

Metadata in a multimedia asset management system:



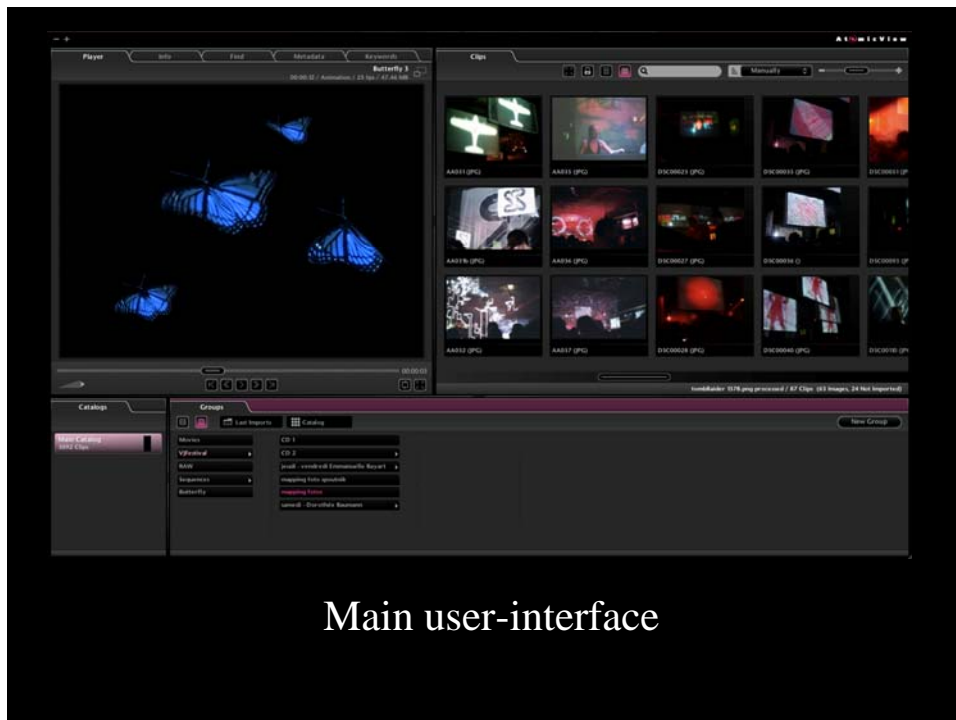
AtomicView

AtomicView

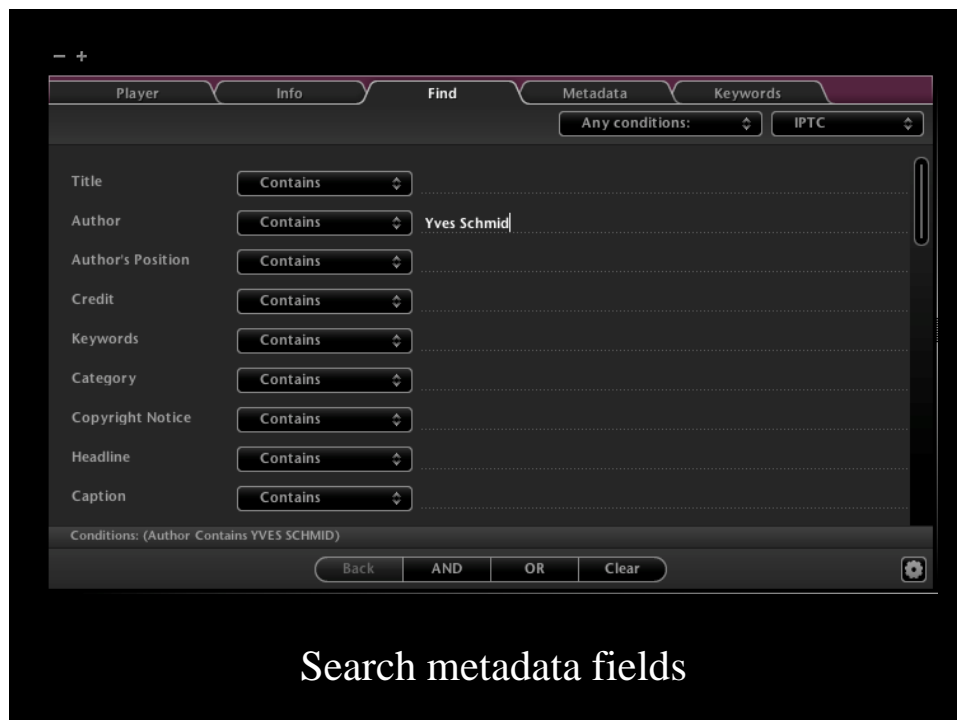
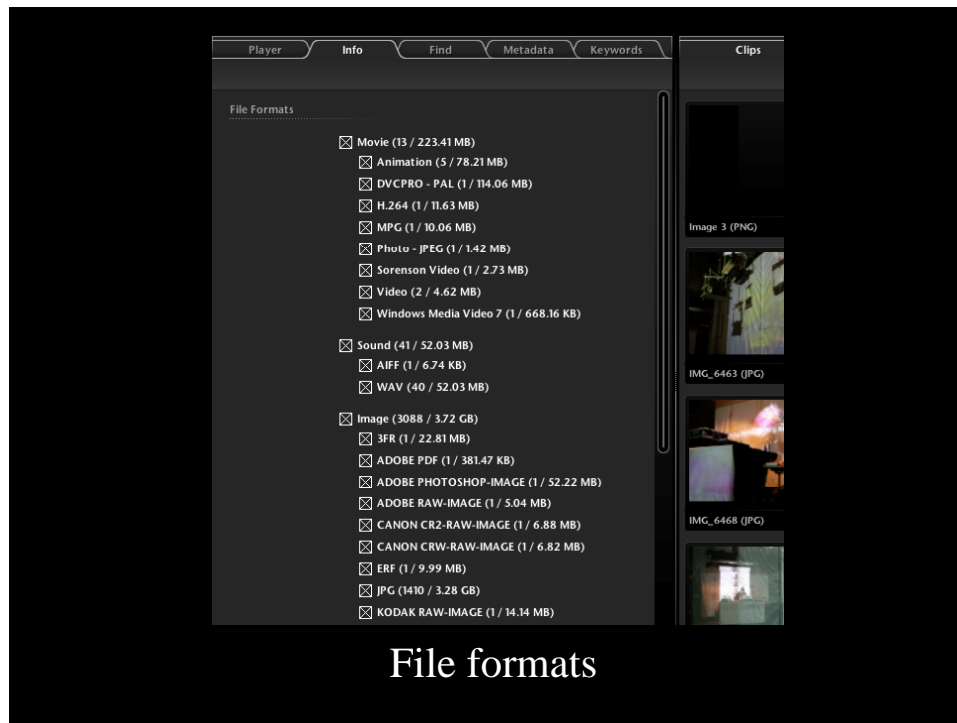
Organize, search, visualize, manage
multimedia assets:
Videos, pictures, sounds, documents...

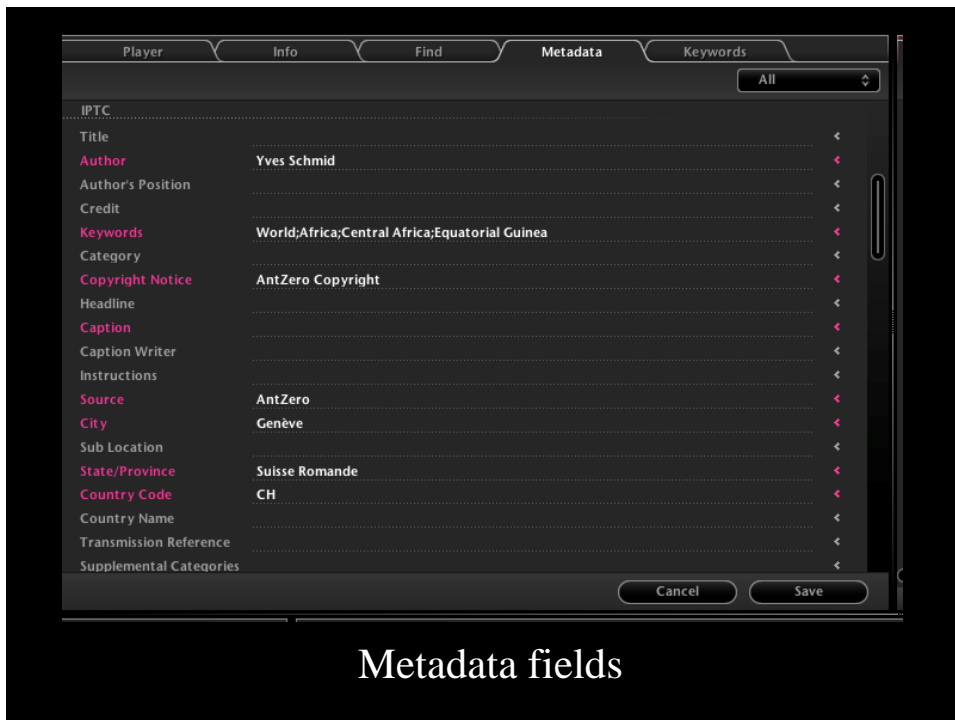
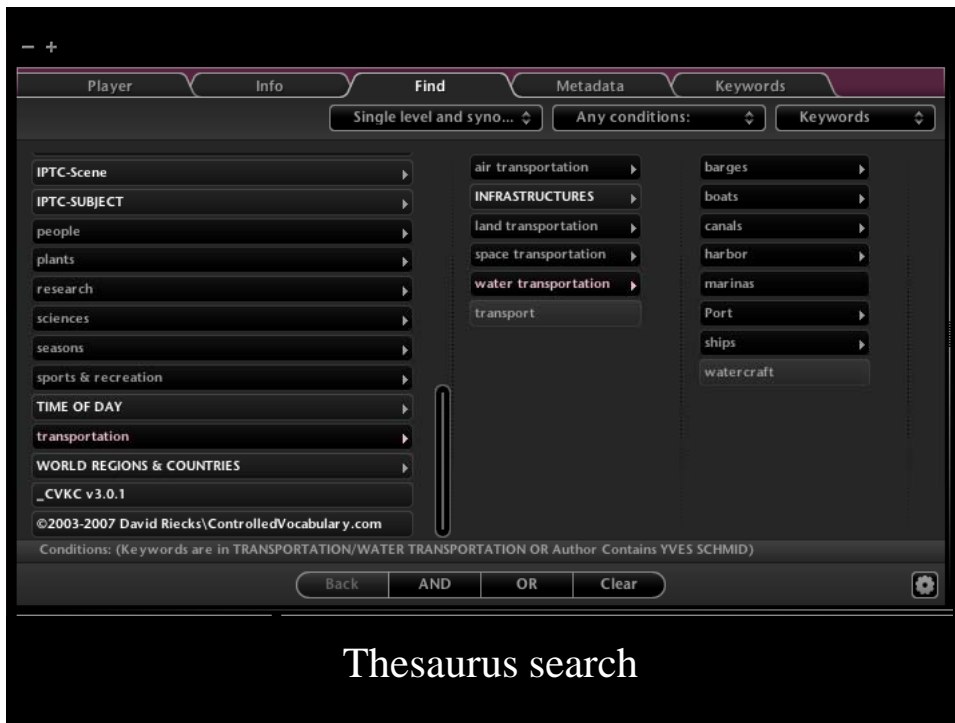
AtomicView Server

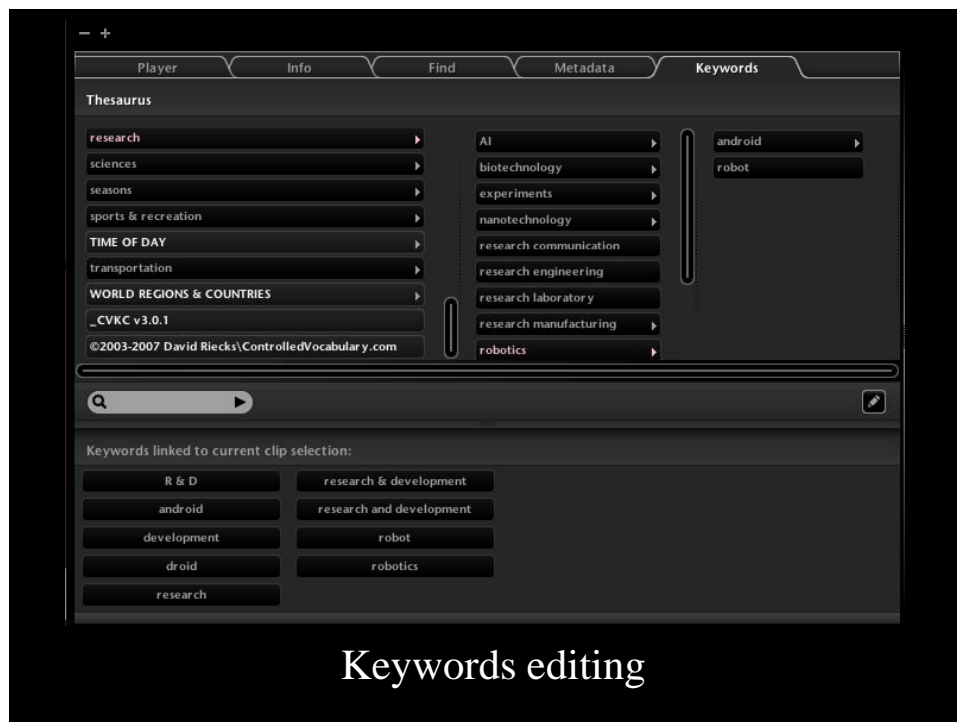
Big centralized media archives
Collaborative work



Main user-interface







Internal database :

- Contains all metadata fields

Internal database :

- Contains all metadata fields
- Support for user defined fields

Internal database :

- Contains all metadata fields
- Support for user defined fields
- Fast search using index

Internal database :

- Contains all metadata fields
- Support for user defined fields
- Fast search using index
- Modify files only when required

Copying metadata to files

- Private fields vs standard fields

Copying metadata to files

- Private fields vs standard fields
- What standard fields can we use in a multimedia context ?

Standard metadata schema

- IPTC : makes sense for all formats

Standard metadata schema

- IPTC : makes sense for all formats
- DublinCore : makes sense for all formats

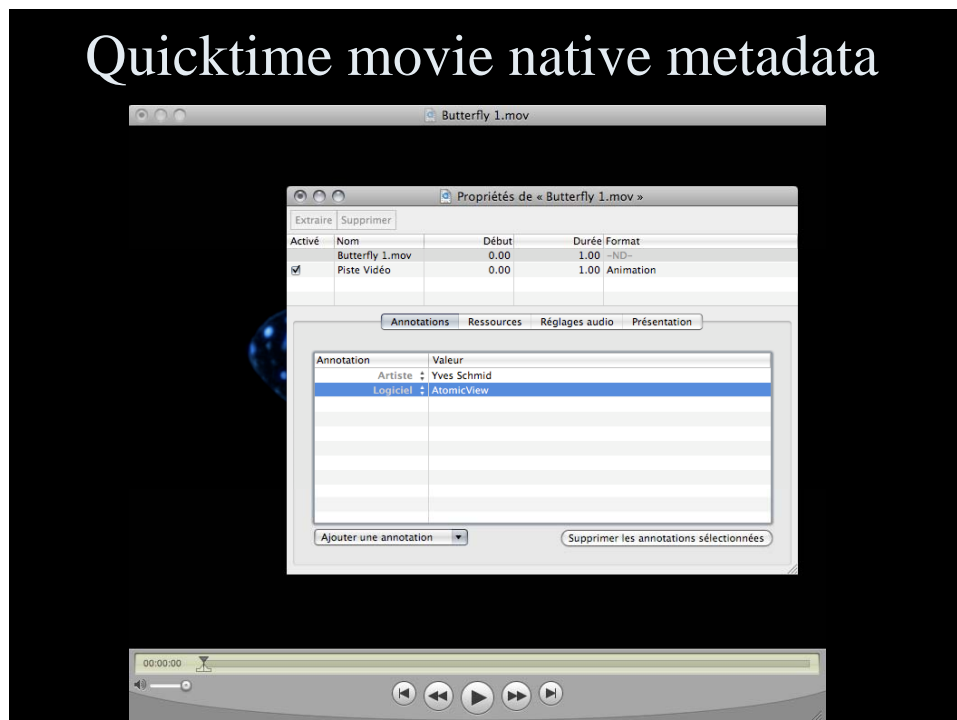
Standard metadata schema

- IPTC : makes sense for all formats
- DublinCore : makes sense for all formats
- Exif : mostly used in digital photography

Format limitations :

- Some formats use their own native metadata schema.
- Some formats do not even support metadata at all.

Quicktime movie native metadata



Strategies :

- 1) Use one metadata schema per file format

Strategies :

- 1) Use one metadata schema per file format
 - Very confusing
 - Difficult to manage in a centralized database

Strategies :

2) Use IPTC fields and remap them to native format fields.

Strategies :

2) Use IPTC fields and remap them to native format fields.

- Not all fields can be written
- Some field remapping is required
- Inconsistent results

Strategies :

3) Use XMP by default and try to remap fields to native formats when possible.

Strategies :

3) Use XMP by default and try to remap fields to native formats when possible.

-AtomicView strategy

XMP

- Support for standard schemas such as IPTC

XMP

- Support for standard schemas such as IPTC
- Every fields are always written

XMP

- Support for standard schemas such as IPTC
- Every fields are always written
- Can be written to many file formats

XMP

- Support for standard schemas such as IPTC
- Every fields are always written
- Can be written to many file formats
- Can be written to a sidecar file

XMP

- Support for standard schemas such as IPTC
- Every fields are always written
- Can be written to many file formats
- Can be written to a sidecar file
- Easy to integrate using Adobe SDK

XMP

- Support for standard schemas such as IPTC
- Every fields are always written
- Can be written to many file formats
- Can be written to a sidecar file
- Easy to integrate using Adobe SDK
- Cross-platform

XMP

- Support for standard schemas such as IPTC
- Every fields are always written
- Can be written to many file formats
- Can be written to a sidecar file
- Easy to integrate using Adobe SDK
- Cross-platform
- Can be extended with custom schemas

IPTC implementation in XMP

- IPTC unique fields are stored in the IPTC Core schema
- Other fields are mapped to various schemas in the XMP

AtomicView metadata schema representation

- Select what representation you want: IPTC, DublinCore, Photoshop, etc.

AtomicView metadata schema representation

- Select what representation you want: IPTC, DublinCore, Photoshop, etc.
- Automatically binds XMP merged fields

IPTC

		IPTC
Title	<
Author	Yves Schmid	<
Author's Position	<
Credit	<
Keywords	<
Category	<
Copyright Notice	<
Headline	<
Caption	<
Caption Writer	<

DublinCore

		Dublin Core
Contributor	<
Coverage	<
Creator	Yves Schmid	<
Date	<
Description	<
Format	<
Identifier	<
Language	<
Publisher	<
Relation	<
Rights	<

Keywords

- Very useful for all media formats

Keywords

- Very useful for all media formats
- Works the same way for all media formats

Keywords

- Very useful for all media formats
- Works the same way for all media formats
- Are stored in the IPTC « Keywords » field as a simple word list.

Keywords

The screenshot shows a software interface with a 'Keywords' tab. The interface is divided into several sections:

- Thesaurus:** A list of categories with expandable arrows, including 'plants', 'research', 'sciences', 'seasons', 'sports & recreation', 'TIME OF DAY', 'transportation', 'WORLD REGIONS & COUNTRIES', and '_CVKC v3.0.1'. Below this is a copyright notice: '©2003-2007 David Bieck's ControlledVocabulary.com'.
- Keyword List:** A vertical list of keywords with expandable arrows, including 'AI', 'biotechnology', 'experiments', 'nanotechnology', 'research communication', 'research engineering', 'research laboratory', 'research manufacturing', 'robotics', and 'development'.
- Search:** A search bar with a magnifying glass icon and a right-pointing arrow.
- Keywords linked to current clip selection:** A section containing a grid of buttons for selected keywords: 'R & D', 'research & development', 'android', 'research and development', 'development', 'robot', 'droid', 'robotics', and 'research'.
- Clips:** A sidebar on the right showing a list of clips, including 'Image 3 (PNG)' and 'IMC_6463 (JPG)' with a small thumbnail image.

With this solution :

- You can use the representation you want for editing and searching the metadata fields

With this solution :

- You can use the representation you want for editing and searching the metadata fields
- All fields are always written and properly mapped to the different schemas

Main limitation :

For applications that do not support XMP only some or none of the metadata fields are readable.

AtomicView Server custom metadata schemas

-Working with metadata over a timeline : keywords, notes, rights, ...

AtomicView Server custom metadata schemas

- Working with metadata over a timeline : keywords, notes, rights, ...
- Versioning: manage multiple versions of a file in the database.

AtomicView Server custom metadata schemas

- Working with metadata over a timeline : keywords, notes, rights, ...
- Versioning: manage multiple versions of a file in the database.
- Collaborative work: task list with team members, work progress on a specific media file, etc.

Current implementation

NOW:

Stored only in the AtomicView internal database

Current implementation

NOW:

Stored only in the AtomicView internal database

IN THE FUTURE:

- Could be added as new XMP schemas
- User-interface could be implemented as XMP plug-ins allowing fields to be read/modified from editing softwares.

Thank you!

Yves Schmid Dornbierer / Software Architect
AntZero / www.antzero.com / antinfo@antzero.com