3

Canonical Processes of Semantically Annotated Media Production

Lynda Hardman,[1] Željko Obrenović[1] and Frank Nack[2]
[1]*Centrum Wiskunde & Informatica, Amsterdam, The Netherlands*
[2]*University of Amsterdam, Amsterdam, The Netherlands*

While many multimedia systems allow the association of semantic annotations with media assets, there is no agreed way of sharing these among systems. We identify a small number of fundamental processes of media production, which we term canonical processes, which can be supported in semantically aware media production tools. The processes are identified in conjunction with a number of different research groups within the community and were published in 2008 in a special issue of the *Multimedia Systems* journal (Vol. 14, No. 6). Our effort to identify canonical processes is part of a broader effort to create a rigorous formal description of a high-quality multimedia annotation ontology compatible with existing (Semantic) Web technologies. In particular, we are involved with work on specifying the structure of complex semantic annotations of non-textual data. This has resulted in the Core Ontology for Multimedia (COMM),[1] (Arndt et al. 2008) based on the MPEG-7 standard (MPEG-7 2001) and expressed in terms of OWL (Bechhofer et al. 2004).

We begin with a brief description of all canonical processes. Then, in order to illustrate canonical processes and connect them to the themes of this book, we describe two existing systems related to the personal photo management use case: CeWe Color Photo Book and SenseCam. CeWe Color Photo Book allows users to design a photo book on a home computer and have it printed by commercial photo finishers. SenseCam introduces

[1] `http://multimedia.semanticweb.org/COMM/`

Multimedia Semantics: Metadata, Analysis and Interaction, First Edition. Edited by Raphaël Troncy, Benoit Huet and Simon Schenk.

different approaches to structuring, searching, and browsing personal image collections, and tries to automatically locate important or significant events in a person's life.

3.1 Canonical Processes

The definition of the canonical processes of media production is the result of discussions with many researchers and practitioners in the community. This was initiated in a workgroup on 'Multimedia for Human Communication' at a Dagstuhl seminar 05091,[2] with a follow-up workshop at ACM Multimedia 2005 on 'Multimedia for Human Communication – From Capture to Convey'.[3] Our goal with these discussions and the open call for papers for the special issue of the *Multimedia Systems* journal was to establish community agreement on the model before presenting it here.

The special issue of the *Multimedia Systems* journal systematizes and extends this discussion, providing more elaborate identification of canonical processes and their mapping to real systems. In the special issue, which we summarize in this chapter, we identified and defined a number of canonical processes of media production and gave their initial formal description. The special issue is also accompanied with other papers that describe existing systems, the functionality they support and a mapping to the identified canonical processes. These companion system papers are used to validate the model by demonstrating that a large proportion of the functionality provided by the systems can be described in terms of the canonical processes, and that all the proposed canonical processes are supported in more than a single existing system. In this way we also wanted to emphasize that building a useful model must always include the agreement of a significant part of the community. The agreement on particular elements of the model and discussion about usability of the model for a particular domain are as important as – sometimes even more important than – detailed and rigorous formalization. In this special issue we present not 'yet another model', but an approach to building such a model, and the benefits of having a model that a significant part of the community agrees on. Our model is the result of long discussions in the multimedia community, and we present not only the result of this discussion, but also a number of system papers that discuss the model in a particular domain.

Based on an examination of existing multimedia systems, we have identified nine canonical processes of media production. Every process introduced into our model has at least several instances in existing systems. Our model, therefore, does not contain processes that are specific to particular systems. In the following sections we give a detailed description of these nine processes:

- *premeditate,* where initial ideas about media production are established;
- *create media asset,* where media assets are captured, generated or transformed;
- *annotate,* where annotations are associated with media assets;
- *package,* where process artifacts are logically and physically packed;
- *query,* where a user retrieves a set of process artifacts;
- *construct message,* where an author specifies the message they wish to convey;

[2] http://www.dagstuhl.de/en/programm/kalender/semhp/?semnr = 05091

[3] http://www.cwi.nl/~media/conferences/mhc05/mhc05.html

- *organize,* where process artifacts are organized according to the message;
- *publish,* where final content and user interface are created;
- *distribute,* where final interaction between end users and produced media occurs.

For each process, we give a brief explanation and state its inputs, outputs and the actors involved. While we give a name to each of the processes, these are meant to be used in a very broad sense. The textual description of each one specifies the breadth of the process we wish to express. Often in real life composite processes are also implemented that combine several canonical processes.

3.1.1 Premeditate

Any media creation occurs because someone has made a decision to embark on the process of creating – whether it be image capture with a personal photo camera, drawing with a drawing tool, professional news video, an expensive Hollywood film or a security video in a public transport system. In all cases there has been premeditation and a decision as to when, how and for how long creation should take place.

In all these cases what is recorded is not value-free. A decision has been made to take a picture of this subject, conduct an interview with this person, film this take of the chase scene or position the security camera in this corner. There is already a great deal of semantics implicitly present. Who is the 'owner' of the media to be created? Why is the media being created? Why has this location/background been chosen? Whatever this information is, it should be possible to collect it, preserve it and attach it to the media that is to be created. For this we need to preserve the appropriate information that can, at some later stage, be associated with one or more corresponding media assets.

3.1.2 Create Media Asset

After a process of premeditation, however long or short, at some point there is a moment of media asset creation. A device can be used to collect images or sound for a period of time, be it photo or video camera, scanner, sound recorder, heart-rate monitor, or MRI.

Note that in this process, we do not restrict the creation of a media asset to newly recorded information. Media assets can also be created in other ways. For example, images can be created with image editing programs or generated by transforming one or more existing images. The essence is that a media asset comes into existence and we are not interested in the method of creation *per se*. If the method is considered significant, however, then this information should be recorded as part of the annotation.

3.1.3 Annotate

The annotate process allows extra information to be associated with any existing process artifact. The term 'annotation' is often used to denote a single human user adding metadata to facilitate search. Here we view annotation as the broader process of adding more easily machine-processable descriptions of the artifact.

The annotations need not be explicitly assigned by a user, but may be assigned by an underlying system, for example by supplying a media asset as input to a feature analysis algorithm and using the extracted result to annotate the media asset. We make no distinction between whether annotations are selected from an existing vocabulary or machine-generated. If deemed relevant, the identity of the human assigner or the algorithm can be recorded in the annotation (Arndt et al. 2008).

We do not prescribe the form of annotations, but require that they can be created and associated with one or more artifacts. We also do not impose limitations on the structure of annotations, due to the high diversity of annotation formats in practice. In most semantically rich systems, however, the structure of an annotation may include a reference to a vocabulary being used, one of the terms from the vocabulary plus a value describing the media asset.

The annotation can refer to any artifact as a whole, but the annotation could also be more specific. In this case, an anchor mechanism is needed to refer to the part of the media asset to which the annotation applies (Halasz and Schwartz 1994). An anchor consists of a media-independent means of referring to a part of the media asset and a media-dependent anchor value that specifies a part of the media asset. For example, for an image this could be an area, for an object in a film a time-dependent description of an area of the image. For further discussion on anchor specifications, see Hardman (1998, p. 53).

3.1.4 Package

The process of packaging provides a message-independent grouping of artifacts. The output of this process is aimed at authors and developers, to help them to maintain process artifacts, and is unrelated to the final presentation organization. This process, for example, can assign a group of related media items and annotations an identity so that it can be retrieved as a unit. One of the simplest forms of packaging is putting related files in one directory, where the directory path provides an identity for the package.

We make a distinction between physical and logical packaging, where physical packaging reflects the organization of units in a database or file system, and logical packaging defines logical relations among items. For example, a Synchronized Multimedia Integration Language (SMIL) presentation is logically one presentation unit, but links media components physically packaged in many files in a distributed environment. On the other hand, a multimedia database can physically be packaged in one file, but contain many logical units.

3.1.5 Query

The query process selects a number of artifacts from a repository of artifacts. Up until now the processes we describe have concentrated on creating, storing and describing primarily media assets. These are needed for populating a media repository. Note that our definition of media repository does not necessarily imply the existence of a complex storage infrastructure, but we assume that systems have a repository where they keep media assets and other artifacts, in the simplest case a hierarchically organized file directory structure. Once there is a repository of artifacts it can be queried for components whose associated media assets correspond to specified properties.

We do not wish to use a narrow definition of the term 'query', but intend to include any interface that allows the artifacts to be searched, using query languages of choice or (generated) browsing interfaces that allow exploration of the content of the archive. It is worth noting that many systems that provide advanced query interfaces also provide support for other processes. For example, browser interfaces can, in addition to a simple query interface, organize intermediate results to present them to a user for feedback, and create temporary presentations that are then published and distributed to the user.

A query of the system may be in terms of media assets, or in terms of the annotations stored with the media assets. A query needs to specify (indirectly) the annotation(s) being used, and includes techniques such as query by example. The mechanisms themselves are not important for the identification of the process.

3.1.6 Construct Message

A presentation of media assets, such as a film or an anatomy book, is created because a human author wishes to communicate something to a viewer or reader. Constructing the message which lies behind the presentation is most often carried out by one or more human authors. When a viewer watches a film or a reader reads a book then some part of the intended original message of the author will hopefully be communicated. In order to give different processes access to the underlying intent, we include an explicit process which brings a processable form of the message into the system. Just as capturing a media asset is input into the system, so is the specification of the message an author wishes to convey.

In some sense, there is no input into the construct message process. However, the real input is the collection of knowledge and experience in the author her/himself. The output of the process is a description of the intended message. For example, a multimedia sketch system, such as described in Bailey et al. (2001), allows an author to gradually build up a description of the message. For the message to be machine processable the underlying semantics need to be expressed explicitly.

A query implicitly specifies a message, albeit a simple one, that an author may want to convey, since otherwise the author would not have been interested in finding those media assets. The query is, however, not itself the message that the author wishes to convey.

In general, we give no recommendation here for the expression of the semantics of the message. We expect that it contains information regarding the domain and how this is to be communicated to the user, but we do not assign anything more than a means of identifying a particular message.

3.1.7 Organize

While querying allows the selection of a subset of media assets, it imposes no explicit structure on the results of one or more queries. The process of organization creates a document structure that groups and orders the selected media assets for presentation to a user. How this process occurs is, again, not relevant, but may include, for example, the linear relevance orderings provided by most information retrieval systems. It also includes the complex human process of producing a linear collection of slides for a talk; creating

multimedia documents for the web; ordering shots in a film; or even producing a static two-dimensional poster.

The process of organization is guided by the message (the output of the construct message process). The organization depends on the message and how the annotations of the process artifacts relate to the message. For example, annotations concerning dates could be used to order assets temporally. The resulting document structure may reflect the underlying domain semantics, for example a medical or cultural heritage application, but is not required to. The structure may be color-based or rhythm-based, if the main purpose of the message is, for example, aesthetic rather than informative.

In the arena of text documents, the document structure resulting from organization is predominantly a hierarchical structure of headings and subheadings. The document structure of a film is a hierarchical collection of shots. For more interactive applications, the document structure includes links from one 'scene' to another. In a SMIL document, for example, `par` and `seq` elements form the hierarchical backbone of the document structure we refer to here (SMIL 2001).

3.1.8 Publish

The output of the organize process is a prototypical presentation that can be communicated to an end user. This serves as input to the publication process which selects appropriate parts of the document structure to present to the end user. The publication process takes a generic document structure and makes refinements before sending the actual bits to the user. These may include selecting preferred modalities for the user and displayable by the user's device. The resulting presentation can be linear (non-interactive, e.g. a movie) or non-linear (interactive, e.g. web presentation).

Publication can be seen as taking the document structure from the internal set of processes and converting it (with potential loss of information) for external use. Annotations may be added to describe the published document, for example the device or bandwidth for which the publication is destined. Annotations and alternative media assets may be removed to protect internal information or just reduce the size of the data destined for the user.

3.1.9 Distribute

Created content has to be, synchronously or asynchronously, transmitted to the end user. This final process involves some form of user interaction and requires interaction devices, while transmission of multimedia data to the user device goes through some of the transmission channels including the internet (streamed or file-based), non-networked media (such as a CD-ROM or DVD) or even analog recording media (e.g. film).

It is important to note that the term 'distribution' in our model has a much broader meaning than in classical linear production of media. It can also be used to describe interactive non-linear productions, such as games or other interactive presentations. The resulting system would implement a complex process including query, organize and publish processes in addition to distribution (end-user interaction). For example, some systems can have a final presentation where the storyline depends on user feedback. In this case, the presentation system would include the canonical processes query (to select next part of the story),

organize (to organize selected items coherently), publish (to create internal document ready for presentation) and distribute (to present and expose the control interface to user).

3.2 Example Systems

In order to illustrate canonical processes on real-world examples, and to connect them to the themes of this book, we describe two existing systems related to the personal photo management use case: CeWe Color Photo Book and SenseCam. CeWe Color Photo Book allows users to design a photo book on a home computer and have it printed by commercial photo finishers. SenseCam introduces different approaches to structuring, searching and browsing personal image collections, to try to automatically locate important or significant events in a person's life. More detailed description of these systems can be found in the special issue of the *Multimedia Systems* journal.

3.2.1 CeWe Color Photo Book

The *CeWe Color Photo Book* system, described by Sandhaus et al. (2008), allows users to design a photo book on a home computer, and then have it printed by commercial photo finishers. CeWe system inputs are images generated by users, mostly images from users' digital cameras. The system uses low-level feature analysis to select, group and lay out pictures, but it also allows manual authoring enabling users to override suggested selection and layout. The system introduces several steps related to feature extraction, clustering, selection, layout, manual authoring, sorting, ordering and printing.

Feature Extraction

The CeWe Color Photo Book system uses feature extraction methods to annotate images with low-level data. Inputs to this process are digital photos with the EXIF header as creation metadata. The photos undergo a sequence of steps, such as edge detection and color histogram extraction. In the terms of the canonical processes this is a *complex annotation process* involving the CeWe software as a computing actor, outputting one or more annotation artifacts (metadata) associated with the input photos.

Clustering and Selection

Following the annotation processes, a time clustering is performed on the photo set. Inputs to these processes are the photos with associated annotations from the feature extraction step. The clustering process is based on the creation time annotation. In terms of canonical processes, this time clustering is a *complex process* which consists of several instances of a *logical package process*.

The photo clusters are the basis for the selection process which chooses the photos to be included in the photo book. Inputs to this process are the clusters of annotated photos with additional user constraints. With constraints the user can state if similar, blurred or poor print quality images should be selected, and/or if brighter images should be preferred. The user can also limit the number of pages and the number of photos per page. In terms

of canonical processes, the selection process is a *complex process* with several instances of the canonical *query* and *organize* processes. Outputs of this query process are selected photo clusters, each cluster with one or more photos.

Layout

The layout process automatically arranges the photos over the pages and chooses the appropriate background for each photo book page. Inputs to this phase are the photo clusters, user preferences, as well as predefined templates for page layouts and backgrounds. User preferences include the style of the page layout and preferred backgrounds. The time clustering information is used to ensure that temporally related photos are kept together, preferably on the same photo book page. Color histogram annotations and user preferences are used to select the background for each page. In terms of canonical processes, the layout process is an instance of the canonical *organize* process. The result of this process is a preliminary photo book layout.

Manual Authoring

The automatically generated photo book layout can undergo various modifications and additions by the user. Alterations on individual photos can be made by 'filters' (such as sepia, black and white, sharpen) which we can describe as instances of the canonical process *transform* (which is an instance of the *create media asset* process). Other modifications include cropping, resizing, moving or rotating of photos, which do not change photos but modify the layout. The user can also delete unwanted photos or add additional photos and text annotations. In terms of canonical processes, this manual authoring is a complex process that, in addition to image transformations, also includes another instance of the *organize* process, as the user has the ability to reorganize images and pages. The result of the manual authoring process is the final photo book layout, which in terms of canonical processes is an instance of a *generic document structure*.

While the user is designing the photo book, he is always able to switch to a preview mode where he can see how the book will look when it is finally produced. In terms of canonical processes, this preview process can be seen as a *complex process* consisting of iterative *publish* and *distribute* processes.

Store and Order

When the user is satisfied with the photo book layout, he can store the layout (with links to the photos) in a physical file. This activity we may describe as an instance of the canonical *physical packaging* process.

To order a physical copy of the photo book, the user can send all required information over the internet or create a CD and send it by mail. This is primarily a *publish* process, with the photo album layout and the associated photos as input. Within this process the required resolution of the photos for the printing process is determined and the photos are scaled down to save bandwidth or space on a CD – that is, this process also involves a canonical *transform* process. The result of this process is a set of image files and a file describing the layout with additional annotations such as shipping and billing information.

Print

The printing process is another instance of a canonical *publish* process, where the abstract photo book layout and selected images are transformed into a physical product, the printed photo book. Inputs to this process are the transformed photos from the the CeWe software, with the order file. The printed photo book is then sent to the customer in a final *distribution* process. The canonical processes and their relation to photo book production are summarized in Table 3.1.

3.2.2 SenseCam

SenseCam (Lee et al. 2008) is a system for structuring, searching and browsing personal image collections. The main goal of the system is to locate important or significant events in a person's life. The system takes as input images created by a small wearable personal device which automatically captures up to 2500 images per day. The system processes these using feature analysis, allowing the user to browse the collection of images organized according to their novelty using different granularities (e.g. today, this week, this month). SenseCam processes images through phases related to capturing, structuring and display.

Capture/Upload

The user can wear the SenseCam device as a visual archive tool for recording images during a particular event or throughout daily life. Before actual capturing of images starts, the user is involved in a form of *premeditate* process, where her wish/motivation to record images for a period of time results in her action of wearing the camera and use of the software system to create the initial user profile. In a more personalized service, the user's motive for wearing the device could be explicitly recorded when she decides to embark on using the system, potentially leading to an adaptive interface optimized for a particular motive.

Wearing the SenseCam throughout the day automatically *captures* a number of photos, and *annotates* them with time stamps and other sensed data (such as ambient temperature and light levels), creating the initial set of media assets and metadata.

Processing and Indexing

A more intensive *annotate* process occurs during the processing time, after the user uploads the file to her computer. Photos with their initial metadata go through a series of content-based analyses to further add machine-generated *annotations*; similarities among the photos are analyzed to determine the boundaries of individual events; a landmark photo is determined from each event; and the novelty value is calculated for each event by comparing the similarities between all events on that day and all other events that occurred during the previous week. From this data, the associations among events are established, both within a day and within a week period.

Using this additional metadata, the system *packages* photos into discrete events so that they can be retrieved as the main units of searching and browsing. Most of these processes are instances of *logical packaging* as the generated metadata are separated from the actual

Table 3.1 Canonical processes and their relation to photo book production

Canonical process	CeWe system
Premeditate	**Capture:** Planning an event (e.g. a holiday trip) which is to be documented in the photo book. Premeditate potentially influenced by external decisions *Input:* Thoughts of author(s) / external decisions *Output:* Schedule/plan
Create Media Asset	**Capture:** Taking a photo *Input:* Spontaneous decision of the photographer, schedule/plan *Output:* Photo equipped with EXIF header (Creation Metadata) **Author:** Altering a photo (cropping, resizing, filtering, rotating) *Input:* Photo in the photo book *Output:* Altered photo **Creation of text annotations:** *Input:* Editor, photo book software, schedule/plan of event *Output:* Text annotation
Annotate	**Author:** Feature extraction on photos (color histograms, edge detection, ...) *Input:* Photos from the input photo set *Output:* Generated metadata
Package	**Capture:** Organizing photos in a separate folder *Input:* Photos, schedule/plan from premeditate *Output:* Folder with photos, identifier is the folder name **Author:** Automatic time clustering of photos *Input:* Photos, time metadata *Output:* Photo clusters **Storing the photo book layout on a hard disc** *Input:* Photo book description *Output:* Physical file with layout information
Query	**Author:** Selecting a subset of images from the clustered input photo set *Input:* Photo clusters, user parameters for photo selection *Output:* Altered photo clusters (subset)
Construct Message	**Capture:** Spontaneous decision to take a photo *Input:* Photographer and his ideas and thoughts *Output:* Decision to take a photo **Author:** Layout decisions for the photo book *Input:* Photographer and his ideas, thoughts, creativity *Output:* Human layout decisions
Organize	**Author:** Actual author process: organizing photos and text over the pages. Split into two steps: the first is automatically done by the CeWe software, the second refinements manually by the CeWe software user *Input:* Photos, text annotations, human layout decisions *Output:* Structured description of the photo book layout

(*continued overleaf*)

Table 3.1 (*continued*)

Canonical process	CeWe system
	Preparing a photo book order which includes additional annotations like shipping, payment information *Input:* Photo book layout *Output:* Photo book order, such as CD image or internet order
Publish	**Author:** Internal preview of the photo in the photo book software *Input:* Photo book layout *Output:* Rendered images of photo book pages **Print:** Turning the structural photo book description into a physical product *Input:* Photo book order *Output:* Manufactured photo book
Distribute	**Author:** Presenting the user the rendered preview images of the photo book *Input:* Rendered images of photo book pages *Output:* Screen presentation **Print:** Shipping the photo book to the customer *Input:* Manufactured photo book

stored photos; that is, they are only marked-up information that points to different parts of the photo set.

Described automatic processes add partial descriptions of the photos in order to kick-start the initial organization and presentation. Subsequently, metadata may be further enriched with user descriptions as the user provides further feedback during interactive searching and browsing at a later stage.

Accessing the Photos

To review past events, the user can visit the online SenseCam image management system, and *query* the system by explicitly selecting dates, or by typing the terms to search annotation text. The system also supports query by example. The result of a query is a structured set of pointers to the subset of the archived photos ranked by the content-based similarity measures.

Results of the user query are further *organized* for the final presentation. This involves creating an interface template that specifies the sizes of each image according to the novelty assigned to each of the selected events and associated attributes. The output of this process is the visual summary of landmark events to be displayed to the user.

In the *publish* process, the organized information is displayed on the web interface in a comic-book style layout.

It is important to note that generating the final presentation also involves a *construct message* process, where the message emphasizes the novelty of events detected in the photos. The inputs to this process are the system parameters, such as relative size of images, as well as the customized user parameters. The system parameters are determined

Table 3.2 Description of dependencies between visual diary stages and the canonical process for media production

Canonical process	SenseCam image management system
Premeditate (1)	The user deciding to use a wearable camera to capture images of a day's events or of significant events such as going to a wedding, birthday party, zoo which (s)he would like to be recalled and reviewed at some later stage *Input:* User intention/motivation to record images *Output:* Decision to wear the device and access the system; initial user profile that the user provides on the initial online registration form
Create Media Asset (2)	Images passively captured by the SenseCam wearable camera *Input:* Sensor data that triggers automatic photo capture *Output:* Raw images along with sensor file
Annotate (3)	All images automatically time-stamped on download of images from SenseCam to computer *Input:* Output of (2) *Output:* Time-stamped SenseCam images Automatically describe each image in terms of its low-level visual features such as color, texture, or edge *Input:* Time-stamped images from computer *Output:* Annotated images During user interaction, user adds text annotation to events to add value to his/her archive
Package (4)	Automatically segment the annotated images into events *Input:* Annotated images from (3) *Output:* Images organized into events
Query (5)	User selects a particular date, week, month, or a range of dates; or types in text query; or requests all similar events by clicking on Find Similar button *Input:* Images from (4) and query statement (specific date range or text query terms or an example image representing an event) *Output:* List of events and their relative uniqueness rating
Construct Message (6)	Setting display options such as the number of events to be displayed on the page; setting the speed of slideshow on each event, etc. *Input:* User's intention to modify the presentation parameters to suit her interaction/viewing style and preferences *Output:* Modified set of presentation parameters, to be used for this user once the process has occurred
Organize (7)	Creating an interface template that will emphasize the most important events in a visual manner to the user *Input:* List of events with importance values from (5) *Output:* Summary of landmark events to be prepared for display

(*continued overleaf*)

Table 3.2 (*continued*)

Canonical process	CeWe system
Publish (8)	Selecting appropriate events and their metadata to be presented on the screen (web interface); alternatively, generating appropriate formats in PDF file (for print-out) or on DVD (for offline interactive browsing on TV screen); generating downsized versions suitable for a mobile phone or PDA consumption *Input:* Results of process (7) and all associated presentation-specific information *Output:* Generated presentation formats
Distribute (9)	Displaying the interactive visual summary on the web interface; printing out the daily summary in PDF format; inserting the generated DVD on interactive TV and browsing with a remote controller; transmitting to a mobile phone or a PDA, etc. *Input:* Results of process (8) *Output:* Viewing on a web browser, on a printed paper, or interacting with a DVD player

by the designer at the design time, but the user can also modify some of them. For example, by default, the number of events presented on a page (whether it is for a single day or multiple days) is set at 20, but during browsing the user can adjust this value, enabling her to partially construct a message. Dependencies between visual diary stages and canonical processes for media production are summarized in Table 3.2.

3.3 Conclusion and Future Work

In this chapter we have described a small number of fundamental processes of media production, termed 'canonical processes', which can be supported in semantically aware media production tools. The processes are identified in conjunction with a number of different research groups within the community and are being published in a special issue of the *Multimedia Systems* journal. In order to introduce canonical processes and connect them to the themes of this book, we have described two existing systems related to the personal photo management use case: CeWe Color Photo Book and SenseCam.

The focus of our work is not on rigorous formalization of canonical processes, but on their identification and mapping to real systems. We do, however, see the need for future work to link our current descriptions into a higher-level ontology, and to specify the structure of annotations more precisely. Describing the processes using a foundational ontology, such as DOLCE (Gangemi et al. 2002), provides a solid modeling basis and enables interoperability with other models. DOLCE provides description templates (patterns) for the specification of particular situations and information objects. A next step in the specification of the ontologies would be to express these as specializations of the DOLCE model – in particular, the DOLCE situations.