Supporting Collaborative Workflows of Digital Multimedia Annotation

Cristian Hofmann and Dieter W. Fellner

Abstract Collaborative annotation techniques for digital multimedia contents have found their way into a vast amount of areas of daily use as well as professional fields. Attendant research has issued a large number of research projects that can be assigned to different specific subareas of annotation. These projects focus on one or only few aspects of digital annotation. However, the whole annotation process as a operative unit has not sufficiently been taken into consideration, especially for the case of collaborative settings. In order to attend to that lack of research, we present a framework that supports multiple collaborative workflows related to digital multimedia annotation. In that context, we introduce an process-based architecture model, a formalized specification of collaborative annotation processes, and a concept for personalized workflow visualization and user assistance.

Introduction

As means of enriching digital content by additional information, collaborative annotation techniques for digital multimedia contents have found their way into a vast amount of areas of daily use as well as professional fields. In the Web, popular platforms can be identified such as *Flickr.com*, *Youtube.com*, or *Delicious.com*, which enable shared manual classification and bookmarking of contents of various media formats in order to organize such files in a structured form and facilitate later retrieval. In addition to these simple ways of applying annotations, more complex environments are utilized for various purposes and objectives, such as information retrieval, content analysis, or group communication, and in the scope of different

C. Hofmann (✉) and D.W. Fellner
Fraunhofer Institute for Computer Graphics Research/TU Darmstadt,
Interactive Graphics Systems Group, Fraunhoferstr. 5, 64283 Darmstadt, Germany
e-mail: cristian.hofmann@igd.fraunhofer.de; d.fellner@igd.fraunhofer.de

M. Lewkowicz et al. (eds.), *Proceedings of COOP 2010*, Computer Supported Cooperative Work, DOI 10.1007/978-1-84996-211-7_6,

application areas and practices, e.g., computer-supported collaborative learning, medicines, engineering, human science, sports science, e-Commerce, edutainment, or gaming [4, 30, 31, 33]. In our research work we focus on two concrete scenarios of collaborative multimedia annotation. The first case deals with annotation of multimedia content with semantic information. At this, users first annotate multimedia documents (e.g., websites, videos, images, etc.) or parts of a document (e.g., single video scenes, image sections, etc.) with simple metadata such as descriptors or textual tags. Subsequently, documents or document parts are assigned to a given ontology. The second annotation scenario is concerned with video content analysis. In that case, users collaboratively select video segments by performing shot detection, write transcriptions, categorize video segments using a predefined vocabulary, enter own ratings and interpretations of the observed content, and discuss their results. In the following, the differing variants of digital multimedia annotation are going to be illuminated.

Digital Multimedia Annotation: A General Overview

In order to comprehend the background of this work, it is important to briefly overview the wide range of the area of digital annotation concerning its various forms and usages. In a general view, annotations are means of enriching arbitrary content by additional information. Thus, from a user's point of view, annotations enable persons to associate own information with provided documents [2]. That personal information can manifest different types and purposes such as commentary, elucidation, explanation, footnote, interpretation, reflection, linking, classification, or linking [1, 26, 30]. In the scope of such usages, annotations may appear as *metadata*, *content*, or *dialog act*. Annotations as metadata serve as additional data associated to existing contents, defining their properties and structure [1, 2]. Annotations also can be applied as content, enriching existing content by a further layer of explanation or interpretation [1, 2, 26]. As part of a collaborative discourse respecting a certain subject, content-annotations can be regarded as dialog acts [1, 14]. Different media formats such as text [36], images [40, 41], audio files [5], video [12, 20, 40], or 3D-models [19, 23] are involved in digital annotation referring to the contents that are annotated on the one hand and the coding of annotations on the other hand [2, 13, 32]. In general, annotation techniques aim the exposure and elucidation of the semantics of given information [2]. In line with that superior objective, annotation environments are used in various application areas such as indexing and retrieval, information classification, content analysis, sharing and discussion, or collaborative authoring. The latter forms of usage reveal that annotations also can be means of exchanging and sharing ideas and opinions of a collaborating community with respect to mutual accessible digital contents [2, 10, 11, 14, 43]. In that context, different types of annotations are identified: annotations written by the producer of the content, and annotations created by the recipients of the provided contents [1]. Furthermore, annotations can be private, shared, or public [27]. Private annotations are visible and accessible only to their authors, whereas shared and public annotations are treated by a certain team or all participating users. In cases of collaborative working, user groups obtain multiple views of the handled information [27]. Thus, annotations support the discourse and information exchange within specific communities improving various forms of collaborative work scenarios [2, 11, 13]. In conclusion, the area of (collaborative) digital annotation includes a large set of different workflows that embrace several research topics.

Support of Collaborative Annotation Workflows: Key Challenges and Our Contribution

Beginning from the times of manual annotation of textual documents, research on annotation has long tradition and issued a large number of research projects that can be assigned to different specific subareas of annotation. These projects focus on one or only few aspects of digital annotation [20, 21]. However, the annotation process, as a whole operative unit which issues several different workflows, has not sufficiently been taken into consideration, especially in cases of collaborative work [1, 17, 20, 21]. The exploration of workflows of collaborative multimedia annotation is a relevant research topic due to user-specific requirements and challenges of application design. First, from a *user's point of view*, problems can be identified with respect to the usage of annotations systems. To take an example, within the process of collaborative video analysis, an amount of tasks or sub-processes can be registered: configuration of specific application and community settings, marking and chunking contents, classification of selected contents, generate transcriptions, writing (shared) interpretations and ratings with respect to the observed facts, searching and commenting of co-analysts results, leading discussions about disagreements, subsequently re-editing of configurations and own contributions, and publishing of results [17, 21, 29, 30, 35]. Different annotation tools, algorithms, etc. can be used to accomplish that large number of analytic (annotation) tasks. Commonly, the amount of available services leads to multi-optional and complex user interfaces. Thus, users struggle in understanding and learning how to use the system's set of tools and have problems in recognizing their current state and the next steps within the process [37]. In that context, annotators have to make an effort in recognizing contents that are to be executed by them and not by other community members. Furthermore, annotation can be an elaborate and time-consuming work, especially when it is conducted in a "manual" way [3]. Second, from an *application design point of view*, relevant key features have been declared with respect to flexibility in order to support different ways of using annotations and integration of different services [1, 11].

In order to attend to the lack of research on the annotation process as (collaborative) workflows, and based on the described challenges and problems which arise from a user's and an application design points of view, the main contribution of this paper is the illustration of a generic framework to support collaborative workflows related to digital multimedia annotation. By workflow-support, we mean the facilitation of loops and transitions between the single workflow steps and tasks in general. More precisely, appropriate services and information shall be provided at the proper time, depending on the current individual or group's state of the work. Consequently, we expect a reduction of the users' load with regard to the use of such applications and hence enhancement of efficiency. In addition to that, we enable the integration of different annotation services by applying a service-oriented infrastructure. Thus, previously declared demand on flexibility can be complied. The annotation framework has been developed in the scope of the THESEUS program. THESEUS is a research program, initiated by the Federal Ministry of Economy and Technology (BMWi), to develop a new internet-based infrastructure in order to better use and utilize the knowledge available on the internet.

In Section Related Work, the identified related work concerning research on support of annotation workflows and practices is illuminated. Section Process of Collaborative Multimedia Annotation gives a brief overview of an abstract model describing annotation processes and tasks. Following, an integrable generic framework for workflow-based collaborative annotation is presented in Section A Framework to Support Collaborative Annotation Workflows. In that context, we illustrate an architecture model that is based on a process model for digital annotation, as well as a concept for workflow visualization and user assistance by interpreting process specifications. Concluding, we summarize the presented results and assume our future work.

Related Work

Bertino, Trombetta, and Montesi present a framework and a modular architecture for interactive video consisting of various information systems. The coordination of these components is realized by identifying inter-task dependencies with interactive rules, based on a workflow management system [4]. The Digital Video Album (DVA) system is an integration of different subsystems that refer to specific aspects of video processing and retrieval. In particular, the workflow for semiautomatic indexing and annotation is focused [31]. Pea and Hoffert illustrate a basic idea of the video research workflow in the learning sciences [22]. In contrast to our research work, the projects mentioned above do not or only to some degree consider the process for collaborative use cases. The reconsideration of such communicative and collaborative aspects requires modifications and enhancements of the existing approaches and concepts.

Processes of Collaborative Multimedia Annotation

In this section, workflows of digital annotation are illustrated in a general view by means of an abstract process model. That model arises from an empirical study of multiple identified annotation workflows of real practices, primarily focusing on

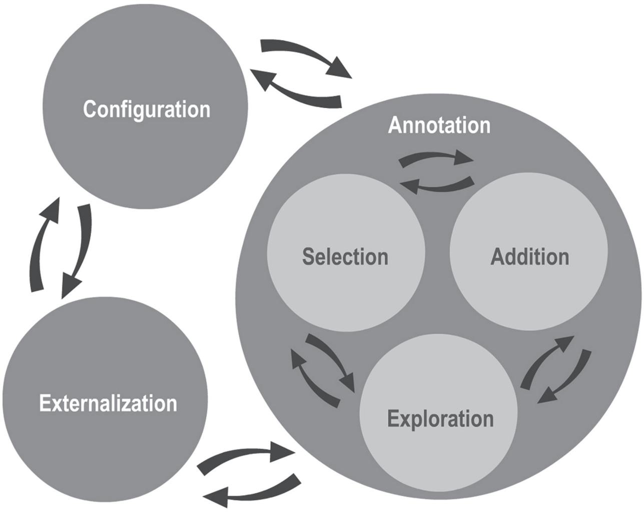

Fig. 1 Abstract process model for digital multimedia annotation

collaborative usage settings [21]. As shown in Fig. 1, the annotation process is divided into the sub-processes *configuration*, *annotation*, and *externalization*. Annotation is subdivided again into the sub-processes *selection*, *addition*, and *exploration*. These sub-processes consist of one or more different tasks that have to be accomplished by certain users or groups applying differing tools and algorithms. Although these sub-processes are subsequent in a first sight, this section shows that annotation workflows have to be regarded as networks including options, iterations, loops and re-entries to previous steps. In the following, the sub-processes of digital multimedia annotation are illustrated as tasks of the annotation process.

Configuration

Before starting an annotation project, the environment has to be configured. Specific project preferences can be adjusted and the graphical user interface may be customized [9]. Furthermore, participants are assigned to accounts and user groups that are associated with specific roles and access rights. The annotation tasks can be distributed among the individual users [9, 25, 40]. For example, tasks of a project are assigned to predefined groups, and furthermore a group administrator is able to distribute the annotation tasks among the individual users [25, 40]. Since annotation processes are iterative and contain loops and re-entries to previous process states [30, 38], predefined configurations of the used environment often need to be modified during a running annotation process.

Annotation

The following phases selection, addition, and exploration can be regarded as one operative unit. The conducted expert interviews revealed that no specific operative sequences can be identified referring to these "real" acts of annotating [30]. Note on video analysis processes that activities of de-composition (segmentation, coding, categorization, and transcription) and re-composition (rating, interpretation, reflection, comparison, and collocation) are closely interrelated. They depict video annotation as a complex process that contains circular and recursive loops, in which the analyst alternately marks, transcribes and categorizes, analyzes and reflects, and needs to conduct searches [28]. Performed a field study with respect of the annotation of digital libraries, reporting that acts of searching, reading and annotating are performed at the same time and can be done together with other activities, e.g. working with colleagues. Hence, selection, addition, and exploration, as higher-level categories, have to be considered related to each other. In the following, these three items are explicitly described.

Selection

Annotators need to mark concrete contents of interest that annotations shall refer to. The simplest variant is marking a whole document such as a website. A second variant is the selection of elements contained within a document. Let us assume that a considered website consists of a text, various graphics, and a video. An annotator is able to mark each of these elements as an independent unit, or select two or more of these elements as a composed unit, and consequently annotate a unit with different information [13, 32]. As a third variant, these single elements can be again subdivided into segments. For example, annotations may be associated to a whole text, or to one or more special parts of the text [11]. How exactly segments are defined depends on the media format of the original content and the specific media properties, as well as the purpose of annotation. Segments can also be artifacts of collaborative work. In some of the investigated use cases, the segmentation task is partitioned and assigned to individual users or groups. For example, group A chunks the video according to a certain characteristic 1, group B seeks for characteristic 2, and so on. Furthermore, in content analysis use cases, users or groups may also work with different classification systems, vocabularies, ontologies, etc.

Addition

After marking the relevant documents, document subsets, or object segments, co-annotators continue adding annotations as additional information to these elements. As already mentioned, a relevant objective of annotation is the classification of given information resources or parts of it, in order to facilitate archiving and later retrieval [3]. Collaborative practices are supported by allowing shared information organization activities and exploiting of information structures establish by collaborators. For example, widespread approaches are *Collaborative Tagging*, *Social Bookmarking*, and *Semantic Annotation* [3, 10, 26]. In a further context, users may describe observed facts, e.g. behaviors and events of a video, objects within an image, sequences of an audio file, etc. In order to explain, elucidate, interpret, or comment on the given contents, users need to give descriptions in a more free way than assigning metadata [35]. In most cases, they are allowed to enter free textual annotations that can be regarded as additional content [1, 2, 26]. Like the selection task, the addition of annotations may be distributed to different users and groups. In that case, annotated information becomes a shared contribution [2, 11, 14, 43]. Shared contributions are an important part of collaborative work. They enable the communication between co-annotators as well as the organization of common tasks and are regarded as dialog acts [1, 14]. Organization and coordination of collaborative activities are essential in the context of co-writing or co-authoring with respect to granulated information exchanges between collaborators creating a shared document [10, 43]. Furthermore, when users work separately, they need to discuss their annotations, conclusions, and the annotation process with other participants [9]. Particularly in the context of consensual approaches applied in content analysis, discussion is a means of agreement and consistency of different annotators' results. Thus, discussion often leads to a return to previous steps of the annotation process [29, 30, 35]. In the end, the final results of the annotation project arise from iterative loops through the process, in which the data is continually modified and adjusted.

Exploration

Selection and information addition always go along with searching, browsing, and reception activities [28, 30]. Especially in collaborative annotation situations, users also need to search for their own results but also for results of co-annotators, experts, or other sources [22, 30]. Exploration of co-annotator's data also can be an issue in asynchronous collaborative projects which proceed over a long time frame. After being absent, users may need to track the changes performed by other annotators involved in the project. Exploration also includes restructuring of the data representation. With regard to this, annotators are allowed to contrast relevant data with each other, or to hide less important information. According to this, exploration also supports reflection. It facilitates the consideration of multiple views of the video where users are allowed to obtain perspectives beyond their subjective point of view [27, 38, 40].

Externalization

The externalization phase refers to two different aspects at the end of the annotation process: Publication of the process' results and export of data for the purpose of processing the data with external applications. Public annotations are treated by all participating users [2, 26]. Normally, publication begins with editing and converting the data into several formats, and moves on to presenting this information by means of corresponding media [30]. Published results can be used for demonstration purposes [29]. Also databases of already annotated material can serve as digital resource for information retrieval in following annotation sessions [35]. Furthermore, a goal of annotation can be obtaining (mostly automatically) generated surveys and assemblies of similarly categorized content subsets, e.g. video summaries [3]. Furthermore, it is often necessary to export data for further processing by means of more specific applications.

A Framework to Support Collaborative Annotation Workflows

In this section, a generic and integrable framework that supports different concrete annotation workflows is presented. First, general requirements with respect to workflow-support and collaboration are derived. Subsequently, we present a developed the architecture model that is based on the sub-processes of collaborative annotation illustrated in the previous section. In particular, it is shown that the architecture aims at establishing a framework that can be integrated in superior systems. Furthermore, we propose a formal specification of processes that is used for the purpose of workflow control and visualization, as well as a process-driven user assistance. Workflow visualization and process-driven assistance are presented as concepts in the last part of this section.

Process-Based Requirements

Conceptual design decisions particularly base on our endeavor to support given operating procedures within the annotation workflow. In that context, we identified basic requirements that have to be fulfilled by the framework. In general, the following requirements result from our explicit consideration of annotation workflows on the one hand, and collaborative practices on the other hand. The requirements are based on a comparative analysis of current annotation environments.

1. *Workflow Control.* Workflows of collaborative multimedia annotation can be complex, since they are network-like and may contain several iterations. Thus, transitions between workflow phases and the control of sequences of sub operations have to be supported. Also loops and re-entries to other phases of the workflow must be considered.
2. *Sub-Process Enclosure.* In order to guarantee workflow control, sub-processes of annotation must be made "tangible" for a certain control instance. Thus, the identified phases and tasks of collaborative annotation workflows need to be pooled into functional units that are mutually delimited. In doing so, task areas can be typecasted and invoked by addressing respective modules.

3. *Process Awareness*. In order to support operating flows of annotation, users must be enabled to monitor the current process. That is, he or she needs to obtain information about the current state as well as actual and upcoming activities or tasks. Thus, users are “aware of process”. For example, Process Aware Systems use work lists to offer work items that have to be performed by people [24]. In that context, especially in the case of collaborative work, that kind of visualization needs to be personalized due to the different needs of varying user groups and roles [7, 8].
4. *Service Integration and Providing*. As mentioned in that paper’s introduction, a large number of potential functionalities can lead to complex and multi-optional user interfaces that obstruct especially unexperienced users. Furthermore, Agosti et al. (2007) and Constantopoulos et al. (2004) note that flexibility in order to support different ways of annotation as well as integration of different services are key aspects of the design of annotation systems [1, 11]. As a result, integration of different annotation services and providing these services to users in an appropriate way have to be supported.
5. *Extensibility*. Flexibility and service integration require a framework that provides interfaces or “docking ports” for multiple tools or external services. In addition to that, the system must enable administrators to integrate, replace and remove tools that can be assigned to task-related modules.
6. *Consistency*. As a consequence of extensibility, since multiple tools read and possibly write on the same data. Thus, the consistency of shared parts of the data set has to be ensured at every point of the annotation process. Furthermore, with respect to collaborative practices in which data is exchanged between members of a community, consistency of data must also be warranted for every peer in the shared system.
7. *Selection Support*. It is essential to select the concrete contents that are to be annotated. In doing so, there are two levels of different kinds of contents. First, a whole multimedia document, one or more media elements contained in a document, or segments of single media elements can be marked. Second, information may be coded in different media formats [2, 11, 13, 32]. In order to support the annotation workflow, selection of the desired contents must be alleviated.
8. *User Management*. User management is required due to the following reasons: Users have to be assigned to accounts, user groups, and roles with respective access rights. In collaborative settings, the annotation task may be portioned and distributed to users and user groups in several ways. As a consequence, a personalized view on the workflow for varying roles of users must be provided [7, 8].
9. *Data Exchange*. In collaborative use cases, the annotation environment has to realize the data exchange between multiple users of the application that are potentially separated over space. For this purpose, stored information must be made available to every participant of the group.

10. *Correct Data Handling*. In the scope of collaborative work, in which each group member has full access to shared contents, Stahl et al. regard the emerged information set as a *Dynamic Information Space*. That is, the information set is continually change and extended [38]. In addition to that, annotations can be assigned to different kinds of contents. Thus, an appropriate handling of media files, its annotated information, as well as their organizational structure must be provided.

Process-Based Architecture Model

In this section, the general assembly of the framework's underlying architecture are presented. In that context, components of the architecture that are related to the sub-processes of annotation are illustrated.

The architecture model of the collaborative annotation framework was drafted relying on the presented requirements on the one hand, and on the identified sub-processes and tasks of collaborative annotation on the other hand. As a result, the key features of the concept are (a) a *Client-Server-Architecture*, (b) a *Component-based Environment*, (c) the realization of an *Model-View-Controller* (MVC) and the *Mediator* design patterns, (d) an *Event-based Communication*, and (e) a *Service-oriented Infrastructure*. The single features of the architecture are illustrated in the following (Fig. 2).

Client-Server Model

A fundamental condition for collaborative processes is the interconnectedness of every peer taking part for information exchanging purposes [9, 13]. Within a range

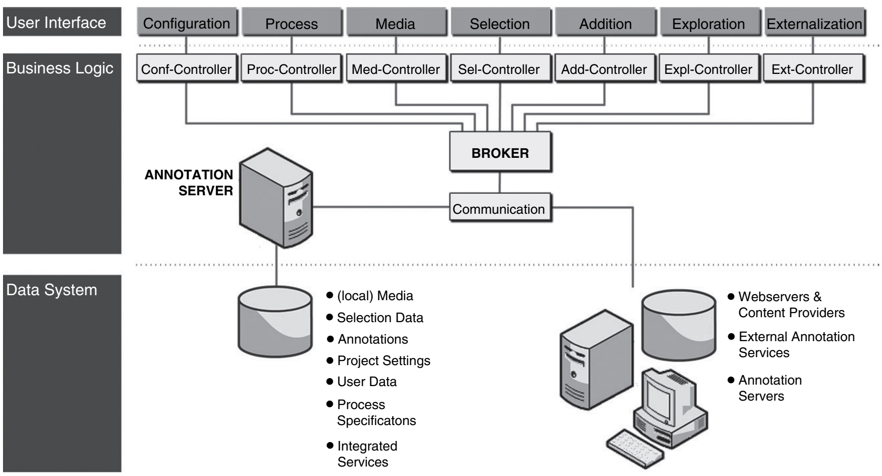

Fig. 2 Process-based architecture model

of optional models that can be considered, e.g., client-server, peer-to-peer, or web-based approaches. We suggest a client-server architecture, not only due to its wide spreading in the area of information systems [9]. In general, data consistency, shared data exchange, and user management are enabled. A centralized *Multimedia Annotation Server* realizes a centralization of the information space. In doing so, the data system is available for any client that is connected. Additionally, it provides several services such as authentication of annotators, and management of accounts and access rights. In that context, modifications of the data set by the community are managed and provided to respective group members. The server is also responsible for a consistent storage and management of local media files, global project configurations, and annotation information, but also workflow-related data like predefined process specification files or integrated services. Centralized management also provides support for distributed authoring processes, e.g. tracking of changes made by co-annotators. With respect to our specific research focus, a relevant function of the server application is to provide predefined workflow specifications. Suchlike description files can be accessed by the local Workflow Management Environment in order to perform controlling activities. For example, one prototype of our annotation framework transmits workflow descriptions coded in *XPDL* (XML Process Definition Language) via HTTP. The client application handles user entries and interaction on the graphical user interface. It provides authoring options and assigned tools for media and annotation editing purposes. Here, the views of the MVC model are pooled as graphical representation of services.

Component-Based Environment

The identified sub-processes of annotation need to be pooled into functional units that are mutually delimited in order to typecast task areas that can be invoked by a central control unit. Furthermore, the architecture must enable administrators to integrate, replace and remove tools that can be assigned to these task-related modules. Thus, the phases of the workflow are implemented as software components. A software component can be seen as an enclosed unit which provides specific services. It can be embedded into a higher-level system and combined with other components. The concrete implementation of a component is concealed from its accessing instance, the communication is provided by specific interfaces [29]. Within our architecture model, the components are abstract and serve as containers for previously assigned tools and methods. Furthermore, they may be implemented several times. To sum up, sub-process enclosure, service integration, and extensibility are guaranteed. Central components are the *Broker Component*, *Process*, *Media*, and the *Communication Component*. The Broker implements the Mediator component as global controller of the MVC model. The Process Component is responsible for process visualization and process-driven user assistance which are going to be explained in the next section. The Media Component serves as display component for selected media files. The Communication component realizes the connection between elements of the shared framework. First, the communication between the framework's central Multimedia Annotation Server is establish. Additionally, access is warranted to external information resources such as web servers, external annotation services, or external annotation servers, e.g., providing and storing semantic annotations or ontologies. Based on the identified process phases and tasks, we conceived and included the abstract components *Configuration*, *Selection*, *Addition*, *Exploration*, and *Externalization*. The Configuration Component is responsible for any administrative task performed by authorized users. It provides input interfaces for several configuration options such as users and task management, process specification, or project settings. Selection components represent any service for content selection or segmentation. For that purpose, interfaces must be provided that support interaction with media players and data visualization components. Analogously, addition components are responsible for the conjunction of any kind of annotation. Exploration components serve as means of reception, browsing, navigation, and searching. Basically, information is visualized, and services for manipulation of the data representation such as filtering or sorting are provided. In addition to that, specific search functionalities may be integrated. Examples are specific timeline views for continuous media like audio or video, or annotation structure visualization tools. Externalization tools are responsible for data processing for the use in external environments.

Model View Controller and Mediator Pattern

In order to realize an appropriate management of media files and its annotated information, we rely on existing approaches with regard to content annotation or linking. In the area of hypermedia research, several models can be identified, e.g. the *Dexter Hypertext Reference Model*. That model divides the system into three delimited layers, separating data, the given hyper structure, and its representation [18]. The *Model View Controller Model* (MVC) equally divides the application into three levels: the model layer represents the involved data, views display the information and assume user interaction, the controller layer processes user entries and is enabled to modify data in the model. Furthermore, data consistency is warranted through a specific notification policy. In the presented process-based architecture, the model layer consists of data and information from contents and assigned annotations to project configuration information, etc. In that context, the storage of predefinable process specifications and (references to) the available services are a central part of our work. The view layer represents any visual component at the graphical user interface. Besides the general elements of the user interface, the single views display the available tools or services that are previously assigned to respective tasks of the annotation workflow. The controller layer includes two different kinds of controllers: local controllers are assigned to every component of the view (as well as the server communication component) and act as interfaces between component and application. A central component serves as global controller and implements the included Mediator pattern. The Mediator pattern provides a central instance, called *Broker Component*, which defines the cooperation and interaction of multiple objects. This central unit holds an intermediary role and coordinates the overall behavior of the system [15]. Thus, workflow control can be supported with regard to transitions between workflow phases, sequences of sub operations, passing through loops, and re-entries to other phases of the workflow. The specific processes and sequences within the annotation workflow are defined by task groups and sub operations, which can be pooled into several system components. The global controller is key component with respect to the realization of *Event-based Communication* and the establishment of a *Service-oriented Infrastructure*. A event-based system is a shared environment composed of multiple active software components and an service that realizes event transmission, commonly called Event-Broker or Event-Dispatcher [34]. There, active software components may obtain the role of an event publisher or an event consumer. The Broker component appears as event transmitter and is responsible for inter-service communication and interoperability, as well as workflow control and governing. Thus, a service-oriented infrastructure can be implemented. Service-oriented infrastructures focus on the realization of components as modular services that can be invoked by certain clients. Thus, based on a service-oriented infrastructure that is enabled by the global controller, modularized sub-processes of collaborative annotation can be individually invoked and controlled.

Process Management I: Specifying Processes

The main objective of our work is to support different workflows by dealing with user-specific problems that arise from multi-optional tool sets and complex user interfaces, and design-specific requirements. More precisely, we enable the integration of different annotation services and realize service providing at the proper time, depending on the current individual or group's state of the work. For that purpose, we applied a service-oriented infrastructure. Hence, the emphasis is to handle components as modular services that are provided to clients [16]. An essential requirement for process management within service-oriented infrastructures is the coordination of shared resources [16]. In general, among these resources are the available annotation services and the single users or user groups. In order to enable management of process-related resources, an appropriate workflow management concept is necessary. Workflow Management comprises all tasks that are to be accomplished at modeling, simulation, and execution and supervision of workflows [34]. A workflow can be regarded as a business process that is executed by means of computing devices [34]. Such processes are automatized, and documents, tasks, or contents are transferred according to specified rules [42]. A process-based system must be driven by a specific process model [24]. According to that, the concrete applied processes must be modeled and fed into the framework in forms of a schema or workflow specification [16]. Workflow schemata describe a process by means of a workflow modeling language for the purpose of documentation, analysis, or execution within a workflow management system [34]. In the following, we specify a formal description of collaborative multimedia annotation processes.

That formal specification aims at supplying relevant requirements for workflow-support in collaborative annotation settings. First, the general tasks (work items), involved annotation services, multimedia contents, and users are specified. Additionally, essential attributes for user assistance within a workflow can be set, such as the state of a task, service types in order to also realize real-time service assignment, e.g. "selection of video segment", and definition of content types. Tasks may be structured in an hierarchical order, e.g., the superior task "addition" can obtain the sub-tasks "annotate metadata" or "add comment" [24, 39]. Transitions between related or subsequent tasks are defined. Thus, the operative flow is specified and workflow control procedures are driven. Furthermore, transitions are linked with contents that are transfered from one task to the following. In doing so, the data flow referring the annotation flow can be defined. Tasks are also assigned to available services, as well as to the single users or user groups that have to execute them. As a result, service providing is supported at a user interface level. In addition to that, on user interfaces, personalized views of the workflow can be provided. As stated in [7, 8] personalization is a key requirement at workflow visualization. Here, personalization is realized at two areas: first, at the area of tasks that have been assigned to users and groups with respective authorizations and, second, the concrete contents that are processed by a user or group. In particular, the latter case can be regarded as useful for selection support. In the following, a formal specification of (collaborative) multimedia annotation workflows, is presented.

Definition 1. An annotation process is defined by a tuple $p = (i_p, W, T, S, C, G, WS, ST, CT)$ where:

- ip is the unique identifier of p.
- $W = \{w_1, w_2, \ldots, w_n\}$ is a finite set of work items which assemble p.
- $T = \{t_1, t_2, \ldots, t_n\}$ is a finite set of transitions located between consecutive work items.
- $S = \{s_1, s_2, \ldots, s_n\}$ is a finite set of services which are offered to perform the single work items included in p.
- $C = \{c_1, c_2, \ldots, c_n\}$ is a finite set of contents that are transfered in between work items.
- $G = \{g_1, g_2, \ldots, g_n\}$ is a finite set of user groups which execute the single work items included in p.
- $WS : W \rightarrow WorkItemStates := \{NotEnabled, Enabled, Active, Accomplised\}$ indicates the state for each work item $w \in W$.
- $ST : S \rightarrow ServiceTypes := \{Configuration, Selection, Addition, Exploration, Externalization, Document, Element, ElementComposition, ElementSubset, Text, Graphic, Audio, Video, 3D, GenericFormat\}$ assigns a set of service types for each service $s \in S$, indicating for what kind of tasks and contents a service can be used.

- $CT: C \rightarrow ContentTypes := \{Document, Element, ElementComposition, ElementSubset, Text, Graphic, Audio, Video, 3D, GenericFormat\}$ indicates the content type for each content $c \in C$ transfered in p.

Definition 2. A work item is defined by a tuple $w = (i_w, i_{pw}, d, S_w)$ where:

- i_w is the unique identifier of w.
- i_{pw} is the unique identifier of the parent work item of w (optional).
- d is the deadline of w (optional).
- $S_w \subseteq S = \{s_{w1}, s_{w2}, \ldots, s_{wn}\}$ is a finite set of one or more services which are offered to perform w.

Definition 3. A transition is defined by a tuple $t = (i_t, i_o, i_d, C_o, C_i)$ where:

- i_t is the unique identifier of t.
- i_o is the unique identifier of the origin work item, from which the transition t is launched.
- i_d is the unique identifier of the destination work item, at which the transition t targets.
- $C_o \subseteq C$ is a finite set of contents that are output data of the origin work item.
- $C_i \subseteq C$ is a finite set of contents that are input data for the destination work item.

Definition 4. A service is defined by a tuple $s = (i_s, l_s)$ where:

- i_s is the unique identifier of s.
- l_s indicates the location of s.

Definition 5. A user group is defined by a tuple $g = (i_g, U_g, W_g)$ where:

- i_g is the unique identifier of g.
- U_g is a finite set of users belonging to g.
- $W_g \subseteq W$ is a finite set of work items which are to be accomplished by the group g (optional).

Definition 6. A user is defined by a tuple $u = (i_u, g, W_u, R)$ where:

- i_u is the unique identifier of u.
- $g \in G$ is the group that the user belongs to.
- $W_u \subseteq W$ is a finite set of work items which are to be accomplished by the user u (optional).
- $R = \{r_1, r_2, \ldots, r_n\}$ is a finite set of roles that are assigned to u. Role assignment also implicates several access rights respecting the content(s) that are to be handled.

Definition 7. A content is defined by a tuple $c = (i_c, i_{pc}, l_c)$ where:

- i_c is the unique identifier of c.
- i_{pc} is the unique identifier of the parent content of c (optional).
- l_s indicates the location of c.

Process Management II – Workflow Visualization and Process-Driven Assistance

The specification of concrete workflows is fundament for workflow execution. Within the presented concept, workflow management particularly focuses on workflow visualization and process-driven assistance. The general objectives of these aspects are to provide users information about what are the current state and task(s), which contents are involved, and which services can be selected in order to accomplish a respective tasks. In that context, a visual-interactive concept for *Workflow Visualization* and *Process-driven User Assistance* is illustrated. Especially in the scope of collaborative annotation scenarios, the processes may be long-running, include a large number of activities, and involves several user groups and roles [7]. In that case, users have different tasks and knowledge about the process. Thus, a uniform workflow visualization for all participants can not cover all expectations and individual requirements [8]. The process must be presented in various ways, including a personalized visualization with an appropriate level of granularity [6–8]. In doing so, the specific needs of different user groups can be fulfilled. For that purpose (a) the process visualization must be able to be reduced by hiding not-relevant information, (b) the representation of process elements such as activities or data must be customizable [6, 7]. In particular, we refer to two dimensions of personalization: control and data flow. Control flow includes a personalized process visualization referring to the activities or tasks that a certain user has to perform. For example, a user with an administrator role may be allowed to conduct any tasks, especially system configuration. Whereas, a "normal annotator" will have only authorization for selecting and adding additional information. From a data flow perspective, personalized process representation only shows the contents that are assigned to the current user and/or his assigned group. Assuming that two groups have created video segments with respect to different topical categories. In the next step, addition of textual contributions, the process representation only highlights the video segments that were created by the group of the current user. Process-driven assistance is a part of the presented workflow visualization concept. First, it includes guidance of users when they specify the necessary parameters before effectively performing a task. Furthermore, assistance is given by indicating the available services that are to be used in the scope of the current state of process depending on the predefined process-related parameters on the one hand, and on the other hand further attributes that are gathered from a process-specification. Such as for process visualization, also user assistance has to consider different types of users, since the expertise of a single user referring to the operation of the application determines the level of assistance that is required. In the following, the concepts of workflow visualization and process-driven user assistance are illustrated. For that purpose, we refer to the prototype application *SemaWeb* that allows semantic annotation of web contents. Here, the multimedia annotation framework and a workflow visualization component are integrated.

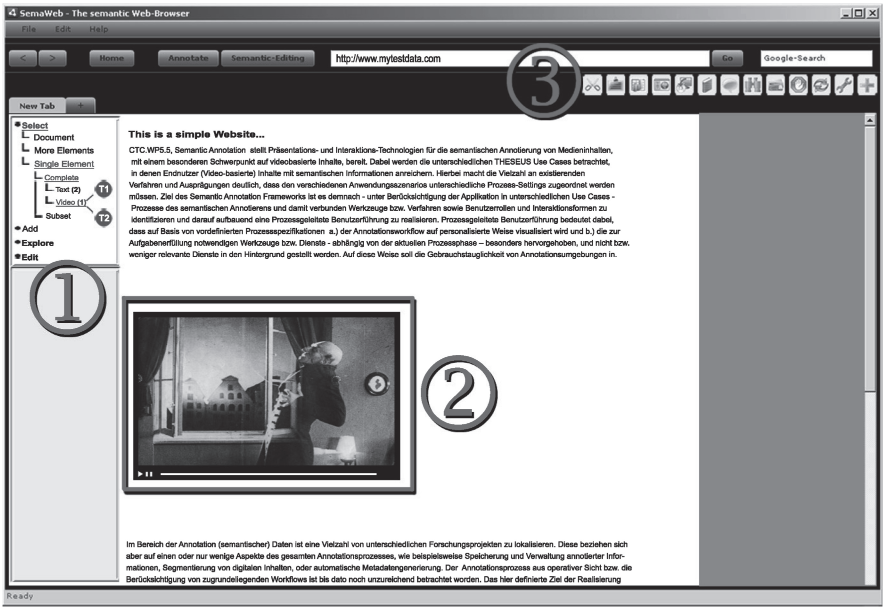

Fig. 3 Workflow visualization and process-driven assistance in *SemaWeb*

As shown in Fig. 3 (1), predefined processes are presented by means of a tree visualization. On the first level, the general superior tasks of the project are listed. Child nodes of all tiers map the next category within a thread of decisions that have to be made by the user in order to initiate the required task. That is, various properties are requested by the system that determine the parameters of task execution. The general idea of the tree view concept is that users are enabled to select a path of the tree structure which guides them through the several options, feeding the system with required information at the same time. Thus, the leafs of the tree are always available services that match to the activated path options. The items displayed by that component are personalized: only the relevant data is shown based on task distribution defined in the loaded process specification and the user management data. The simple example in Fig. 3 shows that a user wants to select a whole video on a website in order to annotate some descriptive metadata. For that purpose, the user clicked the options *select*, *single element*, *whole element* (and not a segment), and *video media*. The available videos are shown in the tree, and after selection of the video two existing video selection tools are faded in. That way of interaction supports users in small steps, and we suggest that especially users with less experience concerning the usage of annotation systems will benefit. Still, other types of more experienced users may not accept that kind of step-by-step guidance. Consequently, we distinguish between the three user types novices, advanced, and expert. According to that, as shown in Fig. 3 (2), advanced or expert users are also able to select the desired contents by means of mouse actions upon the website.

In that case, the tree view responds to that activities and adopts the representation, so that advanced users are still able to localize applicable services by means of the process visualization. Thus, they are allowed to overleap single steps within the tree path. At an expert level, users are also able to select the right tools directly without using the tree view component (see Fig. 3 (3)). To sum up, the process visualization and assistance concept covers multiple forms of habit, knowledge, experience, and specific tasks of the different users and user groups.

Conclusion and Future Work

We illustrated that even though research in the area of digital annotation has long tradition and includes an amount of respective projects the practices and workflows of annotation has not been sufficiently considered to date. Particularly, that applies to collaborative practices which gain more and more in importance. Due to the wide range with respect to types, usages, application areas, and users we emphasized the large number of different workflows of (collaborative) annotation. Accordingly, the main contribution of this paper is the illustration of a concept for a integrable framework that supports multiple collaborative workflows of digital multimedia annotation. Based on a model that describes the annotation process in an abstract way, we introduced a respective architecture model. The significant aspects of the architecture are the separation of the basic elements included, so that the framework can be integrated as a subsystem into existing environments. By means of a service-oriented infrastructure, different components that serve as specific annotation services can be integrated and coordinated. In addition, we presented a formalized specification of the collaborative annotation process including its essential aspects and elements. Based on that formalization, concrete workflow specifications can be generated in order to realize workflow management. In the scope of workflow management, a visual-interactive approach for user-centered support by means of workflow visualization and process-driven assistance was introduced. Basically, that approach deals with providing users with information about the workflow tasks and the services or user interface elements that are to be used with the respect to the current state within the annotation process. There, personalization of the view and the interactive access on the workflows are means of supporting collaboration. In doing so, various levels of knowledge and expertise, as well as use styles and preferences are considered. In summary, the presented concept aims at improving the use of annotation systems, especially in the scope of collaborative scenarios. Up next, the concepts included in the presented framework are going to be evaluated with the aid of usability tests and case studies.

Due to the wide range and complexity of the annotation field, several points of contact for future research work can be identified. Especially in the scope of our research work, two topics are of special interest concerning annotation workflow visualization and assistance through service providing. The first question refers to how process representation and recommending of services can be improved by machine-learning approaches, e.g., by realizing adaption based on automatic interaction analysis. Second, service providing and retrieval can be evolved by advanced approaches from the semantics field, such as research on specific service ontologies.

References

1. Agosti, M., Bonfiglio-Dosio, G.,Ferro, N.: A historical and contemporary study on annotations to derive key features for systems design. Int. J. Digit. Libr. 8(1), 1–19 (2007)
2. Agosti, M., Ferro, N.: A formal model of annotations of digital content. ACM Trans. Inf. Syst. 26(1), 3 (2007)
3. Bertini, M., Bimbo, A. D., Nunziati, W.: Automatic Annotation of Sport Video Content. Alberto Sanfeliu, Manuel Lazo-Cortés (ed.): CIARP, Springer, 1066–1078 (2005)
4. Bertino, E., Trombetta, A., Montesi, D.: Workflow Architecture for Interactive Video Management Systems. Distrib. Parallel Databases 11(1), 33–51 (2002)
5. Bischoff, K., Firan, C. S., Paiu, R.: Deriving music theme annotations from user tags. WWW '09: Proceedings of the 18th international conference on World wide web, ACM, New York, USA, 1193–1194 (2009)
6. Bobrik, R., Bauer, T.: Towards Configurable Process Visualizations with Proviado. WETICE, IEEE Computer Society, pp. 367–369 (2007)
7. Bobrik, R., Reichert, M., Bauer, T.: View-Based Process Visualization. Gustavo Alonso, Peter Dadam, Michael Rosemann, ed., BPM, Springer, pp. 88–95 (2007)
8. Bobrik, R., Reichert, M, Bauer, T.: Requirements for the Visualization of System-Spanning Business Processes. DEXA Workshops, IEEE Computer Society, pp. 948–954 (2005)
9. Brugman, H., Russel, A.: Annotating Multi-media/Multi-modal resources with ELAN. Proceedings of the 4th International Conference on Language Resources and Language Evaluation (LREC 2004), European Language Resources Association, Paris, pp. 2065–2068 (2004)
10. Cadiz, J. J., Gupta, A, Grudin, J.: Using Web annotations for asynchronous collaboration around documents. CSCW '00: Proceedings of the 2000 ACM conference on Computer supported cooperative work, ACM, New York, USA, pp. 309–318 (2000)
11. Constantopoulos, P., Doerr, M., Theodoridou, M., Tzobanakis, M.: On Information Organization in Annotation Systems. Intuitive Human Interfaces for Organizing and Accessing Intellectual Assets, pp. 189–200 (2004)
12. Costa, M., Correia, N., Guimaraes, N.: Annotations as multiple perspectives of video content. MULTIMEDIA '02: Proceedings of the tenth ACM international conference on Multimedia, ACM, New York, USA, pp. 283–286 (2002)
13. Finke, M.: Unterstuetzung des kooperativen Wissenserwerbs durch Hypervideo-Inhalte. Universitäts- und Landesbibliothek Darmstadt (2005)
14. Frommholz, I., Brocks, H., Thiel, U., Neuhold, E. J., Iannone, L., Semeraro, G., Berardi, M., Ceci, M.: Document-Centered Collaboration for Scholars in the Humanities – The COLLATE System. ECDL '03, Springer, pp. 434–445 (2003)
15. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design Patterns. Elements of Reusable Object-Oriented Software, Addison-Wesley, Reading, MA (1995)
16. Gogouvitis, S. V., Kousiouris, G., Konstanteli, K., Polychniatis, T., Menychtas, A., Kyriazis, D., Varvarigou, T.: Realtime-enabled workflow management in service oriented infrastructures. AREA '08: Proceeding of the 1st ACM workshop on Analysis and retrieval of events/actions and workflows in video streams, ACM, New York, USA, pp. 119–124 (2008)
17. Hagedorn, J., Hailpern, J., Karahalios, K. G.: VCode and VData: illustrating a new framework for supporting the video annotation workflow. AVI '08: Proceedings of the working conference on Advanced visual interfaces', ACM, New York, USA, pp. 317–321 (2008)

18. Halasz, F. G., Schwartz, M.: The Dexter Hypertext Reference Model. Communications of the ACM 37(2), 30–39 (1994)
19. Havemann, S., Fellner, D. W.: Seven Research Challenges of Generalized 3D Documents. IEEE Comput. Graph. Appl. 27(3), 70–76 (2007)
20. Hofmann, C., Hollender, N., Fellner, D. W.: Workflow-Based Architecture for Collaborative Video Annotation. A. Ant Ozok, Panayiotis Zaphiris (ed.), HCI (12), Springer, The Netherlands, pp. 33–42 (2009)
21. Hofmann, C., Hollender, N., Fellner, D. W.: A WORKFLOW MODEL FOR COLLABORATIVE VIDEO ANNOTATION – Supporting the Workflow of Collaborative Video Annotation and Analysis performed in Educational Settings. Proceedings of the International Conference on Computer Supported Education (CSEDU 09), INSTICC Press, Lissabon, Portugal, pp. 199–204 (2009)
22. Hollender, N., Hofmann, C., Deneke, M.: Principles to reduce extraneous load in web-based generative learning settings. Workshop on Cognition and the Web 2008, pp. 7–14 (2008)
23. Kadobayashi, R., Lombardi, J., McCahill, M. P., Stearns, H., Tanaka, K., Kay, A.: Annotation authoring in collaborative 3D virtual environments. ICAT '05: Proceedings of the 2005 international conference on Augmented tele-existence, ACM, New York, USA, pp. 255–256 (2005)
24. Leoni, M.; Aalst, W. M. & Hofstede, A. H.: Visual Support for Work Assignment in Process-Aware Information Systems. BPM '08: Proceedings of the 6th International Conference on Business Process Management, Springer-Verlag, Berlin, Heidelberg, pp. 67–83 (2008)
25. Lin, C.Y., Tseng, B. L., Smith, J. R.: Video collaborative annotation forum: Establishing ground-truth labels on large multimedia datasets. Proceedings of the TRECVID 2003 Workshop (2003)
26. Marshall, C. C.: Toward an ecology of hypertext annotation. HYPERTEXT '98: Proceedings of the ninth ACM conference on Hypertext and hypermedia: links, objects, time and space—structure in hypermedia systems, ACM, New York, USA, pp. 40–49 (1998)
27. Marshall, C. C., Brush, A. J. B.: Exploring the relationship between personal and public annotations. JCDL '04: Proceedings of the 4th ACM/IEEE-CS joint conference on Digital libraries, ACM, New York, USA, pp. 349–357 (2004)
28. Marshall, C. C., Ruotolo, C.: Reading-in-the-small: a study of reading on small form factor devices. JCDL '02: Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries, ACM, New York, USA, pp. 56–64 (2002)
29. Mikova, M., Janik, T.: Analysis of elements supporting health in physical education: Methodological approach of video study. Janík T. Wagner R. Mužík, V., ed.,'Mužík, V., Janík, T., Wagner, R. (ed.) Neue Herausforderungen im Gesundheitsbereich an der Schule. Was kann der Sportunterricht dazu beitragen?, MU, Brno, pp. 248–260 (2006)
30. Pea, R.,Hoffert, E.: Video Research in the Learning Sciences. Goldman, R., Pea, R., Barron, B., Derry, S.J. (ed.) Video workflow in the learning sciences: Prospects of emerging technologies for augmenting work practices. Lawrence Erlbaum, London, pp. 427–460 (2007)
31. Pea, R., Mills, M., Rosen, J., Dauber, K., Effelsberg, W., Hoffert, E. The Diver Project: Interactive Digital Video Repurposing. IEEE MultiMedia 11(1), 54–61 (2004)
32. Reif, G.: Semantische Annotation. Semantic Web - Wege zur vernetzten Wissensgesellschaft 5(4), 405–418 (2006)
33. Richter, K., Finke, M., Hofmann, C., Balfanz, D.: Hypervideo. Pagani, M. (ed.) Encyclopedia of Multimedia Technology and Networking, 2nd ed., Idea Group Pub., USA, pp. 641–647 (2009)
34. Schaetzle, R.: Workflow-Management – ein ereignisbasierter Ansatz. PhD thesis, Universitat Karlsruhe (TH) (2000)
35. Seidel, T., Prenzel, M., Kobarg, M. (Eds.): How to run a video study. Technical report of the IPN Video Study. Waxmann, Münster (2005)
36. Sesink, W., Goeller, S., Roeßling, G., Hofmann, D.: emargo: Eine Digitale Randspalte zum Selbststudium (nicht nur) der Informatik. Proceedings der Pre-Conference Workshops der 5. e-Learning Fachtagung Informatik (DeLFI 2007), Logos Verlag, Berlin, pp. 101–108 (2007)

37. Sliski, T. J., Billmers, M. P., Clarke, L. A., Osterweil, L. J.: An architecture for flexible, evolvable process-driven user-guidance environments, SIGSOFT Softw. Eng. Notes 26(5), 33–43 (2001)
38. Stahl, E., Finke, M., Zahn, C.: Knowledge Acquisition by Hypervideo Design: An Instructional Program for University Courses. Journal of Educational Multimedia and Hypermedia 15(3), 285–302 (2006)
39. Streit, A., Pham, B., Brown, R.: Visualization Support for Managing Large Business Process Specifications. Business Process Management, 3649, 205–219 (2005)
40. Volkmer, T., Smith, J. R., Natsev, A.: A web-based system for collaborative annotation of large image and video collections: an evaluation and user study. MULTIMEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia, ACM, New York, USA, pp. 892–901 (2005)
41. Wang, M., Zhou, X., Chua, T.-S.: Automatic image annotation via local multi-label classification. CIVR '08: Proceedings of the 2008 international conference on Content-based image and video retrieval, ACM, New York, USA, pp. 17–26 (2008)
42. Workflow Management Coalition Terminology & Glossary. Document Number WFMC-TC-1011, Issues 3.0, Technical report (1999)
43. Zheng, Q., Booth, K., McGrenere, J.: Co-authoring with structured annotations. CHI '06: Proceedings of the SIGCHI conference on Human Factors in computing systems, ACM, New York, USA, pp. 131–140 (2006)