find assets whose embargo dates have expired.
2. Open the files in an image editing application and resize and save them in JPG format, suitable for the Web.
3. Copy the files to the staging area where your Web developers can find them.

4. Update the asset record’s status to indicate it’s been released.
5. Update the asset’s permissions so that others can see it.

These five steps would be required for every asset that was ready for release! Now, imagine doing that everyday. A Cumulus Scheduler action can do it for you, at intervals you define, which frees your most valuable and costly asset—you’re employees—to do other things. (Figure 12)

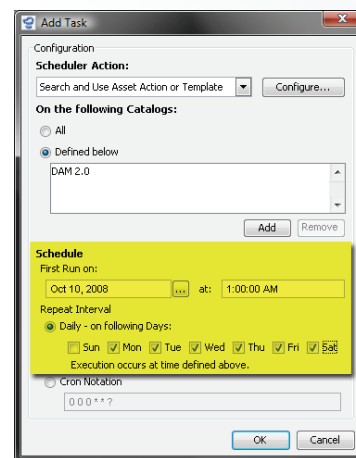


FIGURE 12: This task is set to first run on 10/10 at 1AM. it will run again at 1AM each day of the week, except Sunday. Tasks can be configure to run on all catalogs, or just certain catalogs, as shown here. Cron notation offers even more control over time intervals.

You can configure as many Scheduler tasks as you need, and they can be run at any time. Cron notation enables you to define more detailed time intervals, such as “every 5 minutes” or “the first Sunday of each month.”

In addition to scheduled tasks, Cumulus can initiate automated tasks based on user actions. Figure 13 shows how easy it is to configure an automatic email notification based on a metadata field value.

Notifications can also be sent based on other user actions, like cataloging new assets, creating new categories, changing category assignments and more.

In addition, automation tasks can be “wrapped” into easy-to-use menu options. Figure 14 shows how a selection of actions can be used in a drop-down menu in a Cumulus Web client to make it easy for users to perform otherwise complex tasks.

What’s important to note about what you see in figure 14 is the power going on behind the scene. For example, when the *Create New PowerPoint* menu item is selected, Cumulus actually creates

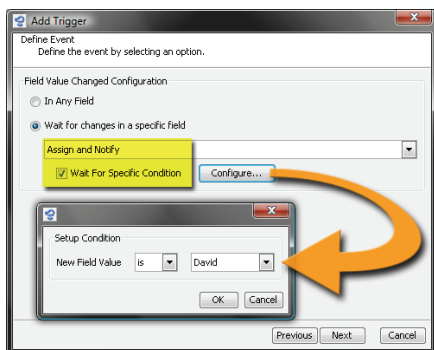


FIGURE 13: When someone sets the *Assign and Notify* field to *David*, Cumulus will send email to David letting him know.

a brand new PowerPoint presentation from the selected assets—even if PowerPoint isn’t available on the user’s computer.

The “wrapper” used in Cumulus to make complex processing like this easy is called an *asset action*. Each menu item shown in figure 14 is an individual asset action. You can have as many as you need, and you can control with pinpoint precision which users or groups can access each.

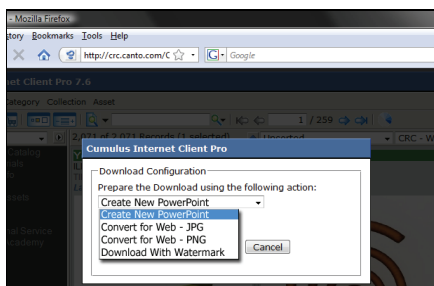


FIGURE 14: What appears to users as standard menu options are actually automation actions Cumulus performs when the menu item is selected. All options are available via the Cumulus native clients and the Cumulus Web clients, as shown here.

Formulas can also be used to determine new or revised metadata values based on other values. As an example, every image asset has a horizontal and vertical dimension, which enables designers to quickly determine an asset’s suitability for print or the Web—an image that’s 10,000 pixels wide is *not* suitable for the Web! But not everyone can tell when a given asset is suitable for emailing, presentation inclusion, etc. A formula can easily set a metadata field value to “Web Ready” or “Do Not Email” based on the image’s dimensions, with no user input required.

Asset Security & Integrity Assurance

A 2008 survey of Canto customers found that an overwhelming number favored Cumulus for the

security features it offers. In short, Cumulus permits pinpoint accurate access control to virtually all aspects of the system. The table that follows outlines some of the control that’s possible.

Permission	Examples	Benefits
Assets	<ul style="list-style-type: none"> No access Thumbnail preview only Check out/in Full access Download variants only (watermarked, low-res, etc.) 	<ul style="list-style-type: none"> Ensure assets as editable only to certain individuals or groups. Provide “FPO” or watermarked Web access. Make assets searchable, without being accessible.
Catalogs & Metadata	<ul style="list-style-type: none"> Catalog new assets Edit metadata Create new categories Delete assets Delete categories 	<ul style="list-style-type: none"> Control who can add assets to your catalogs. Restrict metadata editing to trusted users. Prevent assets (and metadata) from being deleted.
Admin Tasks	<ul style="list-style-type: none"> Determine who can create new actions, view sets, metadata templates and other resources. Limit actions to certain users or groups. Force the use of actions for Web download. Delegate administrative tasks. 	<ul style="list-style-type: none"> Maintain control by determining who has access to what. Ease workflow for users by limiting their access to features they won’t need. “Deputize” select users to take on administrative functions.

Asset access permissions can be set catalog wide, or on a per-asset basis. Templates and automation can be used to make permissions assignments easy, even in the most complex systems.

Access to metadata fields is also controlled via permissions, and it’s easy to restrict access to certain fields to certain users or groups. Figure 15 shows how an editing restriction is made. Because this field will be editable only by the CEO, it can act as a virtual “digital signature” for approvals.

As an added layer of protection for your assets themselves, the Cumulus Vault serves as a virtual wall between your files and your operating system. The Vault isolates your files from the file system, and the potential dangers it presents, such as inadvertent file overwrites and viruses.

Access to the files is controlled via Cumulus, but not in any way that poses risk to your files—even in the worst case scenario, the protection Cumulus offers will never damage your files or modify their contents.

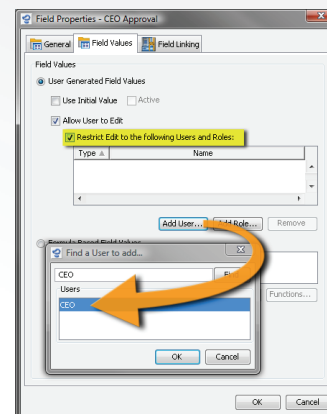


FIGURE 15: It’s easy to restrict the editing of a field to certain users or groups. Here, editing of a field called *CEO Approval* is restricted to the CEO.

Viruses that target files based on file types are rendered impotent, because they will not recognize the files inside the Cumulus Vault.

Check in/out and version control services are available to *all file types* stored inside the Cumulus Vault. You can store as many revisions as you need, and roll back to earlier versions at any time. You can also restrict access to versions to make sure, for example, that your sales force always see only the most recent versions. (Figure 16)

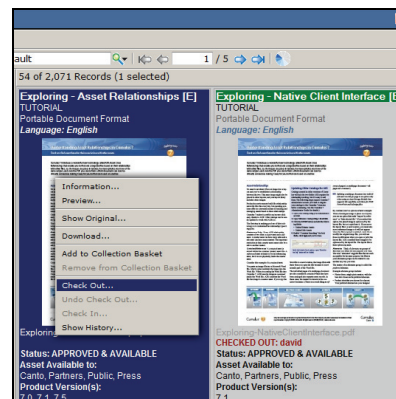


FIGURE 16: Asset check in/out is also available via the Cumulus Web clients. Notice the indication on the asset record to the right that it is already checked out.

Easy Access to Everything Now

Cumulus native client software is available for Windows and Mac OS. User interface standards like menus, buttons, drag & drop, etc. are supported.

The Cumulus Web clients closely resemble the native clients, which makes them powerful alternatives when Web access is required. “Web 2.0”



technologies enable advanced functionality in the Web clients, such as drag & drop category assignments, context menus and more.

Combined, the Cumulus client options make it possible for anyone to access your entire cataloged asset library from anywhere on Earth. The same user accounts, permissions settings, server resources are used throughout all clients, so you never need to do any additional configuration just for Web users.

Cumulus catalogs published over the Web are always “live,” meaning your Web users see exactly the same assets and metadata values your LAN-based users see. When new assets are cataloged in any client, they are accessible (permissions permitting) on both sides of the Internet connection.

Total Access to Everything Tomorrow

All Cumulus Servers are expandable via the addition of user licenses and add-on products. No upgrade requires a loss in productivity due to prolonged downtime. In addition, systems can be enhanced through customizations performed by your IT staff, Canto Professional Services, or the services of any member of the worldwide Canto Certified Partner network.

Access to the underbelly of Cumulus is available via a published API, making even the most advanced enhancements easy. Java technology makes cross platform client and server enhancements possible, and Web Services support extends that power further. Software developer kits for Java and C++ ensure your development team can extend the power of Cumulus using the environments they know best.

Thanks to the vast network of Canto Certified Partners and the products and services they offer, and the well documented APIs that provide total

access to the heart of Cumulus, your system will never become obsolete.

Summary

As money gets tighter for everyone, more and more organizations worldwide are realizing their investments in digital assets are significant and, without proper attention, are easily lost. For this reason (and many others), a digital asset management system like Canto Cumulus makes more sense now than ever.

This document has shown that no matter how complex (or beyond hope!) you might think your current situation is, Cumulus can help you gain the control you need almost immediately. And, unlike most investments, your ROI with Cumulus is entirely within your control—the *sooner* you use it, the quicker you’ll recoup your purchase costs; and the *more* you use it, the safer your asset investment becomes.

You’ll find what you need, using a variety of search options—each perfect for a different situation. You’ll be able to track asset and project development from any Internet connection on Earth, and yours sales or other remote teams will never miss an opportunity because a needed file was on another continent.

Your financial, creative, sales, marketing, distribution and other departments will all share access to your organizational assets, and no one will see or access what’s not meant for them. Even better, nobody will re-license what’s already been paid for, or use something they shouldn’t, just because they didn’t know any better.

You’ll never lose information due to an erroneous error or damaged files, because all previous versions are always safe.

There will be no more getting buried in landslides of duplicate assets, trying to determine which one is current! And no more using the wrong version of a file, because approvals weren’t clear.

You’ll always have the support you need, because no one has a more expansive worldwide network of certified partners who specialize in digital asset management—and no company has been exclusively selling digital asset management systems for longer than Canto.

When it comes right down to it, saving time and money is what digital asset management is all about. With Cumulus, you’ve purchased the most affordable professional DAM system in the world today, and the benefits it offers—many of which have been outlined in this document—will increase that value even further.

For more information on how Canto Cumulus can benefit your organization, contact your local Canto partner, or contact Canto directly.

In the U.S.
+1 (415) 495-6545

In the E.U.
+49 (0) 30 390 485 0

Everywhere
info@canto.com

About Canto & Cumulus

Canto has been dedicated to helping customers fully utilize their digital assets since 1990. Canto Cumulus is a cross-platform solution that enables companies to easily organize, find, share and track their ever-increasing numbers of digital files, in any format. Canto’s worldwide network of certified developers offers an impressive assortment of plug-ins that enhance the Cumulus product line further.

Learn more: www.canto.com