# Canonical Processes for Creating Personalized Semantically-rich Multimedia Presentations

Ansgar Scherp
OFFIS - Institute for Information Technology
Oldenburg, Germany
ascherp@ics.uci.edu

## ABSTRACT

Authoring of multimedia content can be considered as composing media assets such as images, videos, text, and audio in time, space, and interaction into a coherent multimedia presentation. Personalization of such content means that it reflects the users' or user groups' profile information and context information. Enriching the multimedia content with semantically-rich metadata allows for a better search and retrieval of the content. To actually create personalized semantically-rich multimedia content, a manual authoring of the many different documents for all the different users' and user groups' needs is not feasible. Rather a (semi-)automatic authoring of the content seems reasonable.

We have analyzed in detail today's approaches and systems for authoring, personalizing, and semantically enriching multimedia presentations. Based on this analysis, we derived a general creation chain for the (semi-)automatic generation of such content. In this paper, we introduce this creation chain. We present our software engineering support for the chain, the component framework SemanticMM4U. The canonical processes supported by the creation chain and SemanticMM4U framework are described in detail. We also provide an explicit mapping of SemanticMM4U framework components to the processes and argue for the benefits of defining canonical processes for creating personalized semantically-rich multimedia presentations.

## Categories and Subject Descriptors

H.5.1 [**Information Interfaces and Presentation**]: Multimedia Information Systems; H.5.4 [**Information Interfaces and Presentation**]: Hypertext/Hypermedia

## General Terms

Management, Human Factors, Standardization, Algorithms



## Keywords

Multimedia Content, Multimedia Authoring, Multimedia Personalization, Multimedia Semantics, Semantics Derivation, Canonical Processes, Media Production, Component Framework, Component Technology

## 1. INTRODUCTION

Personalization of multimedia content and the dynamic creation of such content has been object of research for more than a decade. Today, we find many different scientific approaches as well as industrial solutions that provide personalized content to their users. However, the creation of personalized multimedia content is still a challenging and tedious task (cp. [9]). A practical support for the dynamic authoring of such content is neither provided by industrial solutions nor research projects. In addition, the systems we find today exploit semantically-rich metadata for selecting media assets and organizing them into personalized multimedia content. However, this semantically-rich and highly valuable source of information is not considered any further [32]. Once the multimedia content has been created, the semantically-rich information exploited is thrown away. The created multimedia presentations carry either none or only a small part of the metadata that actually has been exploited for the creation task.

Consequently, we developed a generic and at the same time practical support for the dynamic authoring of personalized semantically-rich multimedia presentations. This support not only exploits the semantically-rich metadata for the media asset selection and organization into the personalized multimedia content but makes this metadata explicit and available by integrating it into the created multimedia presentations. In addition, it allows for deriving further metadata during the actual organization of the media assets into the multimedia content. The result of this research is a software engineering approach, the component framework SemanticMM4U (short for "Semantic MultiMedia For You") for authoring personalized semantically-rich multimedia presentations [32, 27]. The overall goal of the SemanticMM4U component framework is to ease the creation of personalized multimedia content and to derive semantically-rich information about it. The SemanticMM4U framework is intended as an integrated approach as we believe that the different processes can be well supported and realized by different actors in the multimedia community.

The remainder of this paper is organized as follows: First, we introduce in Section 2 the general creation chain for authoring personalized semantically-rich multimedia presen-

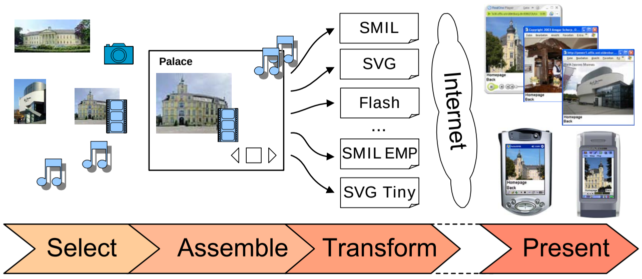


**Figure 1: General creation chain for personalized semantically-rich multimedia content**

tations. In Section 3, we present in detail the canonical processes involved in the different phases of this creation chain. In Section 4, we describe the goals and architecture of the SemanticMM4U component framework and provide a mapping of the framework components to the canonical processes. Finally, we present concrete applications using the SemanticMM4U framework and argue for the benefits of defining canonical processes in order to help the different actors in the multimedia community to work together.

## 2. CREATION OF PERSONALIZED MULTIMEDIA CONTENT

For the creation of personalized multimedia content, we find many different research approaches and industrial solutions. Well known examples from research are the Cuypers engine [17, 36], the projects Opéra [3] and WAM [23, 24], the Samp*Le* framework [14, 15], and the Standard Reference Model for Intelligent Multimedia Presentation Systems [7, 13]. Examples from industry are, e. g., the HotStreams system [18] and the online bookstore Amazon [2] for text-centric content. These systems exploit semantically-rich information to conduct the multimedia content creation task.

Based on an extensive analysis of these and further approaches and systems [27, 32], we developed a general creation chain for authoring personalized semantically-rich multimedia content [27, 32, 30, 31]. This creation chain is illustrated in Figure 1. As the schematic depiction shows, it consists of four phases. These phases span from media query, media organization, to the actual publishing and delivery of the content to the users. In the following, we present the phases of the creation chain from left to right.

*Select.* In the first phase, mono-media assets such as images, text, audio, and video are selected from the media storages. Media storages can be any of today's off-the-shell media databases such as Oracle10g interMedia [25] or the image database QBIC [19] from IBM. They can also be any self-developed solution for storing and managing media assets and their metadata. The query parameters to the media storages are among others the users' or user groups' profile information such as knowledge, preferences, interests, and needs [8, 22, 16] and context information like location, time, and characteristics of the end device [34, 12, 35] (described in detail in [27, Sec. 6.2]). For example, in the context of a mobile tourist guide the media assets of a specific sight are selected based on the user's current location and the display capabilities of the mobile device. As a specific user of the guide might prefer an acoustic presentation rather than a visual experience, only audio clips are selected.

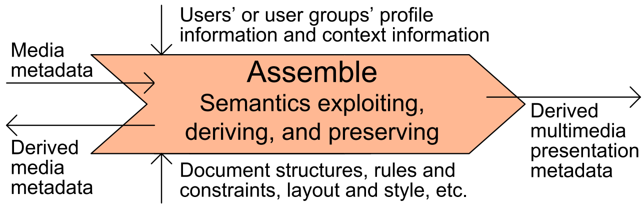


**Figure 2: Semantics exploiting, deriving, and preserving in the Assemble phase**

Besides returning already existing media assets that fulfill the query, also new media assets can be created on the fly. For example, if an image is requested in thumbnail size but is actually stored in full resolution, a downscaled version of the image can instantly be created to fulfill the request (see [27, pp. 172]).

*Assemble.* In the next phase, the selected media assets are organized in time and space into a coherent, structured multimedia presentation. In addition, navigational interaction in form of hyperlinks can be defined. The multimedia presentations are created in an internal, tree-based multimedia content representation model [30]. This internal representation model abstracts from the features and syntax of today's presentation formats such as SMIL [41], SVG [40], and Flash [1]. As depicted in Figure 2, different information can be exploited for creating the personalized multimedia content. These are the users' or user groups' profile information and context information as well as arbitrary external document structures, rules and constraints, layout and style information, metadata associated with the media assets, and any other kind of semantic information. For example, in the context of a mobile tourist guide the layout of a sight presentation is dependent on the display size of the used mobile end device. This layout can be defined with help of templates, providing different styles for different users or user groups such as children and adults. Taking the individual user's preferences into account, the map of the explored area only shows the sights that match the given profile.

While organizing the media assets into structured multimedia content, new metadata can be derived from the created content and the sources used to create it. As shown in Figure 2, this metadata can be derived in two directions, for the created multimedia content as well as for the media assets used to create the content [32]. For example, arranging a set of images in a page-based album presentation allows for deriving information about the semantics of the created presentation and its pages based on the semantics of the used media assets. In the other direction, placing a text close to an image can be interpreted as a textual description of the image. Apart from any possible metadata derived, the created multimedia content carries at least the already existing metadata of the employed media assets. The existing metadata as well as any newly derived metadata are preserved by associating it with the created multimedia content. Figure 2 shows this exploitation, derivation, and preservation of metadata.

*Transform.* In the subsequent phase—called the *last mile*—the multimedia content represented in the internal tree-based model is transformed into concrete multimedia presentation formats [31]. Thus, we make the multimedia presentation available to the users. Figure 1 shows the transformation into the standardized formats SMIL and SVG and the industry format Flash, targeted at Desktop PCs. If the multimedia content is created for a mobile device, it is transformed to presentation formats targeted at mobile devices such as the Extended Mobile Profile of SMIL [41] and SVG Tiny [39]. Besides the actual multimedia content, also the metadata of the created content is transformed into the target presentation formats' syntax. Thus, the metadata are made available with the created multimedia presentations. This allows other applications to process and to use it. For example, search engines can use the provided semantics to better fulfill the information needs of their users [32].

*Present.* Finally, the multimedia presentations transformed into the target presentation formats are distributed to the (mobile) end devices of the users or user groups. Here, existing multimedia player software for the standardized presentation formats are used to render the presentations for the actual consumption. In addition, as mentioned above the content can also be gathered by other systems like search engines to further process and present it to the users.

The general creation chain for authoring personalized semantically-rich multimedia content presented in this section is a generalization of related approaches and systems in the field. As such, the general process chain is generic in a sense that it can be employed for generating personalized multimedia content in most different domains like tourism, e-learning, and sports news. As we will see in Section 4, this is reflected by implementing the creation chain in a generic software framework. However, the general creation chain is not to be seen as an alternative to the canonical processes for media production. Rather, it is to be seen as a subset of the canonical processes developed on an extensive analysis of related work in the field of multimedia personalization. This subset is described in terms of the canonical processes for media production in the following section.

## 3. PROCESSES FOR AUTHORING PERSONALIZED MULTIMEDIA CONTENT

In the previous section, we have introduced the general creation chain for creating personalized semantically-rich multimedia presentations. Here, we present in detail which canonical processes for media production are used by the general creation chain. We present the data structures required as input and the data structures provided as output. The data structures are described in terms of the canonical processes. For a detailed description of the inside work of the presented processes, we refer to earlier publications.

As described in Section 2, the general creation chain for personalized semantically-rich multimedia presentations consists of four phases. These four phases support eight canonical processes for media production, organized in one basic media production process and three complex media production processes. The first phase of the creation chain provides the complex process Select. It employs specializations of the basic canonical processes Query and Create Media Asset. The second phase provides the complex process Assemble. It is a composition of the two processes Assemble Organize and Assemble Annotate, which are specializations of the basic processes Organize and Annotate. It also supports the basic process Construct Message to create the message describing how to organize the media assets. The third phase realizes the complex process Transform, a combination of specializations of the basic processes Publish and Package. Finally, the last phase provides with the process Present a specialization of the basic process Distribute. The processes are described along the phases of the general creation chain from left to right. Table 1 summarizes the provided processes.

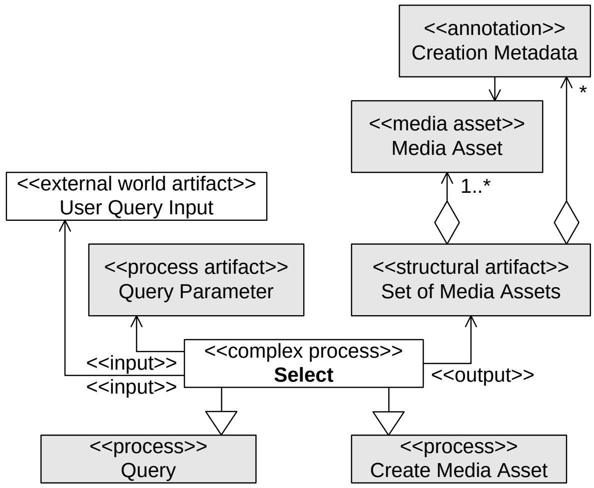


**Figure 3: UML diagram of the Select process**

### 3.1 Select

**Input:** Query Parameter, User Query Input (optional)
**Output:** Set of Media Assets
**Description:** The complex process Select depicted in Figure 3 selects media assets from a media storage according to the Query Parameter stated as input. The Query Parameter includes among others the users' or user groups' profile information and context information (see Section 2). The Select process can optionally also have some User Query Input. Here, the users specify some query parameters on the graphical user interface of the application. For example, the $x$SMART application [29] for authoring personalized photo albums allows the users to specify selection criteria for the photos such as exposure, similarity, sharpness, time, location, and whether the photos are indoor or outdoor shots. Although, this User Query Input is very similar to the Query Parameter, it is mentioned here explicitly to show that there can be query input from the external world to the Select process.

The Select process retrieves media assets from a media storage according to the provided user profile information and context information by means of the media assets' metadata. It determines a ranked list of media assets that are of most relevance in regard of the request. For this task, the complex process employs the basic media production process Query. The result set of the query is represented as a Set of Media Assets. As shown in Figure 3, a Set of Media Assets is a data structure that aggregates the two canonical data structures Media Asset and Creation Metadata. It is returned as output of the process.

The Creation Metadata can be the Media Asset metadata stored in the database that have once been created when the Media Assets have been created. However, in some cases the Query process, i. e., the underlying media storage cannot provide appropriate Media Assets directly. Then, the Select

| Canonical processes | Mapping of creation chain phases with canonical processes |
|---|---|
| Create Media Asset | Part of *Select* phase. On the fly creation of media assets.<br>Input: A set of existing media assets.<br>Output: A set of transformed media assets. |
| Annotate | Part of *Assemble* phase. Annotation of multimedia content with semantically-rich metadata.<br>Input: Multimedia content in abstract representation model.<br>Output: Multimedia content annotated with semantically-rich metadata. |
| Package | Part of *Transform* phase. Packaging of published multimedia presentations for distribution.<br>Input: Multimedia content in abstract representation model.<br>Output: Logically or physically packaged multimedia content in representation model. |
| Query | Part of *Select* phase. Query of media assets based on user profile and context information.<br>Input: Query parameters.<br>Output: A set of existing media assets, e.g., a part of a user photo collection. |
| Construct Message | Part of *Assemble* phase. Creates the message how to organize the media assets.<br>Input: Profile and context information as well as arbitrary content organization information.<br>Output: Message how to organize the media assets in time, space, and interaction. |
| Organize | Part of *Assemble* phase. Composition of media assets into multimedia content.<br>Input: Set of selected media assets.<br>Output: Multimedia content in abstract representation model. |
| Publish | Part of *Transform* phase. Transformation of multimedia content into concrete presentation formats.<br>Input: Multimedia content in abstract representation model.<br>Output: Multimedia presentation in concrete format. |
| Distribute | Part of *Present* phase. Delivering and rendering of multimedia presentations to end users.<br>Input: Multimedia presentation in concrete format.<br>Output: Real world projection of the multimedia presentation. |

**Table 1: Mapping between canonical processes and phases of the creation chain**

process has to create the requested Media Assets on the fly using the basic process Create Media Asset. The Query Parameter and the optional User Query Input are used for this creation task. They are the input to the Create Media Asset process, which instantly creates an appropriate Media Asset. The on the fly created Media Assets are then added to the result set of the media query as if they were already existing in the media storage. Examples where Media Assets can be generated on the fly are, e.g., when an image is requested in a specific size, a video needs to be adapted to certain network conditions, or a Media Asset needs to be provided in a specific format. Besides creating the actual Media Asset, also some Creation Metadata are generated. An example of a media storage that creates Media Assets on the fly is presented in [27, Sec. 8.7.3].

The Create Media Asset process is not obligatory in our creation chain. If a concrete application uses an implementation of the Select process, i.e., uses a media storage that does not support or integrate the Create Media Asset process, the result set will just not contain any Media Assets created on the fly.

## 3.2 Assemble Organize

**Input:** Set of Media Assets, Organize Message

**Output:** Multimedia Content

**Description:** In the basic process Assemble Organize depicted in Figure 4, the Set of Media Assets determined in the Select process are assembled and composed in time and space into coherent personalized multimedia content. In addition, navigational interaction in form of hyperlinks can be defined with the created multimedia content. These hyperlinks can either refer to arbitrary points in time within the multimedia presentation or refer to any external resources on the Internet including another personalized multimedia presentation.

The created multimedia content is understood as a structured artifact. It defines the relation of the media assets in the three dimensions time, space, and interaction as it is supported by today's presentation formats like SMIL, SVG, and Flash. How the Media Assets are arranged in time, space, and interaction is influenced and defined by the Organize Message, created by a specialization of the Construct Message process (not shown in the UML diagram). The Organize Message includes the users' or user groups' profile information and context information, external document structures, rules and constraints, layout and style information, or any other kind of semantic information [27]. For organizing the Set of Media Assets, also the Media Assets' Creation Metadata is taken into account. Thus, the Assemble Organize process leverages any information provided for the multimedia composition (as depicted in Figure 2 of the general creation chain). The Select and Assemble Organize processes are not necessarily conducted in a strict sequential order. Rather, the two processes are typically conducted in loops of selecting and organizing Media Assets.

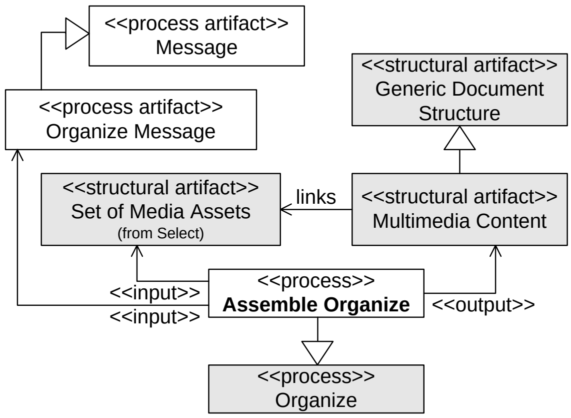


**Figure 4: UML diagram of the Assemble Organize process**

The personalized multimedia content is represented in the Assemble Organize process in an internal, tree-based representation model [30]. It is applicable for the most different domains such as tourism, e-learning, and sports news. The representation model abstracts from the different syntax and features of today's presentation formats such as SMIL, SVG, and Flash. However, it captures the central aspects of multimedia modeling [4, 5, 20], which are the temporal course, spatial layout, and interaction possibilities of the presentation with the users. The nodes of the tree-based representation model are either Media Assets such as audio, video, text, and images, basic composition operators like a Parallel or Sequential presentation of Media Assets, or complex and sophisticated composition operators designed to solve application-specific multimedia composition and personalization tasks.

Output of the Assemble Organize process is the created personalized multimedia content in the internal representation tree. It is provided by the data structure Multimedia Content derived from the canonical data structure Generic Document Structure. The Multimedia Content contains links to the employed Media Assets. The Multimedia Content is enriched by semantically-rich metadata in the Assemble Annotate process described in the following section. Subsequently, the Multimedia Content is passed to the Transform process presented in Section 3.4 in order to be transformed from its abstract representation into the concrete presentation formats.

## 3.3 Assemble Annotate

**Input:** Multimedia Content, Set of Media Assets, Organize Message (optional)

**Output:** Multimedia Content Annotation, Media Asset Annotation

**Description:** While the Media Assets selected in the Select process are organized and assembled in time, space, and interaction into coherent multimedia content in the Assemble Organize process, new metadata can be derived in the Assemble Annotate process that semantically describes the created multimedia content as well as the used Media Assets. As shown in Figure 5, input to this metadata derivation process is the Multimedia Content provided by the Assemble Organize process, the Set of Media Assets from the Select process, and the optional Organize Message exploited in the Assemble Organize process to create the Multimedia Content.

As depicted in Figure 2 of the general creation chain, the Assemble Annotate process allows for deriving metadata in two directions, for the created personalized multimedia content as well as the Media Assets used [32].

For the created personalized multimedia content, semantics is derived among others from the Media Assets' metadata and the characteristics of the created Multimedia Content. For example, the parallel organization of a set of images in a page-based album presentation semantically means that these media assets belong together and placing a text at the top of this page can be interpreted as a headline describing the album page. In addition, any other of the semantically-rich information exploited in the Assemble Organize process, i. e., the Organize Message can be used to derive further metadata about the created content. The derived semantics for the personalized multimedia content is made explicit with its transformation in the final presentation formats. This is conducted in the complex process Transform presented in Section 3.4.

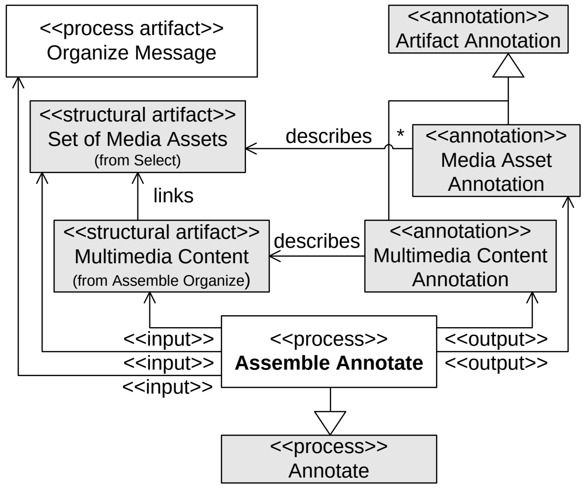


**Figure 5: UML diagram of the Assemble Annotate process**

In regard of deriving semantics for Media Assets, one can interpret a set of media assets that are used together in the same section of a presentation as defining and sharing a common semantic concept. This concept can then be associated as metadata to each Media Asset in this set. A system that allows for such semantics derivation for a selection of images is the image database El Niño [26]. Another example for deriving metadata about Media Assets is to identify that two images show the same building or person. This can be conducted based on the similarity of the images such as their location, color distribution, or face detection and face recognition techniques. The derived Media Asset metadata is back-stored into the media storages from which the assets are originally selected from.

As not all semantic derivation is hundred percent reliable, we added a value of reliability to each piece of derived information. This value of reliability is depending on the input data used for the derivation as well as the method of derivation. For example, in the domain of authoring photo albums one can base on statistical probabilities of real photo album data to determine the reliability value of automatically detecting, e. g., the title of a photo album page or the caption of an image [6, 33]. In a semi-automatic setting, the users provide the feedback for the correctness and reliability of the derived semantics.

The Assemble Annotate process is triggered by the Assemble Organize process. It can be triggered off arbitrarily many times. In fact, there can be multiple organize and annotate loops. As the Multimedia Content is a tree-based multimedia content representation model, it supports these loops by allowing to pass parts of the tree to the Assemble Annotate process. The representation model of the Assemble Organize process is designed such that each node can carry metadata with an expressiveness of the Resource Description Framework (RDF) [37]. We selected RDF as it supports references to other (media) nodes within the same document. It also supports to model information about the derived metadata such as the mentioned reliability as well as the source and time of derivation. Using this metadata model, annotations

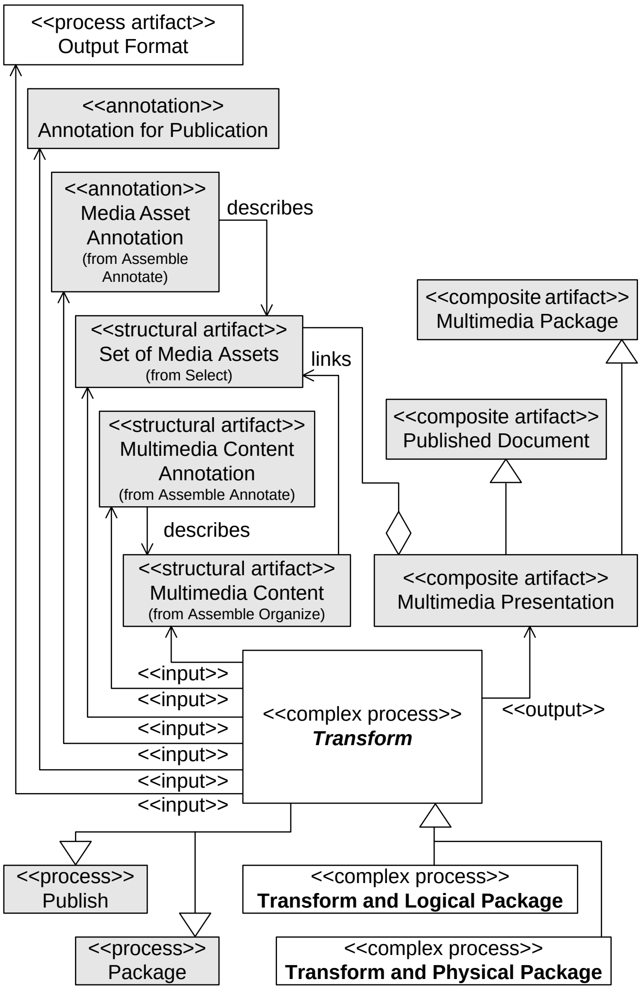


**Figure 6: UML diagram of the Transform process**

of the representation tree's nodes such as the Media Assets and composition operators like Parallel and Sequential can refer to any other Media Asset or operator. For example, a text can have a reference stating that it is a description of an image. The reference carries information which component of the system derived this correlation, when was it derived, and how reliable it is. We use the RDF-based metadata model to annotate the personalized Multimedia Content as well as the Media Assets. The Assemble Annotate process provides support for different (semi-)automatic semantics derivation methods and techniques. These can be rules, knowledge-bases like thesauri and taxonomies, or plain programming of the semantics derivation functionality [32].

The semantics derived in the Assemble Annotate process is returned as the canonical data structures Multimedia Content Annotation for the Multimedia Content and Media Asset Annotation for the Media Assets. Both are specializations of the generic data structure Artifact Annotation.

## 3.4 Transform

**Input:** Multimedia Content, Multimedia Content Annotation, Set of Media Assets, Media Asset Annotation, Annotation for Publication, Output Format

**Output:** Multimedia Presentation

**Description:** In the complex process Transform depicted in Figure 6, the personalized Multimedia Content in the abstract representation model is transformed to the syntax and features of the concrete presentation formats [31]. This is realized by a specialization of the canonical process Publish. Besides the transformation of the actual multimedia content, also the metadata that describe the content are mapped to the target formats' syntax. Thus, input to the Transform process are besides the Multimedia Content created in the Assemble Organize process the derived Multimedia Content Annotation and Media Asset Annotation from the Assemble Annotate process.

Subsequent to the transformation of the personalized semantically-rich multimedia content to the concrete presentation formats, the presentations with their Media Assets are either logically or physically packaged. Thus, also input to the Transform process is the Set of Media Assets, i. e., the Media Assets with their Creation Metadata from the Select process. The Media Assets are in particular important for packaging the Multimedia Content in a physical output document. Here, the specialized complex process Transform and Physical Package is employed. It is applied for presentation formats that physically package the content structure with the Media Assets such as Flash. The complex process Transform and Logical Package is a specialization of the Transform process, which is applied for presentation formats that refer to the used Media Assets by links only. Examples are SMIL and SVG as well as their mobile subsets.

The input Annotation for Publication provides creation metadata of the entire presentation such as copyright, title, and author of the presentation. It is typically provided by the concrete personalized multimedia application using the Transform process. The Output Format specifies into which concrete presentation format the multimedia content is transformed. It is dependent on the users' preferences and the supported presentation formats and capabilities of the end device.

The Multimedia Content in the representation model is transformed by applying application-independent transformation algorithms [31] into different presentation formats like SMIL, SVG, and Flash. While transforming the content, also its RDF-based metadata are collected. This metadata are stored with the multimedia content in the final presentation format. This can be conducted in two ways: If the presentation format supports integrated storage of the multimedia content and the content's metadata, this approach is pursued. As RDF-based metadata can be serialized into XML [38], the metadata can be integrated in the header of XML-based presentation formats such as SMIL and SVG using the <metadata>-tag. If an integrated storage of metadata and multimedia content is not provided, the RDF-based metadata are stored in a separate file. An example is the industry standard and binary presentation format Flash.

Output of the Transform process are the personalized multimedia presentations in the concrete formats like SMIL, SVG, and Flash, together with their metadata. These personalized semantically-rich multimedia presentations are packaged into the data structure Multimedia Presentation, a specialization of the canonical data structures Multimedia Package and Published Document. The Multimedia Presentation is passed to the Present process described next.

## 3.5 Present

**Input:** Multimedia Presentation
**Output:** Real World Projection
**Description:** In the last phase of our creation chain, the basic process Present makes the personalized multimedia presentations accessible to the users. Input to the process is a Multimedia Presentation from the Transform process, comprising the personalized semantically-rich multimedia presentation in a concrete presentation format. The presentation is delivered to the users' (mobile) end device for the actual rendering and consumption. A rendering environment puts the personalized multimedia presentation into life. It is responsible to calculate and render the temporal and spatial course of the presentation as well as to handle the user interaction in form of hyperlinks. If a hyperlink refers to a point in time within the presentation, the rendering environment jumps to this point in time to continue the presentation's display. If the hyperlink refers to an external resource such as another personalized multimedia presentation, the rendering of the current presentation is stopped and the referred resource is displayed (either as replacement of the current presentation or in a new window). In order to reflect the users' or user groups' profile information and context information, an external personalized multimedia presentation is typically created on the fly. For it, the hyperlink provides all the information needed in order to identify the user or user group and to create the content accordingly.

Besides displaying the presentation directly after the Transform process, the presentations can also be first delivered to another system. Such a system could be an Internet search engine that gathers and indexes semantically-rich presentations to provide its users for a better search and retrieval of such content. Finally, these systems allow the users to access and consume the presentations.

# 4. SemanticMM4U FRAMEWORK

So far, we have introduced the general creation chain for personalized semantically-rich multimedia presentations in Section 2. In Section 3, we described in detail the canonical processes supported by this chain. Considering the creation chain and the processes involved, one can say that neither research approaches nor systems we find in industry today provide a generic and at the same time practical support for dynamically creating personalized semantically-rich multimedia presentations. In addition, todays approaches merely exploit semantically-rich information and derive new metadata in order to create the multimedia content. Once the content is created, this highly valuable source of information is thrown away and not further used.

In order to provide a generic and at the same time practical support for creating personalized semantically-rich multimedia presentations, we pursue with the SemanticMM4U framework a software engineering approach. The overall goal of the SemanticMM4U framework is to provide application developers support for an easier creation of personalized multimedia content and allowing them to enrich this content with semantically-rich metadata. The SemanticMM4U framework provides generic implementations of the phases of the general creation chain. It integrates previous work for authoring personalized multimedia content [27, 30, 31, 29, 28] with deriving semantically-rich metadata while creating the content. The metadata are made available for further use by integrating it into the final presentations.

The SemanticMM4U framework is aimed at embracing and integrating different research in the fields of multimedia content personalization and metadata derivation rather than re-inventing personalized semantically-rich multimedia content creation. The component-based architecture of this framework is introduced in Section 4.1. In Section 4.2, we present the relationship of the presented canonical processes to the components of the SemanticMM4U framework. In Section 4.3, we show how the SemanticMM4U framework can be used by other applications. Finally, we argue for why defining canonical processes helps application developers in utilizing the framework.

## 4.1 Architecture

The architecture of the SemanticMM4U framework is depicted in Figure 7 as UML component diagram. It shows the six components of the framework and their central interfaces. The User Profile Accessor and Media Pool Accessor components provide access to a unification of user profile and context information (so-called unified user profile [27]) and to media assets. The central interface of the User Profile Accessor component is IUserProfile providing the user profile and context information. The Media Pool Accessor component has two central interfaces, IMedium as well as IMediaList providing media assets and their metadata. The user profile information, context information, and media metadata are exploited for the multimedia composition task in the Multimedia Composition component. Within this component, the media assets selected by the Media Pool Accessor component are organized in time, space, and interaction into the personalized multimedia content. The content is represented in a tree-based, object-oriented model. The nodes of the multimedia content tree are of type IVariable. They are passed to the Metadata Derivation component, which derives further metadata about it. Once the personalized semantically-rich multimedia content is created, its object-oriented representation is passed to the Presentation Format Generators component. Here, the content and its metadata are transformed into the concrete multimedia presentation formats. Finally, the semantically-rich multimedia presentation in its final presentation format is passed to the Multimedia Presentation component, using the interface IMultimediaPresentation. This component actually renders and displays the presentation on the end device of the user.

## 4.2 Mapping the Processes to Components

In the previous section, we have presented the component-based architecture of the SemanticMM4U framework. Here, we map the processes presented in Section 3 to the components of this framework. The Select process is realized by the components User Profile Accessor and Media Pool Accessor. The User Profile Accessor component merely provides access to the users' or user groups' profile information and context information. This information is passed to the Media Pool Accessor component. This component actually conducts the query and creates media assets on the fly if necessary. The Assemble Organize process is implemented by the Multimedia Composition component. It takes the media assets from the Media Pool Accessor component and organizes them into multimedia content in the internal rep-

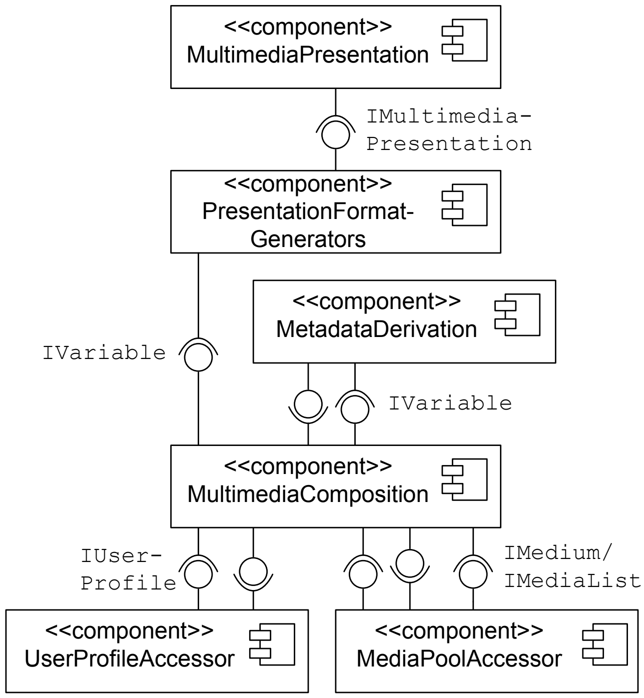


**Figure 7: UML component architecture of the SemanticMM4U framework**

resentation model. The Metadata Derivation component provides the metadata derivation of the Assemble Annotate process. It works on multimedia content provided by the Multimedia Composition component. The implementation of the Transform process is realized by the Presentation Format Generators component. It takes the multimedia content in the representation model and transforms it to concrete presentation formats. Finally, the presentations in the concrete formats are passed to the Multimedia Presentation component, which implements the Present process.

## 4.3 Using the Framework

The SemanticMM4U framework can be used in arbitrary applications that need to provide personalized multimedia content to their users, publish and distribute it in different presentation formats for different (mobile) end devices, and annotating it with semantically-rich metadata. The SemanticMM4U framework and thus the presented processes have already been used for the development of many applications in different domains. For example, we developed a personalized multimedia sports news ticker [27], a context-driven smart authoring tool [32, 29], and a generic personalized city guide [28]. These applications use some or all processes defined in our general creation chain. They leverage the processes in different configurations, i. e., they use different implementations of the framework components. These and further applications are described in detail in [27].

The SemanticMM4U framework is also used by other research groups. For example, the multimedia database METIS [21] developed at the Research Studios Austria uses the SemanticMM4U framework implementation of the Transform process to provide dynamically created multimedia content in different multimedia presentation formats. For it, the Transform process is provided as web service [27, Sec. 10.4] to the researchers. This web service has been made available to the general public. Within the CoCoMA project [10], which is part of the DELOS network of excellence [11] funded by the European Commission, the processes Assemble Organize, Transform, and Present are used. Other possible use of our framework is employing the Select process solitary. Here, an application can use the corresponding components storing media data as well as user profile information and context information and providing media retrieval based on this information.

## 4.4 Benefits of Identifying the Processes

Identifying the canonical processes for creating personalized semantically-rich multimedia content provides in general for a better understanding of the creation chain of such content. Mapping the processes to the components of the SemanticMM4U framework in Section 4.2 supports researchers and industry in using the framework. Users typically describe their approach or system in terms of processes. Having such a description, one can identify whether the required functionality is provided by one or more of our framework components. Once the processes for a specific application are identified, the framework components needed to implement the application can be directly derived.

# 5. CONCLUSIONS

In this paper, we have presented the general creation chain for personalized semantically-rich multimedia presentations. To provide generic and at the same time practical support for this creation chain, we pursue a software engineering approach with the SemanticMM4U component framework. We have presented the canonical processes provided by the creation chain and described how these processes have been implemented in the SemanticMM4U framework. The framework provides application developers a software engineering support for creating personalized semantically-rich multimedia presentations. By the nature of software frameworks, the components of the SemanticMM4U framework can be extended and adapted in regard of application specific requirements. Thus, employing component technology allows for an easy exchange and replacement of the framework's functionality by different implementations. Considering with our SemanticMM4U component framework the entire creation chain for personalized semantically-rich multimedia presentations, we strongly believe that the definition and design of the presented processes are a helpful abstraction and make the processes exchangeable and reusable. They will help different research groups as well as industry to work together on personalized content creation and metadata derivation.

*Acknowledgments.* This research was supported by a Marie Curie International Fellowship within the 6th European Community Framework Programme.

# 6. REFERENCES

[1] Adobe, Inc., USA. Flash File Format Spec., 2007. http://www.adobe.com/licensing/developer/.

[2] Amazon, Inc, USA. amazon.com, 2007. http://www.amazon.com/.

[3] F. Bes, M. Jourdan, and F. Khantache. A Generic Architecture for Automated Construction of Multimedia Presentations. In *Multimedia Modeling*. IEEE, 2001.

[4] S. Boll and W. Klas. ZYX - A Multimedia Document Model for Reuse and Adaptation. *IEEE Trans. on Knowledge and Data Engineering*, 13(3):361–382, 2001.

[5] S. Boll, W. Klas, and U. Westermann. Multimedia Document Formats - Sealed Fate or Setting Out for New Shores? *IEEE Int. Conf. on Multimedia Computing and Systems*, pages 604–610, June 1999.

[6] S. Boll, P. Sandhaus, A. Scherp, and U. Westermann. Semantics, content, and structure of many for the creation of personal photo albums. In *Multimedia*, pages 641–650. ACM, 2007.

[7] M. Bordegoni, G. Faconti, S. Feiner, M. T. Maybury, T. Rist, S. Ruggieri, P. Trahanias, and M. Wilson. A standard reference model for intelligent multimedia presentation systems. *Computer Standards & Interfaces*, 18(6-7):477–496, Dec. 1997.

[8] P. Brusilovsky and M. T. Maybury. From adaptive hypermedia to the adaptive Web. *Communications of the ACM*, 45(5):30–33, 2002.

[9] D. C. A. Bulterman and L. Hardman. Structured multimedia authoring. *ACM TOMCCAP*, 1(1):89–109, 2005.

[10] S. Christodoulakis, C. Tsinaraki, C. Breiteneder, et al. CoCoMA. In *Research and Advanced Technology for Digital Libraries*, page 33 ff., 2005.

[11] DELOS NoE. DELOS Network of Excellence on Digital Libraries, 2007. http://www.delos.info/.

[12] A. K. Dey and G. D. Abowd. Towards a Better Understanding of Context and Context-Awareness. Technical report, Georgia Institute, USA, 1999.

[13] G. P. Faconti and T. Rist, editors. *Workshop Towards a Standard Reference Model for Intelligent Multimedia Presentation Systems*, 1996.

[14] K. Falkovych and F. Nack. Context Aware Guidance for Multimedia Authoring: Harmonizing domain and discourse knowledge. *Multimedia Systems J.*, 3(11):226–235, Mar. 2006.

[15] K. Falkovych, F. Nack, J. van Ossenbruggen, and L. Rutledge. Sample: Towards a framework for system-supported multimedia authoring. In *Multimedia Modelling*, page 362 ff. IEEE, 2004.

[16] J. Fink, A. Kobsa, and J. Schreck. Personalized hypermedia information through adaptive and adaptable system features: User modeling, privacy and security issues. In *Intelligence in Services and Networks*, pages 459–467. Springer, 1997.

[17] J. Geurts, J. van Ossenbruggen, and L. Hardman. Application-specific constraints for multimedia presentation generation. In *Multimedia Modeling*, pages 247–266. IEEE, 2001.

[18] R. Hjelsvold, S. Vdaygiri, and Y. Léauté. Web-based personalization and management of interactive video. In *World Wide Web*, pages 129–139, 2001.

[19] IBM Corporation, USA. QBIC Home Page, 2004. http://wwwqbic.almaden.ibm.com/.

[20] M. Jourdan and F. Bes. A new step towards multimedia documents generation. In *Int. Conf. on Media Futures*, pages 25–28, May 2001.

[21] R. King, N. Popitsch, and U. Westermann. METIS: a flexible database foundation for unified media management. In *Multimedia*, pages 744–745. ACM, 2004.

[22] A. Kobsa, J. Koenemann, and W. Pohl. Personalized Hypermedia Presentation Techniques for Improving Online Customer Relationships. In *The Knowledge Engineering Review*, pages 111–155. Cambridge University, 2001.

[23] T. Lemlouma and N. Layaïda. Adapted Content Delivery for Different Contexts. In *Symposium on Applications and the Internet*, pages 190–197. IEEE, Jan. 2003.

[24] T. Lemlouma and N. Layaïda. Context-Aware Adaptation for Mobile Devices. In *Int. Conf. on Mobile Data Management*, pages 106–111. IEEE, 2004.

[25] Oracle Corporation, USA. interMedia, 2004. http://www.oracle.com/technology/products/intermedia/index.html.

[26] S. Santini, A. Gupta, and R. Jain. Emergent semantics through interaction in image databases. *IEEE Transactions on Knowledge and Data Engineering*, 13(3):337–351, 2001.

[27] A. Scherp. *A Component Framework for Personalized Multimedia Applications*. OlWIR, Oldenburg, Germany, Feb. 2007. PhD Thesis, http://ansgarscherp.net/dissertation/.

[28] A. Scherp and S. Boll. Generic Support for Personalized Mobile Multimedia Tourist Applications. In *Multimedia*, pages 178–179. ACM, 2004.

[29] A. Scherp and S. Boll. Context-driven smart authoring of multimedia content with xSMART. In *Multimedia*, pages 802–803. ACM, 2005.

[30] A. Scherp and S. Boll. MM4U - A framework for creating personalized multimedia content. In *Managing Multimedia Semantics*, pages 246–287. Idea Group, 2005.

[31] A. Scherp and S. Boll. Paving the Last Mile for Multi-Channel Multimedia Presentation Generation. In *Multimedia Modeling*, pages 190–197. IEEE, 2005.

[32] A. Scherp, S. Boll, and H. Cremer. Emergent semantics in personalized multimedia content. *J. of Digital Information Management*, 5(2), Apr. 2007.

[33] A. Scherp and R. Jain. Towards an ecosystem for semantics. In *Workshop on the many faces of multimedia semantics*, pages 3–12. ACM, 2007.

[34] B. Schilit, N. Adams, and R. Want. Context-Aware Computing Applications. In *Workshop on Mobil Computing Systems and Applications*, pages 85–90. IEEE, 1994.

[35] A. Schmidt, M. Beigl, and H.-W. Gellersen. There is more to context than location. *Computers & Graphics*, 23(6):893–901, 1999.

[36] J. van Ossenbruggen, L. Hardman, J. Geurts, and L. Rutledge. Towards a multimedia formatting vocabulary. In *World Wide Web*, pages 384–393. ACM, 2003.

[37] W3C. RDF/XML Syntax Specification, Feb. 2003. http://www.w3.org/TR/rdf-syntax-grammar/.

[38] W3C. Extensible Markup Language, Feb. 2004. http://www.w3.org/TR/2004/REC-xml-20040204/.

[39] W3C. Mobile SVG Profiles, Mar. 2004. http://www.w3.org/TR/SVGMobile12/.

[40] W3C. Scalable Vector Graphics, 2005. http://www.w3.org/TR/SVG12/.
[41] W3C. Synchronized Multimedia Integration Language, 2005. http://www.w3.org/TR/SMIL2/.