Distributed Metadata Management for Post-production Environments

Werner Bailer[1], Konstantin Schinas[2], Georg Thallinger[1], Wolfgang Schmidt[2], Werner Haas[1]

[1] JOANNEUM RESEARCH Forschungsgesellschaft mbH
Institute of Information Systems & Information Management
Steyrergasse 17, 8010 Graz, Austria

[2] DVS Digital Video Systems GmbH
Krepenstraße 8, 30165 Hannover, Germany

Abstract: Efficient and flexible management of digital essence and associated metadata is of critical relevance in post-production. We describe a distributed content management system, which uses a peer-to-peer architecture in order to support the dynamic nature of post-production setups. Each peer indexes content on storage under its control by extracting relevant metadata from the headers of essence files as well as by performing automatic content analysis in a background process. The extracted metadata are indexed in a lightweight database at each peer and stored on disk in MPEG-7 XML format in order to provide a standard compliant interface for metadata exchange. The client tool running on a peer allows to edit the metadata in the data base, in the MPEG-7 file and in the file headers (e.g. to adjust timecodes). Moreover, the software provides tools for essence management (copying, moving, defragmentation). The system allows searching for content across all peers in the network. A visual keyframe-based browsing interface allows exploring the indexed content based on the extracted metadata.

1 Introduction

Digital media technology resulted in a lasting change in post-production workflows. In digital post-production physical “storage media” such as film rolls and video tapes recede in importance. Instead, uncompressed image sequences in high resolution (e.g. in 2K or 4K) can be found as huge amounts of data stored on various storage solutions, and it is not uncommon that a blockbuster movie requires about 100 terabytes of storage. During post-production a lot of different files exist, mostly distributed on several storages. In data management in such environments is a big issue when relying only on standard “tools” like notes, filing cards, labels, and hierarchical file structures with in-house directory naming conventions.

Metadata can most simply be defined as "information about content". Images and sounds are the content in the film business. Metadata in the digital production workflow means the information about the images and sounds, their composition, their sources, their owner rights and what else is needed for a detailed description of a fully digital film project.

In the old days of film production all images were captured on celluloid film, the editing tool was not much more than a pair of scissors, and special effects were applied by optical or chemical methods. The metadata, the knowledge about the existing film material, the editing decisions and intended special effects, existed either in the brains of the people, working on the film or as written notes. Whether it's Post-it notes stuck on film canisters, a cinematographer's diary or scribbling in the margins of a script, filmmakers have always found ways to convey information about the artistic intent of the captured images and sounds.

The adoption of digital processes and the advent of a hybrid analog-digital workflow, called the digital intermediate process, have introduced digital storage of metadata accompanying the images and sounds processed in the digital domain, as the digital process depends on the interoperability between adjacent nodes in the system. However, most of this metadata consist just of the mandatory technical parameters for describing the digital format of the image and sound essence.

While in the beginning just selected scenes were processed digitally, today whole films are going through the digital intermediate process. Over time the filmmakers identified the need for metadata that reliably conveys – and preserves – their creative intent, metadata that keeps track of what happens to each image at each stage of the process, that makes it possible to track changes, and finally ends up with a film that looks the way the cinematographer meant it to look. The implementation of the vision of digital cinema still extends the variety of necessary metadata. With the latest digital film cameras and digital projectors that find acceptance by at least a reasonable part of the cinematographers all components for a fully digital film process are now available.

An efficient file management is possible with metadata, referring to the stored content, and an intelligent data management software. However, most of the available software solutions are not flexible enough to account for the different workflows in the film making industry. In this paper we describe an alternative approach with a distributed content management system, which uses a peer-to-peer architecture in order to support the dynamic nature of post-production setups.

It is fair to state that the current post-production process is based on files. But in the film making business this is mostly the case for the essence only – i.e. audio and video data. During the content creation the creative staff has to handle a serious number of files nowadays. However, browsing through big amounts of files and searching for the required ones should not be the major tasks of the staff. Unfortunately, in a standard network environment with several storages exactly these tasks are very time-consuming because the standard search functions of current operating systems are not optimized for a post-production workflow. Moreover, metadata mainly exist on paper or whiteboards only.

In order to overcome such problems various different Content Management Systems (CMS) are available. But such solutions are often developed for the special demands of the broadcast industry and not for movie post-productions. Most CMS are intended to be long-term installations with a central database and a central storage, and the manufacturers of those systems rely on the assumption that workflows remain static and the maximum number of clients in the network is known.

However, workflows in the post-production process of a movie differ from the workflows of broadcasters. Often several departments use “their own” storage solutions. Workflows, and with them the number of workstations and storages in use, may even differ between the different creative phases. Thus, a very flexible CMS is required.

In post-production environments, users typically deal with large amounts of audiovisual material, such as newly shot scenes, archive material and computer generated sequences. A large portion of the material is unedited and often very redundant, e.g. containing several takes of the same scene shot by a number of different cameras. Typically only few metadata annotations are available (e.g. which production, which camera, which date). The goal is to support the user in navigating and organizing these material collections, so that unusable material can be discarded, yielding a reduced set of material from one scene or location available for selection in the post-production steps.

Viewing complete multimedia items in order to locate relevant segments is undesirable even for relatively small content sets due to the required user time for viewing and the amount of data that need to be transferred. Classical search and retrieval approaches require sufficient metadata to index the content and the feasibility to formulate a query in terms of the available metadata. Manual annotation of the content yields semantically meaningful metadata that can be effectively searched by human users, but at high annotation costs. Automatic metadata extraction approaches are in many cases not able to fully capture the semantics of the content. This makes it difficult to query the right content.

Multimedia content abstraction methods such as visual browsing are complementary to search and retrieval approaches, as they allow for exploration of an unknown content set, without the requirement to specify a query in advance. In order to enable the user to deal with large content sets, it has to be presented in a form that facilitates its comprehension and allows to quickly judge the relevance of segments of the content set. Media content abstraction methods shall (i) support the user in quickly gaining an overview of a known or unknown content set, (ii) organize content by similarity in terms of any feature or group of features, and (iii) select representative content for subsets of the content set that can be used for visualization.

This paper is organized as follows: In Section 2 we present a novel approach for distributed content management in post-production. Section 3 discusses the metadata model used in the system. In Section 4 we present the services and functions provided by the back-end of the system, while Section 5 discusses the functionality available in the client applications. Section 6 concludes the paper.

2 Content Management for Post-production

Users want to know at any time where an individual clip is located and they want to have easy access to its metadata. This makes efficient data tracking and data retrieval a must. Therefore, all commonly used file formats and their metadata must be supported in a content management system (CMS). In the post-production such a system has to support DPX file sequences with their header data, for example. Furthermore, a CMS should be flexible when dealing with different workflows and should not compel users to use only one specific workflow for the data management.

With these requirements in mind we have developed a software called Spycer®, capable to find clips in several terabytes of data, spread all over various storages. In contrast to common CMS, which provide web-based clients only in order to browse a central database on a server, Spycer® is a software that can be installed on every workstation involved in the post-production process. It extracts metadata automatically as soon as a new file is stored on the local storage and makes this metadata available to the SpycerNet. This forms a peer-to-peer network structure, operating without a central database or server.

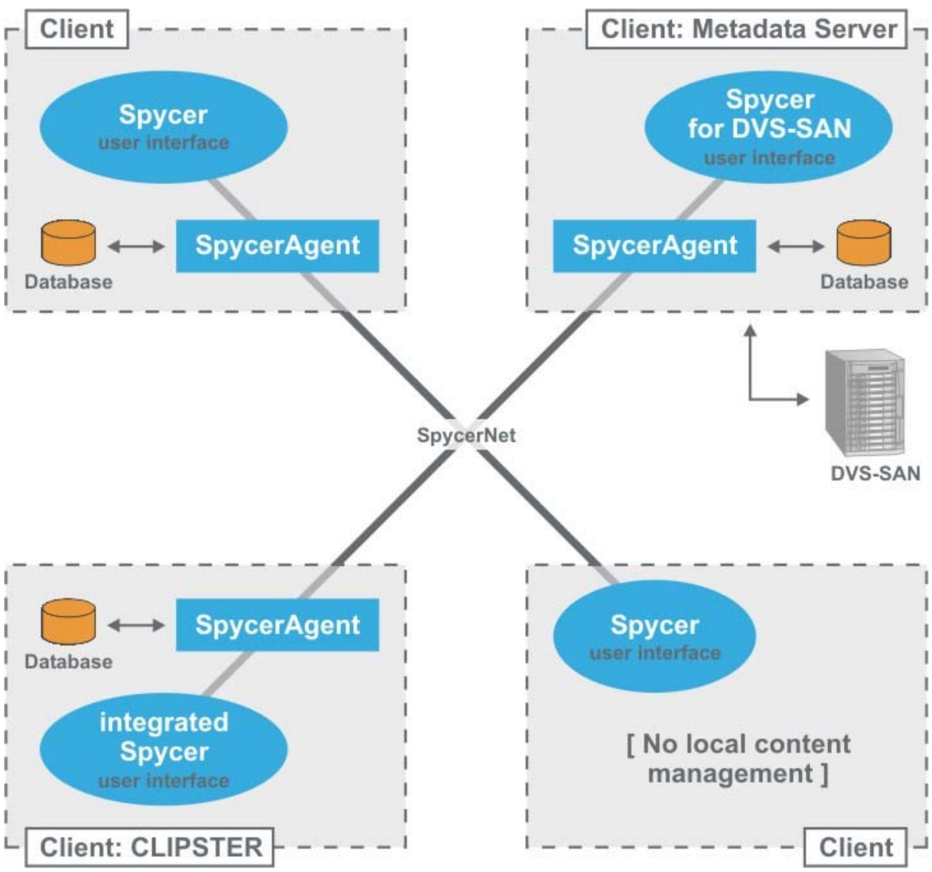


Figure 1: SpycerNet with four Spycer® applications on different client computers. Each client has a Spycer® user interface and the possibility to provide its own file information to the SpycerNet.

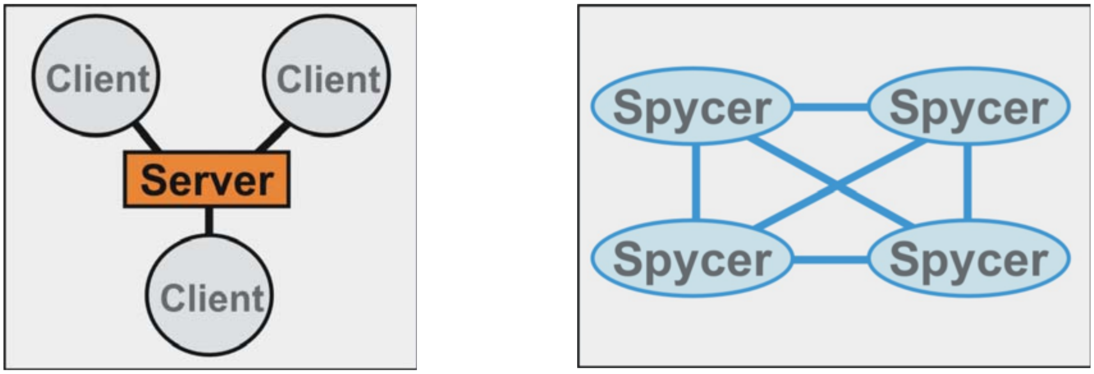


Figure 2: Centralized client-server system approach (left) vs. SpycerNet approach (right), a scalable distributed content management with several virtual connections.

It is hardly ever the case that a single central storage solution is the only storage where all data of a post house are stored and that can be directly accessed from everyone involved in the project. In the majority of cases post-production facilities deal with several islands of storage solutions for security, performance or economic reasons.

In a traditional IT infrastructure a central storage and database are used to administer all information. The problem is that many media files never get recorded in the database. This is especially true if an ordinary client-server structure forms the basis for the content management system (CMS). In such a case the client software provides just a GUI for the database of the CMS. The user has to enter all data manually. For automatic metadata extraction the files have to be stored only on the central storage system where background processes running around the clock can continuously update the database. These types of systems are often limited in scalability and performance due to a single server and database being responsible for all files. Additionally, a more serious problem may occur if the central database suddenly went offline. This single point of failure cannot happen in a pure distributed solution.

What if the database and active CMS applications were distributed on several workstations? Imagine a distributed content management solution where every workstation automatically provides its own database that can be shared with applications of the same kind in a network. Every file gets a record in one of the databases, thus becoming available to all others in the company network. This is how a scalable system should work. According to the rights assigned to each individual user, the user is able to find any media file in the network effortlessly by using common metadata information. In Spycer® we use exactly this approach.

Via the SpycerNet several Spycer® applications communicate with each other for browsing, data retrieval and file handling purposes. Metadata is automatically extracted from file headers or gained by content analysis. This metadata can be used for searching and the retrieval of data in the network. The distributed content management system grows with every additional Spycer® application added to the network. Such a peer-to-peer architecture supports the dynamic nature of post-production setups.

3 Metadata Representation

The metadata representation used in the system determines the capabilities for supporting the diversity of metadata elements and is also crucial for interoperability with the outside world. In the following we collect the requirements to a metadata model for such an application, review relevant standards and formats and the classes of metadata supported by them and discuss their respective strengths and weaknesses.

3.1 Requirements to Metadata Model

Comprehensiveness. The metadata model must be capable of modeling a broad range of multimedia descriptions (e.g. descriptions of different kinds of modalities and descriptions originating from different sources), covering the existing variability in terms of types, properties and sources.

Fine-grained representation. The metadata model must allow describing arbitrary fragments of media items. The scope of a description may vary from whole content to small spatial, temporal or spatiotemporal fragments of the content.

Structured representation. The metadata model must be able to structure descriptions hierarchically with different scopes and descriptions assigned to fragments of different granularity.

Modularity. The metadata model should not introduce interdependencies within the description that do not exist in the content, such as between content analysis results from different modalities (e.g. speech segments and visual shots). The metadata model shall also separate descriptions with different levels of abstraction (e.g. low-level feature descriptions and textual descriptions of content semantics). This is important, as descriptions on higher abstraction levels are usually based on multiple modalities and often use domain-specific prior knowledge, while low-level descriptions are typically specific to one modality only.

Extensibility. It must be possible to extend the metadata model easily to support types of metadata not foreseen at design time or which may be domain-specific or application-specific.

Interoperability. It must be possible to easily import metadata descriptions from other systems or to export them to other systems.

3.2 Standards and Formats

A number of standards and formats for representing metadata in post-production are given. Different organizations have defined them and they cover diverse application areas. Practically all multimedia file formats include technical metadata of the content. This is especially true for container formats that combine essence and associated metadata in one file. The Material Exchange Format (MXF) [MXF] is a prominent example for such a container format (discussed in more detail below). The Digital Picture Exchange (DPX) [DPX] format is commonly used for still image sequences in digital cinema production and also allows to store a vast range of metadata in its header, supporting the same metadata format as MXF. Another example for a container format is the Digital Cinema Package (DCP) [DCI], used to transport digital movies and associated metadata to cinemas.

The following overview does not discuss all the formats capable of holding some technical metadata but includes only standards that are relevant in audiovisual media production and distribution and that offer sufficient support for descriptive metadata.

MPEG-7. The ISO/IEC standard *Multimedia Content Description Interface* (MPEG-7) [MPEG7] has been defined as a format for the description of multimedia content in a wide range of applications. MPEG-7 defines a set of description tools, called description schemes (DS) and descriptors (D). Descriptors represent single properties of the content description, while description schemes are containers for descriptors and other description schemes. The definition of description schemes and descriptors uses the Description Definition Language (DDL), which is an extension of XML Schema [XSD]. MPEG-7 descriptions can be either represented as XML (textual format, TeM) or in a binary format (BiM).

A core part of MPEG-7 are the *Multimedia Description Schemes (MDS)*, which provide support for the description of media information, creation and production information, content structure, usage of content, semantics, navigation and access, content organization and user interaction. Especially the structuring tools are very flexible and allow the description of content on different levels of granularity. In addition, the *Audio* and *Visual* parts define low- and mid-level descriptors for these modalities.

The concept of profiles has been introduced to define subsets of the comprehensive standard, which target certain application areas. Three profiles have been standardized: the *Simple Metadata Profile (SMP)*, which describes single instances or collections of multimedia content, the *User Description Profile (UDP)*, containing tools for describing personal preferences and usage patterns of users of multimedia content in order to enable automatic discovery, selection, personalization and recommendation of multimedia content, and the *Core Description Profile (CDP)*, which consists of tools for describing general multimedia content such as images, videos, audio and collections thereof. The *Detailed Audiovisual Profile* [BS06] has been proposed to cover the needs of audiovisual media production and archiving. It defines the semantics of the supported MPEG-7 description tools in this application context and, includes in contrast, to the three other profiles described before also the tools for describing low-level audio and visual features of the content.

EBU P_Meta. The European Broadcasting Union (EBU) has defined P_Meta [P_Meta] as a metadata vocabulary for programme exchange in the broadcast industry. It is not intended as an internal representation of a broadcaster's system but as an exchange format for programme-related information in a business-to-business use case. P_Meta consists of a number of attributes (some of them with a controlled list of values), which are organized in sets. The standard covers the following types of metadata: identification, technical metadata, programme description and classification, creation and production information, rights and contract information and publication information.

Material Exchange Format. The Material Exchange Format (MXF) [MXF] is a standard issued by SMPTE, defining the specification of a file format for the wrapping and transport of essence and metadata in a single container. The Material Exchange Format is an open binary file format targeted at the interchange of captured, ingested, finished or “almost finished” audio-visual material with associated data and metadata. Support for technical metadata is built directly into the MXF specification. In order to provide enough flexibility to deal with different kinds of descriptive metadata, a plugin mechanism for descriptive metadata is defined. These descriptive metadata schemes (DMS) can be integrated into MXF files. So far SMPTE has standardized the Descriptive Metadata Scheme 1 (DMS-1) and the EBU has defined a DMS for P_Meta.

DMS-1. The SMPTE Descriptive Metadata Scheme 1 (DMS-1, formerly know as Geneva Scheme) [DMS-1] uses metadata sets defined in the SMPTE Metadata Dictionary (see below). Metadata sets are organized in descriptive metadata (DM) frameworks. DMS-1 defines three DM frameworks that correspond to different granularities of description: production (entire media item), clip (continuous AV essence part) and scene (narratively or dramatically coherent unit). When DMS-1 descriptions are embedded into MXF files they are represented in KLV [KLV] format, but there also exists a serialized format based on XML Schema.

SMPTE Metadata Dictionary. The SMPTE Metadata Dictionary [MDDict] is not a metadata format on its own, but a large thematically structured list of narrowly defined metadata elements, defined by a key, the size of the value and its semantics. It is used for all metadata embedded in MXF files, but the elements defined in the dictionary are also used outside the scope of MXF.

3.3 A Classification of Metadata Elements

There are many criteria that can be used to classify metadata. In the following we try to classify the most commonly used descriptive metadata elements based on the grouping used in the standards described above.

Identification. Identification information contains usually IDs as well as the titles related to the content (working titles, titles used for publishing, etc.). In some formats identification metadata constitutes a separate group of metadata, in MPEG-7 this information is part of the *CreationInformation DS*.

Production. This describes metadata related to the creation of the content, such as location and time of capture as well as the persons and organizations contributing to the production. In MPEG-7 this information is part of the *CreationInformation DS*, in P_Meta production metadata complements identification metadata and the SMPTE Metadata Dictionary contains a group called *administration*, which contains production and rights metadata.

Rights. In the business-to-business oriented P_Meta standard, rights related information is a separate large subgroup, while in MPEG-7 this information is part of the *CreationInformation DS*.

Publication. Publication information describes previous use (e.g. broadcasting) of the content and related information (e.g. contracts, revenues). In P_Meta production metadata complements identification metadata, in MPEG-7 this information is contained in *UsageDescription DS*.

Process-related. This is a separate group in the SMPTE Metadata Dictionary, describing the production and post-production history of the essence, e.g. information about capture, digitization, encoding and editing steps in the workflow.

Content-related. Content-related metadata is descriptive metadata in the narrowest sense. An important part is the description of the structure of the content (e.g. shots, scenes). MPEG-7 provides comprehensive structure description tools for that purpose. In MXF DMS-1 structuring is realized by so called frameworks, related to the complete content, clips and scenes. Content related metadata also includes the textual and semantic description of the content, keywords, segment classification, etc.

Relational/enrichment information. This information describes links between the content and external data sources, such as other multimedia content or related textual sources. In MPEG-7 this information is part of the *CreationInformation DS*, in P_Meta it is ancillary information to identification and the SMPTE Metadata Dictionary contains a *relational* metadata group.

3.4 Discussion

The diversity of the investigated standards is high. None of the standards supports all types of metadata stemming from the production process in a comprehensive way and fulfills all the requirements to a metadata model. Technical and identification metadadata are supported in all standards. P_Meta supports very well metadata necessary for business-to-business exchange of programmes (production, rights and publication metadata). The SMPTE Metadata Dictionary provides support for metadata of the entire production process, but does not contain any structuring capabilities. Only MPEG-7 and DMS-1 support a good description of content structure, with MPEG-7 providing the more powerful and flexible tools for structure description. The MPEG-7 description tools also offer possibilities for linking controlled vocabulary such as classification schemes. Due to its XML Schema based definition, MPEG-7 is extensible and can be jointly used with elements from other XML based standards if necessary.

4 Back-end Services

4.1 Defragmentation of Image Sequences

In a digital intermediate process or digital cinema post-production, clips are stored as uncompressed file sequences. This means that large amounts of files are stored on any storage involved in the workflow. Most storage solutions do not provide efficient defragmentation software for these file sequences. Common defragmentation applications just defragment single files and are “not aware” of a possible relationship between files. For a real-time play-out it is inadequate to have the files individually defragmented only. Instead the whole sequence of files belonging to a clip has to be stored contiguously in large blocks on the storage and not scattered all over it.

File Format	Defrag	Size	Resolution	Color Depth
DPX	99%	1.02 GB	1920 x 1080 Pixel	10
DPX	0%	2.32 GB	1920 x 1080 Pixel	10
DPX	3%	3.56 GB	1920 x 1080 Pixel	10
DPX	100%	395.90 MB	1920 x 1080 Pixel	10
YUV8	100%	197.75 MB	1920 x 1080 Pixel	8
DPX	100%	1.37 GB	1920 x 1080 Pixel	8
DPX	100%	594.14 MB	1920 x 1080 Pixel	8
DPX	100%	594.14 MB	1920 x 1080 Pixel	8
DPX	100%	534.73 MB	1920 x 1080 Pixel	8
DPX	100%	475.08 MB	1920 x 1080 Pixel	10
BMP	100%	183.94 MB	1920 x 1080 Pixel	8
BMP	100%	282.75 MB	2048 x 1556 Pixel	8
DPX	100%	1.16 GB	1920 x 1080 Pixel	10
DPX	100%	1.36 GB	1920 x 1080 Pixel	10
BMP	100%	995.97 MB	1920 x 1080 Pixel	8
DPX	100%	942.24 MB	1920 x 1080 Pixel	10

Figure 3: A data manager must show the current defragmentation status of scattered sequences.

Spycer® is able to defragment any file and to analyze and de-scatter file sequences. A monitoring process updates the current defragmentation status of every sequence, adds this information as additional metadata to the database and keeps the user up-to-date. Figure 3 shows the visualization of the defragmentation status in the user interface. This technical metadata is used to defragment and de-scatter individual file sequences.

4.2 Metadata Extraction

For data handling and tracking, users need intelligent software that monitors every change made in a directory. It is not always possible to use an ideal hierarchical directory structure for a project where everyone can find the required material on large central storage systems easily. Every Spycer® application monitors its own pre-defined "watch folders" and automatically provides extracted metadata to the SpycerNet. Metadata is stored on the local disks of the Spycer® application that extracted it, but is still searchable and – depending on permissions – editable from any client in the SpycerNet.

4.3 Audiovisual Content Analysis and Indexing

Automatic audiovisual content analysis and indexing is a background service that performs extraction of metadata that are contained in the headers of the files of an image sequence and by using automatic content analysis tools. The metadata extraction service is invoked by the Spycer® application that acts as a client. Jobs are submitted by the client into a queue and are processed sequentially. The client can query the progress and status of jobs and the status of the service. The client also has the possibility to pause/resume the service if there are tasks with high priorities and real-time requirements to be carried out on the machine.

The metadata extraction is performed on the clip and a single MPEG-7 file is created. The clip's UUID is used to identify the associated processing job. The metadata extraction service produces the following outputs: An MPEG-7 XML document is created/updated for the clip.

In the following, the features extracted during content analysis are discussed. After automatic content analysis has been performed, the created MPEG-7 documents are indexed for efficient use by the visual browsing tool.

4.3.1 Extracted Features

Shot boundary detection and key frame extraction. First shot boundary detection and representative frame extraction are performed. The results of this step are used as prerequisites for the extraction of other features discussed below and for visualization. As shot boundaries are natural limits of the occurrence of most visual features (such as camera movements, objects), they are an important prerequisite for further visual feature extraction algorithms. For each shot, a number of representative frames are selected. The selection of representative frame positions is based on the visual activity in the material, i.e. the more object and/or camera motion, the shorter the time interval between two representative frame positions.

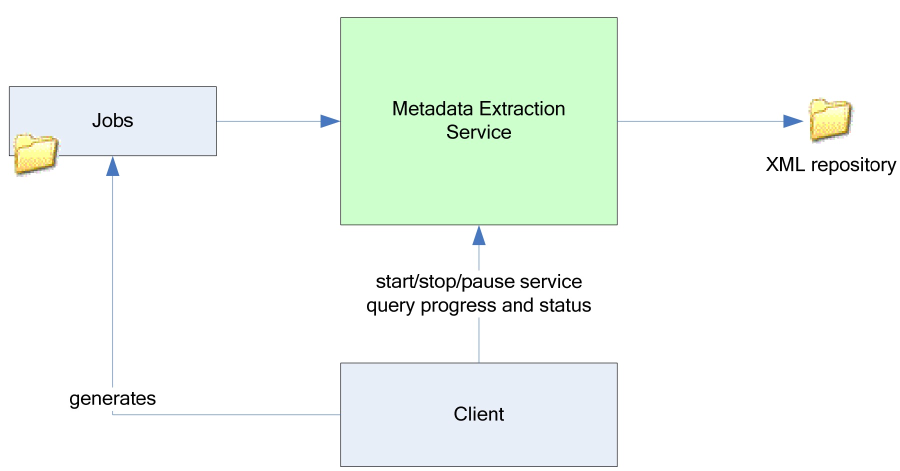


Figure 4: Metadata extraction service.

Camera Motion. The use of camera motion as a browsing feature is twofold: It is often used to guide the user's attention and express relevance of certain parts of the scene, for example, zooming on an object or person is an indicator of relevance, and in field sports, pans indicate the direction of the game. Secondly, it is an important selection criterion during editing, as visual grammar imposes constraints on the camera motion of sequences to be combined. The extraction algorithm (described in detail in [BST05]) is based on feature tracking, which is a compromise between spatially detailed motion description and runtime performance. The feature trajectories are then clustered by similarity in terms of a motion model and the cluster representing the global motion is selected. Camera motion is described on a sub-shot level. A new camera motion segment is created, when there is a significant change of the camera motion pattern (e.g. a pan stops, a zoom starts in addition to a tilt). For each of these segments, the types of motion present and a roughly quantized amount of motion are described.

Visual Activity. Visual activity is a measure of the dynamics in a scene. Together with camera motion information, it is a measure for the local motion in a scene and can thus be used to discriminate quiet scenes from those with object motion. In this application we just measure the amplitude of visual change. The list of amplitude values is then median filtered to be robust against short-term distortions and split into homogeneous segments. Each of these sub-shot segments is described by its average activity value.

Audio Volume. Audio volume can for example be used to discriminate shots without any sound, calm shots of inanimate objects, interviews with a constant volume level and loud outdoor shots in city streets. As no content-based audio segmentation is available in the system, we use segments of a fixed length of 30 seconds. A list of audio volume samples is extracted for each of these segments by calculating the average volume of a 0.5 seconds time window. The list is then median filtered to be robust against short-term distortions and split into homogeneous segments. Each of these sub-segments is described by its average volume value.

Face Occurrence. The occurrence of faces is a salient feature in video content, as it allows inferring the presence of humans in the scene. The size of a face is also a hint for the role of the person, i.e. a large face indicates that this person is in the center of attention. Our extractor is based on the face detection algorithm from OpenCV[1]. In order to make the description more reliable and to eliminate false positives, which mostly occur for a single or a few frames, we only accept face occurrences that are stable over a longer time (we use a time window of about a second to check this). As a result we get a continuous segmentation into sub-shot segments with and without faces. There is no need for a specific clustering algorithm, as there are only two groups of segments (face and non-face).

4.3.2 Indexing

An indexing service is responsible for ingest of content into the abstraction component. The inputs to the indexing service are MPEG-7 descriptions conforming to the MPEG-7 Detailed Audiovisual Profile [BS06], which are produced by the metadata extraction service. In the current implementation, the service watches a directory for new MPEG-7 descriptions. The ingest process could also be triggered by other mechanisms, for example a Web Service call. The indexing service processes the metadata descriptions and fills the index data structures. The core implementation of the service just performs feature-independent tasks such as registering new content, while plug-ins are invoked for all other tasks.

Feature specific indexing is performed by a set of indexing plug-ins. A plug-in extracts the feature related information from the MPEG-7 documents and creates the necessary database entries in the browsing index. A plug-in will also create additional tables in the database if they are not yet there (i.e. if the plug-in for the feature has not been used before). This increases the flexibility of the framework, as new indexing plug-ins can be easily registered with the indexing service and will be used for all further incoming documents.

[1] http://opencvlibrary.sourceforge.net

5 Client Functionality

This section describes the functionalities that are provided by the Spycer® application as well as by the separate visual browsing tool.

5.1 File Handling

Users require dependable copying and renaming processes for such masses of data. High-speed copying mechanisms should be easy to use for the creative staff at their workstations. This means that drag’n’drop procedures must be available instead of inconvenient command line tools. A fast copy tool that avoids fragmentation, is integrated in the application. As shown in Figure 5, users see two browsing panes which show, for example, the source and destination storage of the files for easy drag’n’drop procedures.

With a built-in renaming function for file sequences it is easily possible to change the index number or the whole naming pattern of the sequence (see Figure 8a). Spycer® goes through all files and renames them according to the given pattern automatically.

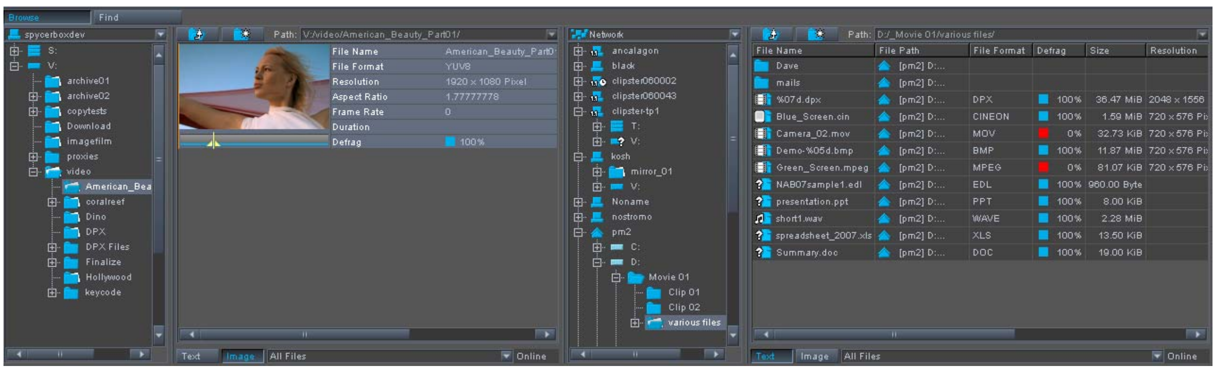


Figure 5: Browsing panes with two directories in parallel enable easy drag'n'drop tasks.

5.2 Metadata Viewing and Editing

The Spycer® application provides tools for visualizing all the metadata in the headers of an image sequence and to edit this information. For example, it is possible to adjust time and keycodes of an image sequence depending on a selected keyframe.

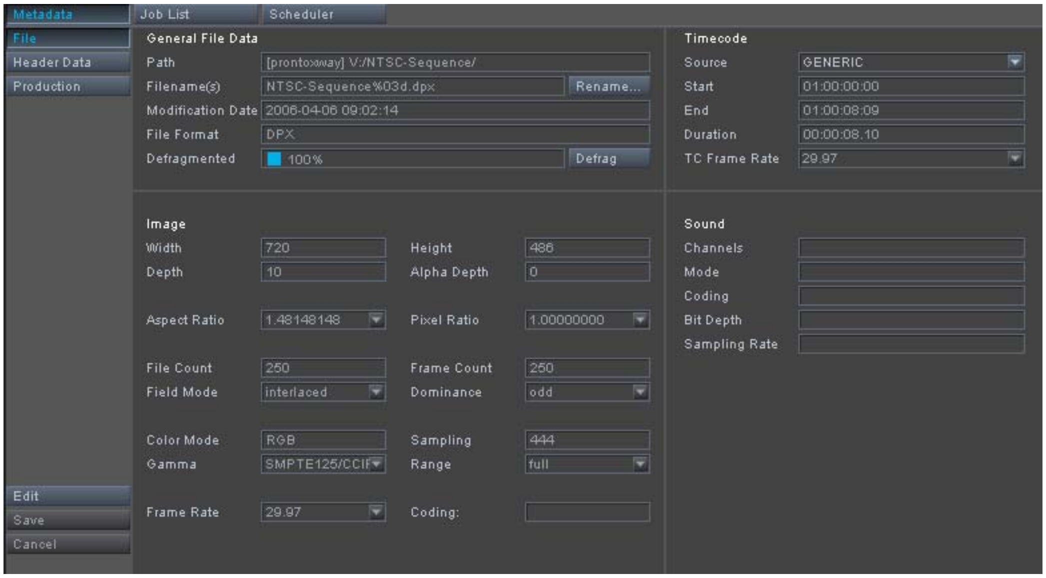


Figure 6: Selected metadata fields of a DPX file sequence

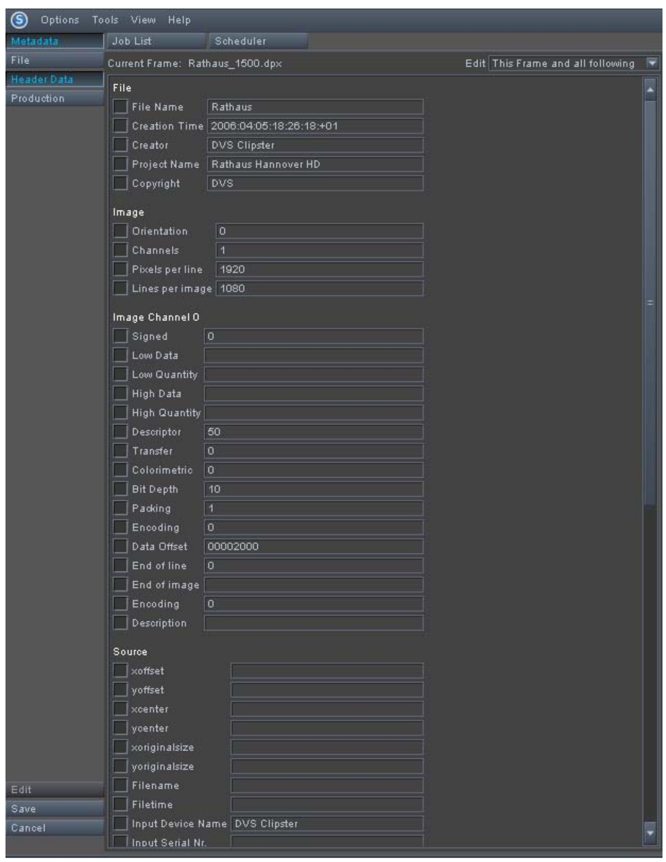
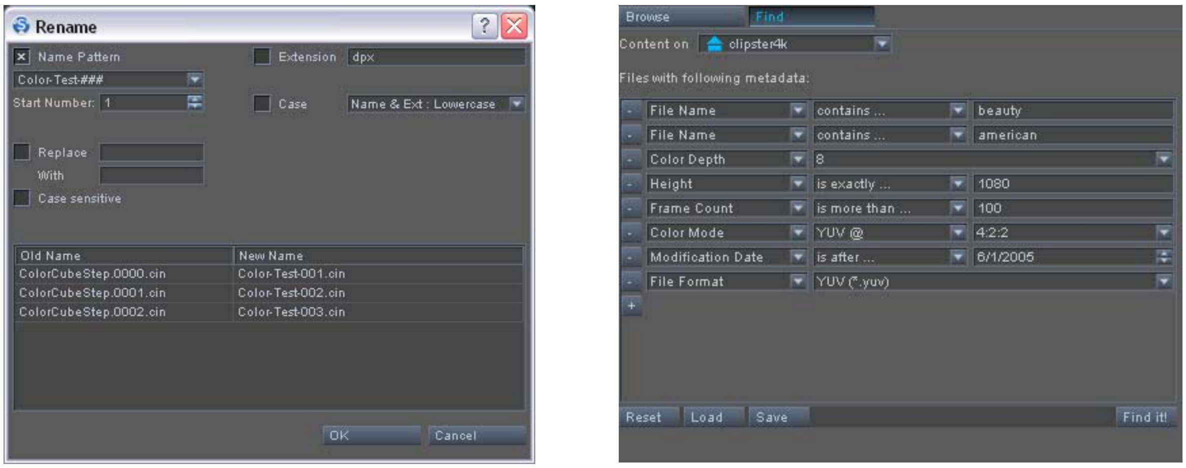


Figure 7: Editable header data of a single frame in a DPX sequence

5.3 Finding Content

In order to find a certain file, it is much faster to use a search tool that provides specific attributes for a search query, instead of browsing through several directories manually. This is a new working paradigm. In Spycer® a powerful search engine is implemented where important metadata attributes can be specified for a search query so that fast data localization in the whole network can happen.

Figure 8b shows the part of the user interface that is used to define search criteria based on the metadata that have been extracted by Spycer® when it detects that the essence has been added/modified.



(a) (b)

Figure 8: (a) Renaming function for file sequences, (b) searching for clips using extracted metadata.

5.4 Visual Browsing Tool

The visual browsing tool is the component handling clustering, filtering and selection of representative media items. It accesses the data structures created and filled by the indexing service and has a generic interface towards the presentation layer in order to allow the use of different visualization and interaction paradigms. The browsing tool has a state that is defined by the current data set and its cluster structure. It also keeps a history of the clustering and selection operations carried out so far, as well as their parameters. This allows implementing undo and redo functionality in interactive applications, as well as storing the users' browsing trails in order to improve the clustering and selection algorithms. Each plug-in of the browsing tool provides the following functionality for one feature: the clustering algorithm and optionally algorithms for selecting a subset of the current data and for selecting/creating representative media items for a data set.

From the user's point of view, the basic difference between content browsing and search is the limited knowledge of the user about how to formulate a query and about what to expect from the content set. Thus, the content browsing tool must support the user in building a query step by step, by trying to add new restrictions and reducing the content set when applying the chain of restrictions built up so far (cf. the ostensive model of developing information needs [CvR96]).

The basic workflow in the browsing tool is as follows: the user starts from the complete content set. By selecting one of the available features the content will be clustered according to this feature. Depending on the current size of the content set, a fraction of the segments (mostly a few percent or even less) is selected to represent a cluster. The user can then decide to select a subset of clusters that seems to be relevant and discard the others, or repeat clustering on the current content set using another feature. In the first case, the reduced content set is the input to the clustering step in the next iteration. The user can select relevant items at any time and drag them into the result list. A more detailed description of the content abstraction framework and its application in the browsing tool can be found in [BT07].

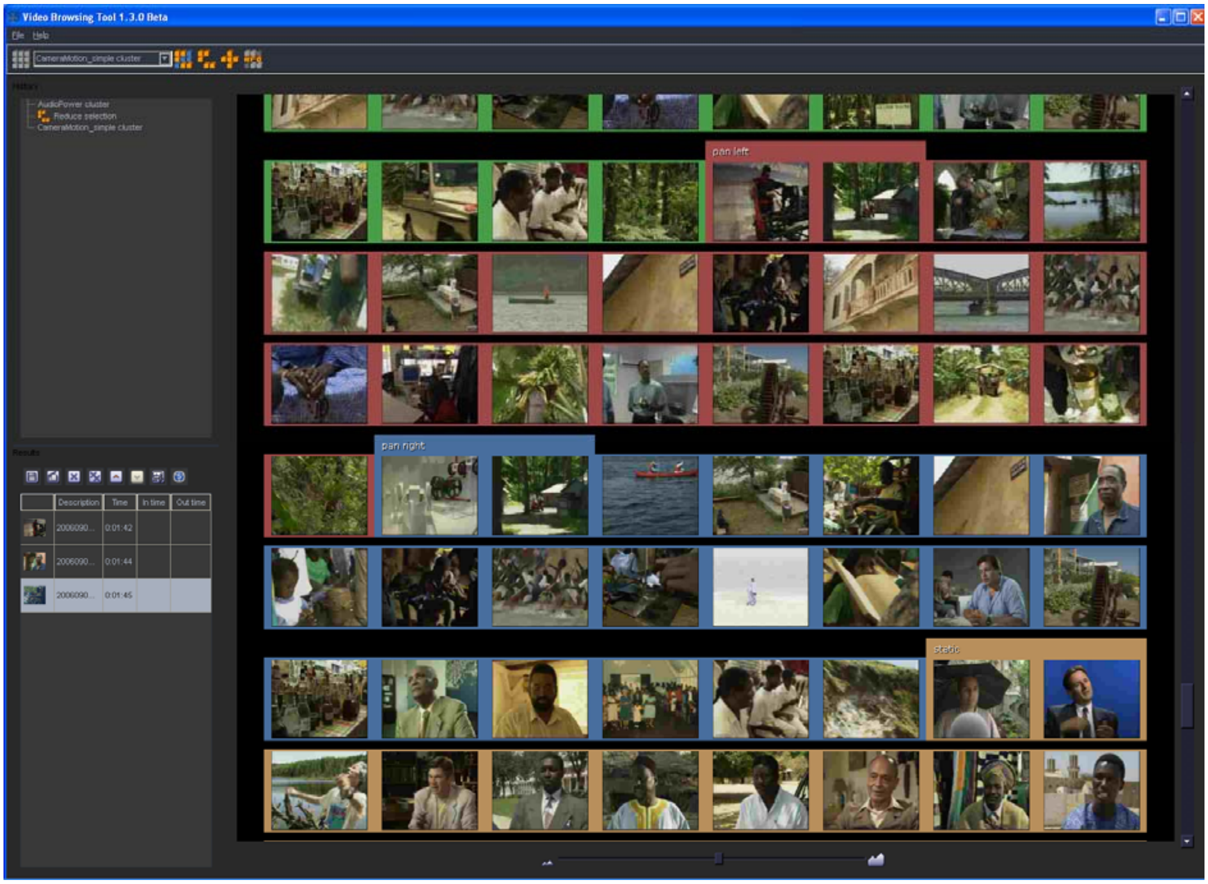


Figure 9: User interface of the visual browsing tool[2].

The user interface is a part that is not covered by the framework; the summarizer just defines an interface towards the presentation components. A screen shot of the browsing tool is shown in Figure 9. The central component is a light table view, which shows the current content set and cluster structure using a number of representative frames for each of the clusters. Colored areas around the images visualize the clusters, with the cluster label written above the first two images of the cluster. The size of the images in the light table view can be changed dynamically so that the user can choose between the level of detail and the number of images visible without scrolling. The tool bar at the top of the screen contains the controls for selecting the feature for clustering and confirming the selection of a subset. The light table view allows selection of multiple clusters by clicking on one of their member images. By double clicking an image in the light table view, a small video player is opened and plays the segment of video that is represented by that image. The size of the player adjusts relatively to the size of the representative frames.

On the left of the application window the history and the result list are displayed. The history window automatically records all clustering and selection actions done by the user. By clicking on one of the entries in the history, the user can set the state of the summarizer (i.e. the content set) back to this point. The user can then choose to discard the subsequent steps and use other cluster/selection operations, or to branch the browsing path and explore the content using alternative cluster features. At any time the user can drag relevant representative frames into the result list, thus adding the corresponding segment of the content to the result set. Items from the result list can directly be opened in the Spycer® application. The result list can then be saved as edit decision list (EDL). The export of the result as an AAF file is planned as a future feature.

---

[2] The data shown is from the BBC 2006 rushes data set. BBC 2006 Rushes video is copyrighted. The BBC 2006 Rushes video used in this work is provided for research purposes by the BBC through the TREC Information Retrieval Research Collection.

6 Conclusion

In this paper we have presented a novel peer-to-peer network based approach for content-management in post-production. This approach is capable of satisfying the demand for handling large data and supporting flexible workflows and dynamic setups.

The system monitors new and modified essence and extracts metadata from file headers and by using automatic content analysis tools. Metadata is stored locally at the peers, but is searchable from any client in the network. In addition to searching by the available metadata a browsing tool allows exploring the essence visually.

A limited version of the content management application, called Spycer® Basic, is provided by DVS for free. The software is available for Windows® and Linux® and can be downloaded at http://www.spycer.net.

Acknowledgements

The authors would like to thank their colleagues Franz Höller (now with HS-Art Digital Service), Harald Stiegler (JRS) and Christian Schober (JRS) for their support and feedback as well as the partners in the IP-RACINE consortium for the good cooperation. This work has been funded partially under the 6th Framework Programme of the European Union within the IST project "IP-RACINE" (IST-2-511316, http://www.ipracine.com).

7 References

[BST05] Werner Bailer, Peter Schallauer and Georg Thallinger, "JOANNEUM RESEARCH at TRECVID 2005 – Camera Motion Detection", Proceedings of TRECVID Workshop, pp. 182-189, Gaithersburg, MD, USA, Nov. 2005.

[BS06] Werner Bailer and Peter Schallauer, "The Detailed Audiovisual Profile: Enabling Interoperability between MPEG-7 Based Systems", Proceedings of 12th International Multi-Media Modeling Conference, pp. 217-224, Beijing, CN, Jan. 2006.

[BT07] Werner Bailer and Georg Thallinger, "A Framework for Multimedia Content Abstraction and its Application to Rushes Exploration", Proc. of ACM International Conference on Image and Video Retrieval, Amsterdam, NL, Jul. 2007.

[CvR96] Iain Campbell and Cornelis J. van Rijsbergen, "The ostensive model of developing information needs", Proceedings of COLIS-96, 2nd International Conference on Conceptions of Library Science, pp. 251-268, Copenhagen, DK, 1996.

[DCI] Digital Cinema Initiatives, LLC, Digital Cinema System Specification v.1.1, 2007.

[DMS-1] Material Exchange Format (MXF) – Descriptive Metadata Scheme-1, SMPTE 380M, 2004.

[DPX] File Format for Digital Moving-Picture Exchange (DPX), Version 2.0, SMPTE 268M, 2003.

[KLV] Data Encoding Protocol using Key-Length-Value, SMPTE 336M, 2001.

[MDDict] Metadata Dictionary – Registry of Metadata Element Descriptions, SMPTE RP210.8, 2004.

[MPEG7] ISO/IEC 15938, Multimedia Content Description Interface.

[MXF] Material Exchange Format (MXF) – File Format Specification, SMPTE 377M, 2004.

[P_Meta] P_Meta – The EBU Metadata Exchange Scheme, EBU Tech 3295, version 1.2, 2005.

[UMID] Unique Material Identifier (UMID), SMPTE 330M, 2000.

[XSD] XML Schema, W3C Recommendation, 2 May 2001, http://www.w3.org/XML/Schema