

Tracking Actual Usage: the Attention Metadata Approach

Martin Wolpers, Jehad Najjar, Katrien Verbert and Erik Duval

Katholieke Universiteit Leuven, Belgium // {martin.wolpers, jehad.najjar, katrien.verbert, erik.duval}@cs.kuleuven.be

ABSTRACT

The information overload in learning and teaching scenarios is a main hindering factor for efficient and effective learning. New methods are needed to help teachers and students in dealing with the vast amount of available information and learning material. Our approach aims to utilize contextualized attention metadata to capture behavioural information of users in learning contexts that can be used to deal with the information overload in user centric ways. We introduce a schema and framework for capturing and managing such contextualized attention metadata in this paper. Schema and framework are designed to enable collecting and merging observations about the attention users give to content and their contexts. The contextualized attention metadata schema enables the correlation of the observations, thus reflects the relationships that exists between the user, her context and the content she works with. We illustrate with a simple demo application how contextualized attention metadata can be collected from several tools, the merging of the data streams into a repository and finally the correlation of the data.

Keywords

Context, Attention Metadata, Usage Data, User Behaviour, Attention Recorders

1. Introduction

Knowledge intensive work like teaching and learning requires the handling of large amounts of information in a personalized way. Therefore, learning management systems need detailed user profiles to be able to provide personalized services (Duval & Hodgins 2003). Instead of basing personalization on stereotypes (Henze & Nejdil 2003), such systems utilize detailed information about a specific user including observations on the handling of digital content and the user attention. This allows information systems to more correctly conclude on the user aims and goals (Jones et al., 2000, Najjar, et al., 2004).

Recent research focuses on the attention of users based on the hypothesis that attention information supports correct conclusions on the user aims and goals. Attention here refers to which activities the user carries out on her computer with which content and in which context.

Two examples illustrate how observations about the attention of a user can help her to deal with large amounts of digital information. The first example, consider the scenario to enable personalized access to information. Martin is a lecturer and assembles course material for his course on “Multimedia Modelling and Programming”. By analyzing what he has done so far, and by looking at what other people following the same task have done, the system can help him find suitable learning material – suitable with respect to the course but also with respect to Martin’s knowledge and preferences derived from his previous interactions. By also taking the previous interactions of his students with the content of the course into account, the system can help Martin identify the knowledge that the students are still missing so that he can then include it in his course. Furthermore, based on the observations how the students deal with the course material Martin has provided so far, he is able to tailor the material to their needs. For example, if the majority of students prefer to work with graphical representations, Martin can tailor his course material to this need by including more images, videos, tables, etc. Without this information, Martin would include the material he thinks (but not knows) is best suited for his students.

The second example deals with the support of the system in managing the vast amount of available information, by driving the information provision through analysis results of the information sources and their handling. While Martin works with the course material, he wants to receive emails related to his course only, but wants to block out any other communications. Therefore, incoming emails from his students are shown but emails from colleagues related to completely other subjects are suppressed. In addition, if a student email asks for an appointment, the email will be automatically annotated with possible meeting times that, if acknowledged, are automatically inserted into Martin’s diary. Furthermore, while working, Martin likes to listen to music from musicals. The system has already

observed this behaviour and therefore provides him with a selection of suitable titles. Incoming chats are blocked unless related to the course. The system also helps Martin find appropriate course material: while searching for course material, he is automatically pointed to recommended sources of material. The necessary recommendations are, for example, extracted from emails, documents and chats and ranked according to how Martin appreciated the sender and/or author.

What is needed to enable such scenarios is a technology that easily spans across system boundaries and captures the attention a user spends on content (Wolpers et.al. 2006). We propose a framework that is able to capture the observations about the user activities with digital content. We explain our understanding of attention, observations about the user, digital content and context in chapter 2. In order to describe the observations, we developed a metadata schema that is outlined in chapter 3. A case study in section 4 provides first examples on how advanced services use the observations made about the user, her attention and her context. Concluding this paper, section 5 discusses first results and future work.

2. Attention

Collecting appropriate data to enable improved personalized services is the focus of this paper. This is achieved by collecting and formalizing observations on how and what the user spends her attention and in which context. In general, there is no agreed definition of attention. Nevertheless, as Roda and Thomas point out (Roda & Thomas, 2006), “most researchers refer to attention as the set of processes enabling and guiding the selection of incoming perceptual information.” We follow this interpretation and apply it to the everyday handling of digital content. By limiting our approach to unobtrusively observing user behaviour in dealing with information, the computer of the user is used as the major source of observations. Note, that when referring to digital content, the term document is used in the remainder of this paper. A document can be a web page, a word file, a jpeg images, an email, a chat session, an mp3 music file, etc.

The user deals with the documents in a particular way that is specific only to her. By capturing and generalizing her behaviour, emerging patterns of her behaviour can be used to describe her. For example, such patterns can be derived from data observing how the user gets the documents, e.g. via email, chat or if she found them through web search services, etc. Her behaviour while working with the documents provides further data; e.g. just storing it or copy parts to be used on a slide show, etc.

The observations can be generalized into behavioural patterns. Behavioural patterns describe in general how a user handles information, e.g. which activities she carries out with them. The comparison of behavioural patterns from various users allows clustering similar users. Based on such clusters, we expect to be able to precisely predict future steps and goals of user.

2.1 Capturing User Attention

Capturing observations about the attention of users requires solving two problems: First, applications need to be developed that capture the observations of the user attention. Such applications need to integrate into the user daily working environment without disturbing or interrupting her. Moreover, the applications need to capture the observations from existing applications of daily use, e.g. the Microsoft Office Suite, Web Browsers, Mail Clients, the WINAMP Music Player or MSN Messenger. A number of suitable tools are provided in chapter 4. Each of these tools continuously provides observations, thus generates a stream of information on the interactions of the user with the respective application.

Furthermore, user observations need to be captured in a generalized format that allows merging and processing of the various streams of observations. By merging the observations, it is possible to contextualize each observation, e.g. by identifying which activities the user carried out simultaneously or within a short time-span. For example, a context that can be identified is which music the user listened to while writing an email, with which content and to whom. Another example is with which keywords the user found relevant documents which she really wanted. We therefore speak of contextualized observations of the user attention. We broaden our notion of context by describing the context through all additional information (e.g. information about parallel activities) available at the time the user

activity is taking place. For example, consider our previously outlined scenario where Martin puts course material together for his course on “Multimedia Modelling and Programming.” He might stop this activity to resume it a couple of days later. We aim to keep the context in which the course material is assembled thus must also capture when Martin is resuming working on the respective course material. Therefore, from our perspective, contextual information is related to the observed user activity in time and/or content.

There are numerous approaches to capture observations about the user attention; e.g. within the European projects Nepomuk (2006), Aposdle (Lindstaedt, et al., 2006), Gnows (Sauermann, 2005) as well as others (Völkel & Haller, 2006), (Holz, et. al., 2006), (Braun & Schmidt, 2006), (Garofalakis, et al., 2006), (Broisin, et al., 2005), (Roda & Thomas, 2006). The focus lies on the classification of the observations according to some predefined taxonomy or ontology. The predefined ontology(ies) usually describes a specific set of activities that is usually related to some task or process. Furthermore, the underlying tasks and processes are also described through pre-configured ontologies. As this works perfectly for the purposes for which the ontologies were made, the approaches fail when the general and usually less predictable activities of users are to be observed. Therefore, observations that do not fit the classification are disregarded in these systems.

Other approaches focus on capturing observations about the user by monitoring key strokes, mouse gestures, click-streams, etc. These approaches provide a vast amount of data that is highly fine-grained. This fine-grained data is highly valuable to examine how a user used a website (Weinreich, et al., 2006a) or an information system. Nevertheless, it is quite problematic to derive user interest and behavioural patterns from such data. One reason is the noise included in the data (Weinreich, et. al., 2006b), the other is that it is not related to the user working contexts. The data is captured without keeping track of and without relating the activities where the content in question was involved in, usually because of its highly granular structure. In addition, this data is provided from one tool without taking the content into account on which the activities are performed. Therefore, the data cannot really support contextualizing the streams.

Our approach is more general, because it observes the user at the application level instead of the key-stroke level. Furthermore, our approach specifies a common schema in which the observations should be captured. This enables correlating the various application specific data streams and thus contextualizes them.

We have developed the contextualized attention metadata (CAM) schema to capture the observations. Previous versions have been published in (Najjar et.al. 2006a,b). It is based on attentionXML (2004), which is an open specification to capture data on how people use information in browsers, web pages, news feeds and blogs. This data is then analysed and provides interested users with statistical information on their interests and activities over time. Furthermore, interested parties use the data to predict trends, to identify trend setters, etc.

AttentionXML is an XML-based schema that groups the captured observations according to users. Each user has one or more feeds. A feed can be a click stream within a browser, news feeds, etc. thus all information obtained and exchanged through the web asynchronously, therefore without regarding email and chat as observation sources. Each feed has a number of properties attached that can but must not be used, e.g. the title, the URL, etc. Furthermore, some user-specific data is captured like the date and time when it caught the user attention and when the user discarded it, user specific arbitrary tags, voting information, etc. Each feed consists of one or more items which are described through a number of properties such as the global identifier (GUID), the title, when it was last read, when it was last updated, its duration, its MIME type, if the user followed any links within this item, voting data and tags.

AttentionXML is targeted to capture observations of users related to browsing behaviour and information consumption in blogs and news feeds. It therefore does not allow capturing observations related to activities like searching and downloading documents on the web, reading, writing and editing documents, listening to music, communication messages (e.g. chats), etc. Furthermore, it does not allow capturing the context within which an activity took place. We therefore propose an extension to AttentionXML that allows us to significantly broaden the types of observations that can be captured and described and the context where they occur. Our extension enables capturing the type of event (including all relevant properties) that occurs with each item. The type of event allows us to classify the event that belongs to each item. We therefore break the implicit assumption of AttentionXML up, that each item is either a news-feed, a wiki or blog item. This approach allows us to capture observations about user activities in any kind of tool, not just a browser or newsreader as in AttentionXML. We call the extension Contextualized Attention Metadata (CAM) and explain it in detail in chapter 3.

2.2 Collecting Contextualised Attention Metadata

As outlined above, we aim to collect observations in the CAM format from all applications available in the user computer environment. We identify sources along the dimensions: location of the observation generating application and its social type. The location describes where the CAM data is generated, either on the user desktop or at the server-side. The social type describes if the observations relate to the user alone or to her social interactions with other users. For example, the MS Office application or a web browser focus on the user alone, while an email and chat client provides CAM data that describes her social relations with other users. This simple classification is used to identify the various possible types of correlations of observation data streams.

Our main focus, for the moment, is on applications that are used on a daily basis by the user, like MS Office, Browsers, etc. Web browser extensions like Slogger (2003) already enable the capturing of the user browsing behaviour in web environments. Tools based on ALOCOM (Verbert, et al., 2005) enable the capturing of CAM data from MS Office tools. Examples of their application are given in chapter 4.

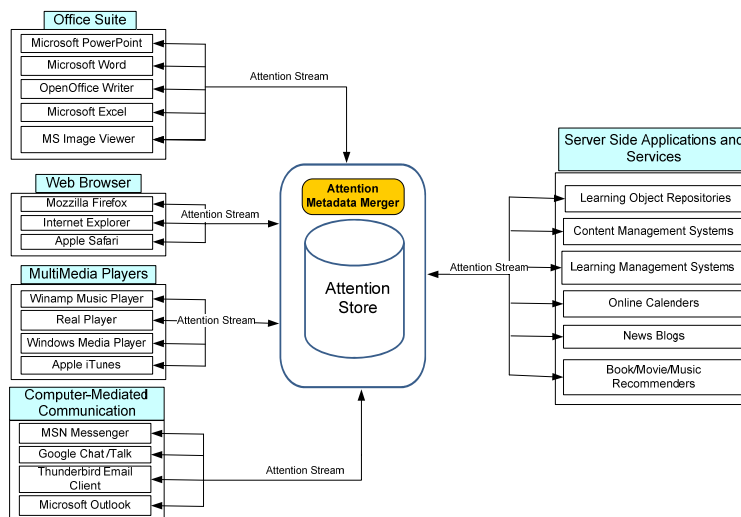


Figure 1: The CAM framework

Apart from desktop applications as sources, log files from centralized systems like web search engines, information repositories or virtual worlds like “Second Life” (2006) and online games like Warcraft (2006) provide highly valuable sources for observations about users. Recently, more and more network systems use the open log file format of the Apache foundation. Therefore, we will map the Apache open log file format to CAM and thus be able to correlate CAM data from desktop applications to CAM data from web-based systems.

The CAM framework is developed, shown in figure 1, to collect, merge and store the various streams of observations in CAM format. Each desktop application (left side of the above figure) generates a stream of observations that captures activities within the application with timestamp and content-related data. These application-related streams are collected and merged into a single stream per user. The CAM format enables therefore the provision of attention metadata streams on a per user basis that is further categorized into the various applications. Merging is therefore rather simply done based on the user category of CAM. Afterwards, the streams are stored in the CAM store.

CAM streams from central systems like web servers (right side of the above figure) leave the user category of the CAM stream empty. These streams are already merged on the application category and can simply be stored in the CAM store.

3. Contextualized Attention Metadata schema (CAMs)

Figure 2 shows the CAM schema developed to allow tracking user activities across different systems.

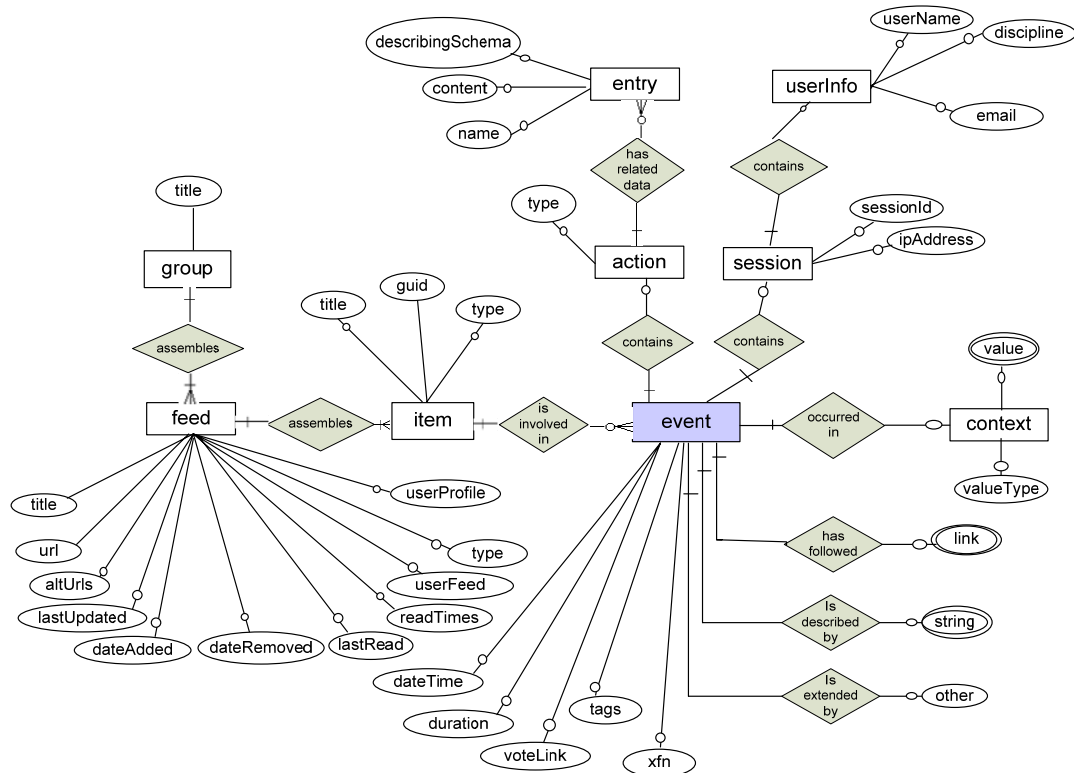


Figure 2: The CAM schema elements (in ER notation)

The CAM schema is designed to allow tracking user activities in all systems she may interact with while working with documents. As shown in figure 2, CAMs collects user attention in all systems within one *group* element. The purpose of the *feed* element is to group the attention of the user in one specific system. The *item* element collects the attention given to one specific digital document. Because of the fact that every digital document may be accessed on different occasions and be involved in different tasks (reading, editing, updating, listening to, etc.), it requires capturing information related to every event in which the document was involved in. As this is not possible in AttentionXML, we extended AttentionXML at the *item* element level, by adding the *event* element. Each item may be involved in one or more event that has different relevant information. For example, in one event, the document can be edited in one system and afterwards, in another event, the same document can be read or updated. Note that the same document may also be accessed in another system. Each time a document is accessed, more attention metadata is collected, like access datetime, context, duration spent on working with the document and data actively created by the user (ranking, annotations and tags). A teacher may use the same document in her online courses for two different groups of students and in two different contexts (time, application, and topic). In each course where the document is used, different metadata information can be collected, like the time spent on learning it, the topic (computer, management, etc.) of the course and the teacher evaluation of the usefulness of the document to each group of students (it might be well perceived by computer science students but not by physics student).

Making the *event* element central in CAM schema allows the identification and relation of information about every event the document was involved in across different systems. As also shown in figure 3, in the CAM schema the elements *duration*, *voteLink*, *xfn* and *tags* are moved from the item element into the event element. This reordering enables the identification of the duration spent with the document, tags given to the document by the user, social relationship and user experience (*voteLink*; like-dislike information) per event.

We will now describe the *item* element and its sub-elements including the *event* element, where we extend AttentionXML. The *group* element groups all attention metadata of one user in all applications she may work with (all in one group instance) while the *feed* element groups the attention of one user in one specific tool (each tool in a

separate feed instance). The group and feed elements of CAM schema are the same as in AttentionXML and therefore will not be further explained.

- **Item:** the *item* element groups attention metadata of one document. Each document can be involved in different actions (read, listen to, edit, etc.), in different dates, for different periods, and in different contexts.

The *item* element has three sub-elements that do not change over the different actions and events the document may be involved in; those elements record the properties of the document itself. The data is collected here to recognize and identify the document across different systems and contexts:

- **Title:** the *title* captures a human readable name given to the document, when the document is created or edited. This element is necessary to enable users to easily recognize the document. For example, “Global warming - Wikipedia, the free encyclopedia” is a title of a document about global warming on wikipedia.
- **GUID:** the *guid* element represents a global unique identifier to the document within a given context. For example, http://en.wikipedia.org/wiki/Global_warming is the unique identifier that is used to locate the document with the title mentioned above on the internet.
- **Type:** the *type* element holds the MIME technical type of the document. For example, “html” is the correct MIME type of the above document on global warming.

On the other hand, all data about the different events the document may be involved in are grouped in the event element. This data describes the attention given to the document, like the time spent with it, tags attached to the document after reading it, and the context where it was used. Each event related data is grouped in one *event* instance.

- **Event:** the *event* element groups attention metadata of each event the document was involved in. For every event instance the following attention metadata is collected:
 - **Action:** the *action* element provides information on the action that the document was involved in (e.g. if it was inserted into local file system or digital repository, opened in a viewer application, etc.):
 - **Action Type:** the *actionType* element holds the type of action (task) the document was involved in. Its value is normally a reference to a value in the action value space. For example, the URL <http://.../Actiontype/insert> can be a reference to the insert value if the action was inserting a document into a repository.
 - **Entry:** the *entry* element records data related to the action performed. About the same action one or more entries can be recorded. For example, if the document was found using a query, one instance of the *entry* element can store the query terms used by the user to form her query. Another entry can store the results list of that query. In case of insertion, it also records the name of metadata schema (for example IEEE LOM or Dublin Core) used to index the document. If the action was chat conversation with other person, this element can store name of the chat partner in one entry instance and the text of the conversation in another entry instance.
 - **Date Time:** the *dateTime* element holds the date and time at which the event took place. Unlike the *item.lastRead* element in AttentionXML, this element keeps all timestamps of events where the document was involved.
 - **Duration:** the *duration* element records the time spent with the document (in seconds).
 - **Session:** the *session* element holds the information that is needed to identify the working session.
 - **Session ID:** the *sessionId* element holds a unique identifier for the session.
 - **IP Address:** the *ipAddress* element holds the IP address of the user computer.
 - **User Info:** the *userInfo* element collects information about the user name, email address and scientific discipline of the user performing the action.

The session information (*sessionId* and *ipAddress*) are used to identify the user throughout the different events and tasks she may interact in with the document. The data about the user is collected per event because the same user may have, for example, different user name, IP address every time she works with the same document. Working with a document from the computer at work or at home may result in different IP addresses for the same user.

- **Context:** the *context* element captures information that describes the environments the user may interact with. For example, information about a course (discipline and description) where a user has uploaded a document. The title and description of a course about Human Computer Interaction in the Blackboard (2006) or Moodle (2006) systems are contextual information about the usage of a document. Data captured here can be extracted from the properties of the courses where the documents is used; each course in Moodle, for example, has a title and description, this data can be used to extract information about the context. This data is essential to identify the different contexts where digital content is used. This element has two sub-elements:
 - **Value Type:** the *valueType* element holds a reference to an element of an ontology or taxonomy that describes the discipline as derivable from the value element above. The topics might serve as search terms to identify the appropriate discipline with online services like Swoogle (Ding Et al., 2004).
 - **Value:** the *value* element holds a free text that is extracted, for example, from the title of the course in applications like Moodle or Blackboard. It describes the topics of other documents involved in working with the recent document, e.g. the topics of all documents involved in a course. This element takes multi string value description entries. Those string entries can be used to express the same value in more than one language.

In addition to the information captured in the above two elements more contextual information can be extracted from other elements described earlier. For example, *event.dateTime*, *event.action.actionType* and *event.session.userInfo.discipline* are rich contextual information. Such data enable identifying interesting patterns about user attention given to documents. For instance, using the element *event.action.actionType* we know if the user is browsing a webpage, working with PowerPoint slides or listening to music, or may do all at the same time. This data can, for example, enable identifying the songs that the user listens to when working with MS PowerPoint and when browsing the web. Using the *event.dateTime* element it is possible to identify the music a user listens to in the morning from the music she listens to in the evening. In addition, it is also possible to identify the web pages that a user consults when working with PowerPoint slides.

- **Followed Links:** the *followedLinks* element groups the set of URIs included in the document and followed by the user. This can be a link to a relevant webpage of a document that is currently read by the user.
- **XFN:** the *xfn* element tracks the social relationship of the author of the document to the reader consulting the document, if the value of *event.action.actionType* element is read in a web browser application. In a chat tool, for example, it can also record the relationship of the user with other persons involved in a chatting event.
- **Vote Link:** the *voteLink* element records the user interest (likes and dislikes). The element can take one of the following values:
 - vote-for: means “I like the document.”
 - vote-abstain: means “I have neutral opinion.”
 - vote-against: means “I did not like the document.”

The data can be used, for example, to recommend a user with documents that are similar (of similar author for instance) to the documents that a user voted positively. Documents that are similar to the *vote-against* documents can be hidden from the user. More interestingly, this data can be used to rank the document based on the votes of a set of users. If many users voted for one specific document, this means that the document is interesting.

- **Tags:** the *tags* element holds a free text label or keywords that is used to describe the document. For example, attention metadata, user data and user tracking are valid tags for this paper.
- **Description:** the *description* element covers descriptive annotations that might be provided by users to express their experience with the document. It uses the value space of the IEEE LOM (2002) “Description” element which is a multi string value, to allow providing the same information in different languages. This element is useful to collect reviews or descriptive data about user experiences about the read item. Some users are interested in annotations other users provide to digital content.
- **Other:** The *other* element is used to allow providing customised elements that are not covered by this schema.

In the next section, we illustrate how attention metadata is collected in four applications using the CAMs schema.

4. Case Study

In this section, our case study explains how attention metadata is collected from four tools and then managed in an attention repository. The respective tools providing CAM data are Microsoft PowerPoint, Mozilla Firefox browser, WINAMP music tool and MSN Messenger. As explained in section 2, these four tools cover four categories of applications (Office Suite, Web Browsers, Multimedia Players and Computer-Mediated Communication respectively) that are used on a daily basis by users. Therefore, collecting and merging user attention from the tools enable building a rich source of information about the user attention, expressed as observations about her behaviour and interest.

Figure 3 illustrates the technical framework of this case study. For Microsoft PowerPoint, we generate attention streams directly in CAMs XML format using the ALOCOM framework (Verbert, et.al., 2005). In the other three tools, right side of figure 3, (Mozilla Firefox, WINAMP and MSN Messenger) attention streams are first stored in the local XML formats of each application and then transformed into CAMs XML instances.

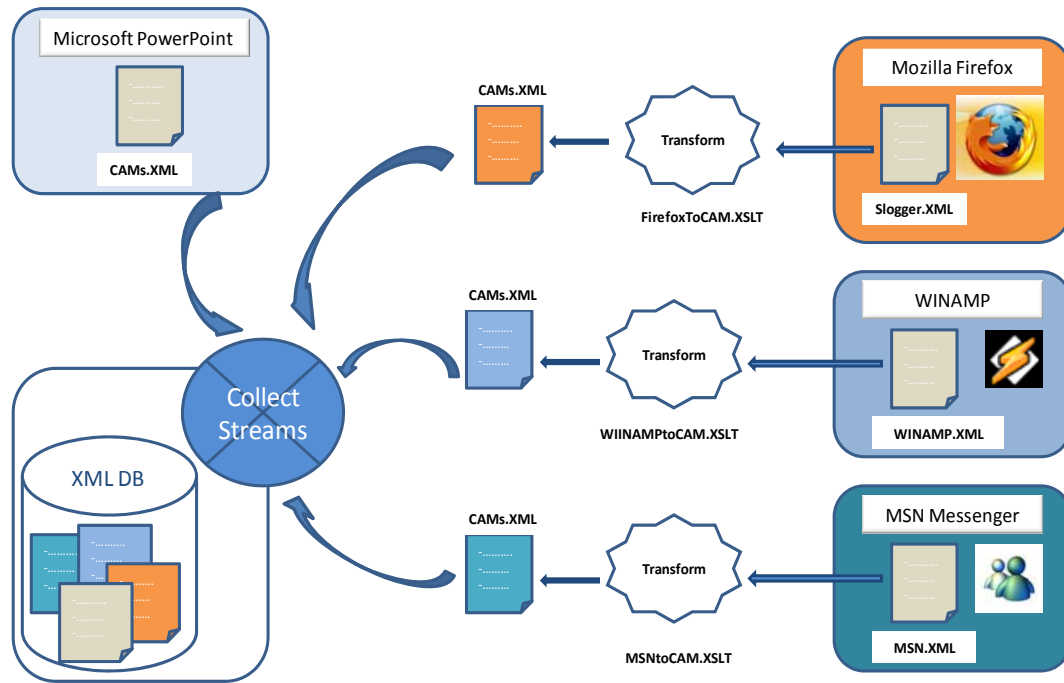


Figure 3: The framework used to collect, transform and manage attention streams

In Firefox, attention metadata is collected in the Slogger XML format, using the open source Slogger extension (Slogger, 2003). In the MSN Messenger, there is no need to install an extension to record user activities. MSN Messenger provides the tracking functionality build in: a user can simply enable or disable the logging of her messages. All user activities are logged in an MSN XML format. For WINAMP, a plug-in that was developed by (HMDB, 2006) to track user activities in WINAMP is used to collect usage information about user music activities. This data is also stored in a local XML format before transformed to CAMs.

In order to manage the attention streams in one place, we developed a component that uploads all CAMs valid streams (stored in folders on the user hard disk) into an XML database that is also installed at the client side.

In the next sub-subsections, the recording of attention metadata in each of the tools shown in figure 3 is described.

4.1 Generation of Attention Metadata in MS PowerPoint

The ALOCOM plug-in for MS PowerPoint supports capturing attention metadata from the users and the slides with which they work. The Microsoft PowerPoint .Net API (Khor, 2005) is used to extract properties from slide presentations such as the title, the author, the total editing time and the last save time, and to capture different events (open, save, print...). Other properties such as the user name and IP address are retrieved using the System.Environment and System.Net class libraries.

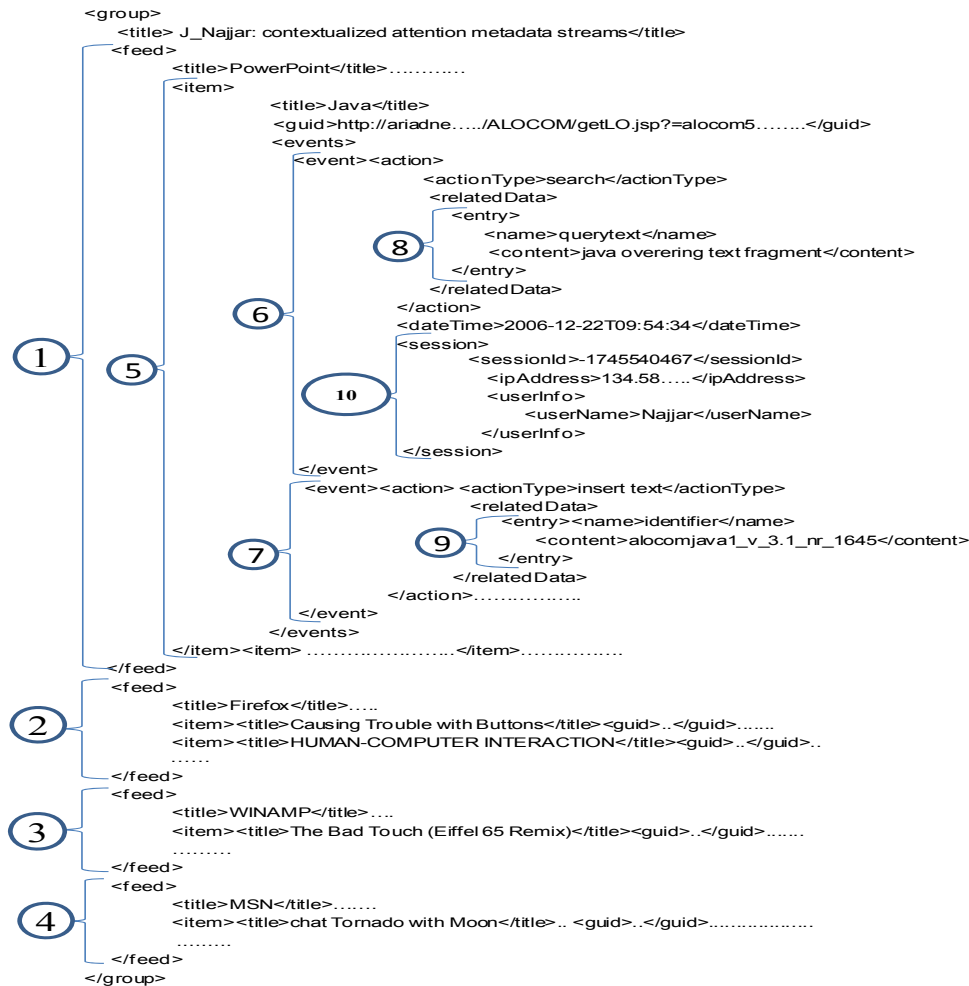


Figure 4: Part of one user CAMs instance

Furthermore, ALOCOM supports searching-for and repurposing existing documents (slides, images, text fragments, etc.). A user can specify the type of document component she is interested in (e.g. slide, diagram, table, image, text fragment), as well as keywords that best describe the component. All components that satisfy the specified search criteria are shown and the author can easily incorporate them into the presentation she is working on. Attention metadata relevant to those actions is captured directly into CAMs instances.

As shown in figure 4, all user attention information in PowerPoint is grouped in one *feed* instance (see 1 in figure 4). Data about every document (accessed slide presentation) is grouped in one *item* instance (see 5 in figure 4). For every accessed document, the following CAMs elements are captured among others:

- *group.feed.title*: the title of the presentation. The Microsoft PowerPoint API (Microsoft.Office.Interop.Powerpoint assembly) is used to retrieve this property.
- *group.feed.item.guid*: an identifier that is generated by the ALOCOM framework.

- *group.feed.item.type*: the MIME type of the document; application/powerpoint.
- *group.feed.item.lastUpdated*: the timestamp of last save time (Built in document property of the presentation document - "Last Save Time").
- *group.feed.item.event.duration*: The period of the total editing time in seconds. Its value is extracted from a built in document property of the presentation document named "Total Editing Time".

In the two events numbered 6 and 7 (shown in figure 4), attention metadata about the user interaction with the document in each event is recorded in a separate *event* instance. In the first event (6 in figure 4), the action "search" is recorded in the *actionType* element. Data related to this "search" action is stored in the *entry* element instances (see 8 in figure 4). This entry has two sub-elements; the *name* element stores the kind of the content to be stored- "querytext". The *content* element stores the real content of the data related to the search action; search terms used to search for the document. In the second event (see 7 in figure 4), the *action* element stores data about a user activity of inserting a text fragment into the slide. This text fragment is found using the ALOCOM plug-in. For this action the identifier of the document is stored in an *entry* element of *action.relatedData* element (see 9 in figure 4).

- *group.feed.item.event.session* (see 10 in figure 4):
 - *sessionIdentifier*: hash code of the time stamp
 - *ipAddress*: The System.Net API is used to retrieve this property.
 - *userName*: The System.Environment API is used to retrieve this property.
- *group.feed.item.event.dateTime*: time stamp of the event. This time is used to identify the different events that a document can be involved in.

The data recorded is data about either the document itself (e.g. *title*, *MIME type*, *guid*) or about the user (*sessionId*, *ipAddress* and *userName*). The reason for capturing the data is two-fold: to be able to recognize the documents and the users who worked with them. Furthermore, to identify and relate (using the *dateTime* and *session* information) the events where a user may have worked with each document.

4.2 Collection of Attention Metadata from the Firefox Browser

Figure 4 also presents part of an attention stream generated by using the Firefox browser. As described in the previous section, all Firefox attention streams are grouped in one *feed* element (see 2 in figure 4). This example shows the user attention given to web pages (items) she accessed while working on preparing a Human Computer Interaction lesson. The user has accessed a set of learning documents (figure 4 shows attention given to only two accessed pages). The first document is a paper about the design of buttons in interface design. The second document is a link to a book by the author of the paper.

In this context, the following elements of the CAM schema for each accessed web page are captured in the Firefox browser, using the Firefox Slogger extension. For each page (item), the following information is captured among others:

- *group.feed.item.title*: title of the accessed document (page). For example, see 2 in figure 4, the title of one webpage about the design of buttons in the interface design by Alan Dix is recorded.
- *group.feed.item.title.guid*: the primary URL of the webpage.
- *group.feed.item.dateTime*: the timestamp of access time.
- *group.feed.item.event.tags*: user tags given to the accessed webpage, free text tags in a folksonomy (Gruber, et al., 2004) approach. These tags can be provided as labels in Firefox Slogger.
- *group.feed.item.event.followedLinks*: the links to web pages (or any other documents) the user followed from the current web page; the navigated links get recorded as new items with their relevant attention data. By tracking and relating those followed links, relationship models between documents can be built. This data supports determining the user navigation and reading patterns and can be used to build relations between the accessed documents. For instance (see 2 in figure 4) while reading the paper about the design of buttons in interface design by Alan Dix, the user followed the link to Alan Dix book "HUMAN-COMPUTER INTERACTION" which is available as a reference in the webpage the user was initially busy with.

The recorded data help to determine the reading pattern of the user and to determine the topics she is interested in. This can be achieved, for example, by extracting popular keywords from titles and URLs of accessed web pages. In

addition, such keywords can be extracted from the annotations (recorded in *event.description* element) and tags (recorded in *event.tags* element) a user may attach to those pages after reading them.

4.3 Collection of Attention Metadata from WINAMP Music Application

Data about the attention given to music files are captured using an attention plug-in developed by HMDB (2006) to visualize user attention in the WINAMP application. The following elements of the CAM schema are captured (see 3 in figure 4) for each accessed music item (file).

- *group.feed.title*: title of the accessed document (music file) which is in general the name of the accessed song generated by the used music attention plug-in.
- *group.feed.guid*: the primary URI of the accessed document. In this case study, we used the combination of the song title and the artist name as an identifier for each document. Both values are generated by the used music attention plug-in.
- *group.feed.item.event.dateTime*: timestamp of the event.
- *group.feed.item.event.duration*: the time duration the document was played.
- *group.feed.item.event.action*
 - *actionType*: the type of action that was applied on the song, for example, “play” if the user played the song, or actions like “volume up” and “volume down” can also be recorded. However, the used plug-in we used in this case study does not log such detailed data.
 - *relatedData*: here, we can use this element, for example, to store data about the artist, album or the genre of the played music. In case of the volume up/down actions, this element can store the relevant values.

Data logged about user attention given to the songs she listened to is a rich source of information that can enable building a music user profile. This profile represents the user quite precisely, because it is created based on the songs she actually played and the time she spent on listening to them. For example, the user may listen to the full song in her play list (e.g., 4:23 minutes); the actual length of the song can also be retrieved using the music attention plug-in. On the other hand, the same user may listen to only the first minute of the second song on the same play list. This data enables us to determine:

- The genre of the songs the user listens to. Also, what kind of music she listens to during different periods of the day.
- The artists she likes most because she listens most of the time to their songs.
- The songs the user skips at her play lists. This can be derived from the duration the song is played and the actual length of the song.

Using the above data, it is possible to determine the genre of music the user listens to when she works on different tasks. This can be achieved by combining the music attention of one user with her attention in other applications. For example, combining the Firefox attention data with the music attention of the same user specifies the kind of music she listens to when reading a specific type of web pages, e.g. news pages. It is also possible to determine the kind of music the user listens to when she chats with her colleagues or when assembling course material in Microsoft PowerPoint or other applications.

4.4 Collection of Attention Metadata from MSN Messenger

As mentioned earlier, the attention metadata is extracted from the MSN Messenger XML log files (local format) stored on the hard disk of the user. Mapping the collected data into CAM schema is different from the previously stated examples of MS PowerPoint and the Firefox web browser. In this case we deal with contacts (users) and messages only. In the previous tools, the focus was on documents that are accessed by users who are involved in different events. Here, the focus is on users who are involved in chat events; the chat messages themselves are the documents the users interact with. Therefore, we mapped the data as follows:

- *group.feed.item* element groups all chat events that a user has with one specific user.
- *group.feed.item.event* element groups the messages that occurred in one MSN session. A session in MSN Messenger ends by the time a user closed the chat window with the other chat partner.

- Every exchanged message text is stored in a separate entry under the *event.action.relatedData.content* element. In this case, the value for *event.action.relatedData.name* is “messagetext” and the value “chat” is stored in *event.action.actionType*.

4.5 Management and Usage of Attention Metadata

So far, we discussed the collection of attention metadata from different applications. The attention streams of the four tools in this case study are merged into one XML stream using the *group* element (see figure 4). Attention metadata of each tool is represented in one feed instance. The whole CAMs XML document representing the user attention metadata is stored in an Exist database (Exist, 2006). XQuery statements are used to identify and relate the different attention streams and group them into one document which is exemplified below.

```

xquery version "1.0";
{
  for $i in 1 to 10
  let $x := distinct-values(for $r in collection('/db/CAM')/group/feed/item[title ne
    '']/events/event/datetime order by $r descending return distinct-
    values($r/../../../../guid))
  return
  <table id="{collection('/db/CAM')/group/feed/item[guid eq $x[$i]]/../../title/text()}">
    <tr>
      <td>{let $u := collection('/db/CAM')/group/feed/item[guid eq $x[$i]] return
        distinct-values(collection('/db/CAM')/group/feed/item[guid eq
          $x[$i]]/title/text())
        {let $y:=collection('/db/CAM')/group/feed/item[guid eq
          $x[$i]]/events/event/datetime return
          $y[last()]} </td>
    </tr>
  </table>
}

```

Figure 5 shows an XQuery script. A bracket labeled '1' groups the first two lines: the `for` loop and the `let` statement. A second bracket labeled '2' groups the `return` block, which contains the XML output structure.

Figure 5: An XQuery script to retrieve last 10 documents a user worked with

Figure 5 presents an XQuery example that lists the 10 last documents a user worked with in the four applications.

The script identifies the last 10 documents that a user worked with in all applications (using the *event.dateTime* element, see 1 in figure 5). It then lists the titles and access time of every relevant document (see 2 in figure 5). The query runs against the CAM collection (XML folder) where the XML CAMs document is stored.

The next step is making use of the collected rich attention metadata that tracks what a user was giving attention to and was interested in. The first obvious usage for this data is building a user attention profile that represent user actual interest based on her pervious interaction with different tools and the documents she worked with. In order to achieve this, users and documents need to be identified among different applications.

Identifying the user in this case study (desktop applications) is not a problem, since the client XML repository stores only the data of one user. Even if more users would store their attention metadata in the repository, the *group* element would allow the clear identification of each user. Here, we will consider each document with unique identifier a user worked with as a different document.

Figure 6 shows a screenshot of our simple demo tool that explores general information about user attention. The figure shows the last 10 documents (see 1 in figure 6) in the four tools that the user gave attention to; the XQuery script shown in figure 5 is used to retrieve those documents. At the left side of the tool, the keywords that represent the user interest are shown (see 2 & 3 in figure 6). Those keywords are extracted from the following elements captured about each document the user worked with in the four tools:

- Title of document.
- Descriptions a user gives to accessed documents, using the *event.description* element
- Tags a user may attach to accessed documents, using the *event.tags* element.

- Action related data, data about search terms a user used to find relevant information, artists of songs, chat text messages, etc, using the *event.action.relatedData* element.

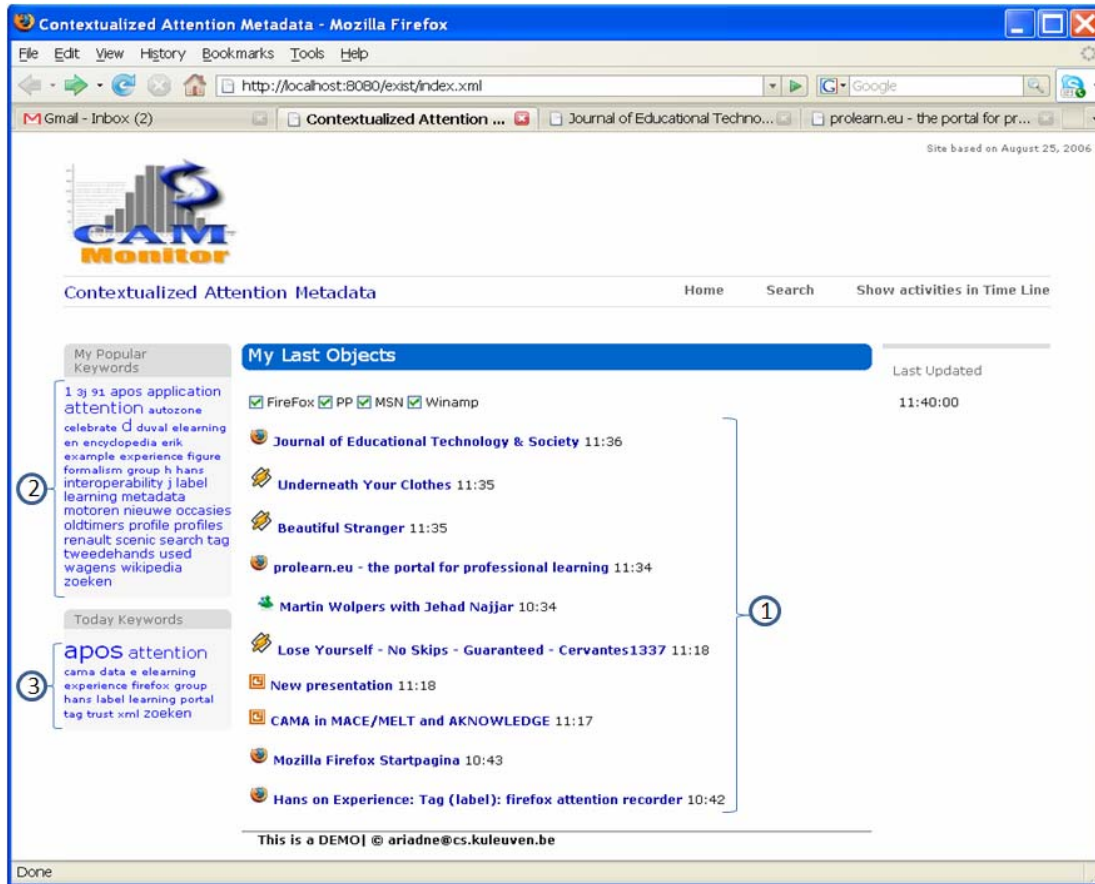


Figure 6: Screenshot of the Attention Monitor Tool

As shown in figure 6, the popular keywords represent user interest. The last documents used by the user reflect on the tasks the user was involved in. The data enables building user attention and interest profiles. Such profiles can later be related to other peoples' profiles. For example, the profile keeps information about the tools that a user frequently uses, the time spent (*duration*) in those tools and the respective documents, the datetime a user access the tools and documents (*dateTime*), the courses the user works on and the keywords that represent her interest.

As mentioned earlier the profiles of user attention represent user attention and interest across different systems. By mining the collected information from the different tools, the profile can hold information about her musical behavioural pattern, for example, “whenever a user works in Microsoft PowerPoint she listens to Classic music, while she listens to Rock music when browsing the web.” Another pattern can be “website X and Y are accessed in about 80% of the times a user starts a new slide presentation in Microsoft PowerPoint” or “Slides X and Y are reused in about 85% of user created presentations”. Such patterns can be used to alert or recommend relevant information based on the interest or task that the user is currently working on. Furthermore, such patterns enable the visualisation of user activities and attention to easily generate work reports.

5. Conclusions and Future Work

One of the main problems today is the large amount of information that teachers and learners need to deal with on a daily basis. We aim to use observations about the user and her attention to help her digesting the information flood.

This paper extends and builds on our earlier work on attention metadata. In (Najjar, et al., 2005), a simple conceptual attention metadata framework that uses the AttentionXML schema to capture attention metadata in learning systems is proposed. In (Najjar, et. al., 2006a), an extension for AttentionXML schema is proposed, to enable collection attention metadata from different sources. In (Najjar, et. al., 2006b), the use of attention metadata for technology enhanced learning was discussed. In (Wolpers, et. al., 2006), the use of attention metadata in knowledge management systems is discussed. In this paper, we present the stable version of the schema, which enables collection of contextualized attention metadata across systems boundaries, with a detailed explanation of its components and their use as well as a sample application demonstrating first correlation examples.

We start with the collection of contextualized attention metadata that describes in detail in which context a user spends her attention on which content. We therefore propose a conceptual framework and the contextualized attention metadata schema that enables the recording and management of rich and detailed sets of data about user attention given to documents in different applications. As a case study, we generate attention metadata from a number of desktop applications like MS PowerPoint, WINAMP music player, MSN Messenger and Firefox browser and show in a demo application a time-based sequence of activities. Such correlations of contextualized attention already indicate how useful its explanation can be for specific users; e.g. for personalization and information management in the learning context.

Future work includes the collection of more attention metadata from several applications to form a rich and large base of attention metadata from users. Based on this repository, we will develop further algorithms to extract knowledge about the user, the context and the content. A simple example is statistical data about the usage of learning documents. Advanced examples include the identification and extraction of patterns of behaviour, e.g. through the correlation of activities carried out by one user and related to one context and/or one content. In addition, the clustering of users based on observations about their behaviour in certain contexts provides information about the sequences of activities carried out by users. Such clusters would then allow the generation of recommendations of similar users, eventually combining them with the behavioural patterns to more precisely identify next steps. Ultimately, behavioural patterns and similarity measures among users can be used to detect the finalisation of user goals and, subsequently, the identification of user aims and goals.

In parallel to the collection of large amounts of attention metadata, an empirical evaluation study will be conducted to determine the usefulness and efficiency of attention metadata in facilitating the user tasks, and in enhancing user experience with the tools she works with. However, the evaluation of such systems is quite challenging because attention metadata is about detailed information about user behaviour and interest. Therefore, the evaluation will be designed to determine the efficiency of the contextualized attention metadata approach, the used schema and framework. Furthermore, we will conduct research in applying existing solutions to the privacy and security issues at hand when dealing with this highly personal data. An example of our evaluation approach is the determination of the difference in task completion, before and after using attention metadata -supported systems. In addition, interviews and questionnaires may be used to determine the user satisfaction with the attention-supported systems; like comparing information retrieval attention-supported systems against current systems.

The CAM schema presented in this paper is now being implemented in a number of European projects like MACE (2006), MELT (2006) and AKNOWLEDGE (2006). The schema will be used to facilitate the collection, management and exchange of user attention metadata form different tools in those projects. By applying the schema in these projects, we will be able to evaluate our approach and identify further research questions related to contextualized attention metadata.

6. References

AKNOWLEDGE (2006). *Accessible & Open Knowledge Infrastructure for Flanders*, Retrieved June 8, 2007, from <https://projects.ibbt.be/acknowledge/>.

Aposdle (2006). *EU project APOSDLE*, Retrieved June 8, 2007, from <http://www.aposdle.tugraz.at/>.

AttentionXML (2004). *AttentionXML specifications*, Retrieved June 8, 2007, from <http://developers.technorati.com/wiki/attentionxml>.

Braun, S., & Schmidt, A. (2006). Don't annoy the Informal Teacher: Context-Aware Mediation of communication for Workplace Learning, *Paper presented at the 6th International Conference on Knowledge Management*, September 6-8, 2006, Graz, Austria.

Broisin, J., Vidal, P., & Duval, E. (2005). A Model driven Approach for the Management of Learning Object Life Cycle. *Paper presented at the Intelligent, Interactive, Learning Object Repositories Network (I2LOR)*, November 14-18, 2005, Vancouver, Canada.

Cardinaels, K., Meire, M., & Duval, E. (2005). Automating Metadata Generation: the Simple Indexing Interface. *Paper presented at the International World Wide Web Conference*, May 10-14, 2005, Chiba, Japan.

Chirita P. A., Firan, S. C., & Nejdil, W. (2006). Summarizing Local Context to Personalize Global Web Search. *Paper presented at the ACM Conference on Information and Knowledge Management (CIKM)*, November 6-11, 2006, Arlington, Virginia, USA.

Ding, L., Finin, T., Joshi, A., Pan, R., Cost, R. S., Peng, Y., Reddivari, P., Doshi, V., & Sachs, J. (2004). Swoogle: A Search and Metadata Engine for the Semantic Web. *Paper presented at the ACM Conference on Information and Knowledge Management (CIKM)*, November 8-13, 2004, Washington D.C., USA.

Duval, E., & Hodgins, W. (2003). A LOM Research Agenda. *Paper presented at the 12th International Conference on World Wide Web*, 20-24 May 2003, Budapest, Hungary.

Exist (2006). *Exist open source database*, Retrieved June 8, 2007, from <http://exist.sourceforge.net/>.

Garofalakis, J., Giannakoudi, T., & Sakkopoulos, E. (2006). Semantic Web Site Usage Analysis: The ORGAN System. *Paper presented at the World Wide Web 2006 Workshop Logging Traces of Web Activity: The Mechanics of Data Collection*, May 23, 2006, Edinburgh, UK.

Gonçalves, M., Panchanathan, G., Ravindranathan, U., Krowne, A., Fox, A. E., Jagodzinski, F., & Cassel, L. (2003). The XML Log Standard for Digital Libraries: Analysis, Evolution, and Deployment. *Paper presented at the Third Joint Conference in Digital Libraries*, May 27-31, 2003, Houston, Texas.

Gruber, T. (2005). Folksonomy of Ontology: A Mash-up of Apples and Oranges. *Paper presented at the First on-Line conference on Metadata and Semantics Research (MTSR'05)*. Retrieved June 8, 2007, from <http://tomgruber.org/writing/ontology-of-folksonomy.htm>.

Henze, N., & Nejdil, W. (2003). Logically Characterizing Adaptive Educational Hypermedia Systems. *Paper presented at the workshop on Adaptive Hypermedia and Adaptive Web-Based Systems, Session at WWW2003 Conference*, May 20, 2003, Budapest, Hungary.

HMDB (2006). *Case Study on Visualization of AttentionXML*, Retrieved June 8, 2007, from <http://ariadne.cs.kuleuven.be/infovis/casestudies.php#attention>.

Holz, H., Rostanin, O., Dengel, A., Suzuki, T., Maeda, K., Kanasaki, K. (2006). Task-based Process Know-how Reuse and Proactive Information Delivery in TaskNavigator. *Paper presented at the ACM Conference on Information and Knowledge Management (CIKM) 2006*, November 6-11, 2006, Arlington, VA, USA.

IEEE LOM (2002). *IEEE Learning Object Metadata*, Retrieved June 8, 2007, from <http://ltsc.ieee.org/wg12/>.

Jones, S., Cunningham, J. S., McNab, J. R., Boddie, J. S. (2000). A transaction log analysis of a digital library. *International Journal on Digital Libraries*, 3(2), 152-169.

Khor, M. S., & Leonard, A. (2005). *Installing and Using the Office 2003 Primary Interop Assemblies, Microsoft Office 2003 Technical Articles*, Retrieved June 8, 2007, from [http://msdn2.microsoft.com/en-us/library/aa159923\(office.11\).aspx](http://msdn2.microsoft.com/en-us/library/aa159923(office.11).aspx).

Lindstaedt, S., Ley, T., & Ulbrich, A. (2006). APOSDLE – New ways to work, learn and collaborate. *Paper presented at the 6th International Conference on Knowledge Management*, September 6-8, 2006, Graz, Austria.

- MACE (2006). *Metadata for Architectural Contents in Europe*, Retrieved June 8, 2007, from <http://www.mace-project.eu/>.
- MELT (2006). *Metadata Ecology for Learning and Teaching*, Retrieved June 8, 2007, from http://info.melt-project.eu/ww/en/pub/melt_project/about.htm.
- Moodle. (2006). *Moodle Course Management System (CMS)*, Retrieved June 8, 2007, from <http://moodle.org/>.
- Najjar, J., Meire, M., & Duval, E. (2005). Attention Metadata Management: Tracking the Use of Learning Objects through Attention.XML. *Paper presented at the ED-MEDIA conference*, June 27-July 2, 2005, Montreal, Canada.
- Najjar, J., Ternier, S., & Duval, E. (2004). User Behavior in Learning Objects Repositories: An Empirical Analysis. *Paper presented at the ED-MEDIA conference*, June 21-26, 2004, Lugano, Switzerland.
- Najjar, J., Wolpers, M., & Duval, E. (2006a), Attention Metadata: Collection and Management. *Paper presented at the World Wide Web 2006 Workshop Logging Traces of Web Activity: The Mechanics of Data Collection*, May 23, 2006, Edinburgh, UK.
- Najjar, J., Wolpers, M., & Duval, E. (2006b). Towards Effective Usage-Based Learning Applications: Track and Learn from User Experience(s). *Paper presented at the IEEE International Conference on Advanced Learning Technologies (ICALT 2006)*, July 5-7, 2006, Kerkrade, The Netherlands.
- Nepomuk (2006). *Nepomuk Semantic desktop*, Retrieved June 8, 2007, from <http://nepomuk.semanticdesktop.org>.
- Neven, F., Duval, E., Ternier, S., Cardinaels, K., & Vandepitte, P. (2003). An Open and Flexible Indexation and Query tool for ARIADNE. *Paper presented at the ED-MEDIA conference*, June 23-28, 2003, Honolulu, Hawaii.
- Roda C., & Thomas J. (2006). Attention Aware Systems: Theories, Application and Research Agenda. *Journal on Computers in Human Behaviour*, 22, 557-587.
- Sauermann, L. (2006). The Semantic Desktop - a Basis for Personal Knowledge Management. *Paper presented at the 5th International Conference on Knowledge Management*, June 29 – July 1, 2005, Graz, Austria.
- Second Life (2006). *Second Life Viewer*, Retrieved June 8, 2007, from <http://www.secondlife.com>.
- Slogger (2003). *Mozilla Firefox Extension: Log Browsing History*, Retrieved June 8, 2007, from <https://addons.mozilla.org/firefox/143/>.
- Verbert, K., Jovanovic, J., Gasevic, D., & Duval, E. (2005). Repurposing Learning Object Components. *Paper presented at the OTM 2005 Workshop on Ontologies, Semantics and E-learning*, November 3, 2005, Agia Napa, Cyprus.
- Völkel, M., & Haller, H. (2006). Conceptual Data Structures (CDS) -- Towards an Ontology for Semi-Formal Articulation of Personal Knowledge. *Paper presented at the 14th International Conference on Conceptual Structures*, July 16-21, 2006, Aalborg, Denmark.
- Weinreich, H., Obendorf, H., & Herder, E. (2006a). Data Cleaning Methods for Client and Proxy Logs. *Paper presented at the World Wide Web 2006 Workshop Logging Traces of Web Activity: The Mechanics of Data Collection*, May 23, 2006, Edinburgh, UK.
- Weinreich, H., Obendorf, H., Herder, E., & Mayer, M. (2006b). Off the Beaten Tracks: Exploring Three Aspects of Web Navigation. *Paper presented at the World Wide Web conference 2006*, May 23-26, 2006, Edinburgh, UK.
- Wolpers, W., Martin, G., Najjar, J., & Duval, E. (2006). Attention Metadata in Knowledge and Learning Management. *Paper presented at the 6th International Conference on Knowledge Management*, September 6-8, 2006, Graz, Austria.