2011

Semantically Reconnecting Fragmented Information through User Activity Monitoring

Hinnerk Brügmann
University of Erlangen-Nuremberg, hinnerk.bruegmann@wiso.uni-erlangen.de

Matthias Kurz
University of Erlangen-Nuremberg, matthias.kurz@wiso.uni-erlangen.de

University of Erlangen-Nuremberg

Semantically Reconnecting Fragmented Information through User Activity Monitoring

Hinnerk Brügmann
Institute of Information Systems II
University of Erlangen-Nuremberg
Nuremberg, Germany
hinnerk.bruegmann@wiso.uni-erlangen.de

Matthias Kurz Institute of Information Systems II
University of Erlangen-Nuremberg
Nuremberg, Germany
matthias.kurz@wiso.uni-erlangen.de

Dieter Sauer Institute of Information Systems II
University of Erlangen-Nuremberg
Nuremberg, Germany

ABSTRACT

Today information items on user's workstations are usually stored in separate collections depending on their format. This results in a disconnect between information systems and user needs leading to high lookup times during task related information retrieval. This paper presents an approach to reduce document based information fragmentation by semantically reconnecting electronic documents to each other without imposing additional training or tagging workload on the user. To this end the actions knowledge workers perform on their desktop are transparently monitored to analyze the user's interaction with his computer system. These action metadata are further clustered by superordinate activities performed by the user. Finally documents attached to window instances within the identified activity clusters are semantically to each other related reducing the fragmentation of their contained information. This allows a subsequent associative information discovery navigating from one document instance to other related document instances. A prototypical implementation and evaluation in a small scale testing setup indicates the validity of the approach.

1. INTRODUCTION

Due to the high importance computer-mediated communication and IT systems have in the execution of today's business processes, enterprises are a main driver in the overall growth of information by generating a rapidly increasing amount of information as by-product of their business activities [38]. Lyman and Varian estimate the annual growth of newly created information in enterprises to be 30% [25]. According to studies of Merrill Lynch and Ferris Research only 20% - 40% of this information is stored in a structured, semantically described form in electronic databases or structured business applications (e.g. SAP R/3) [13, 5]. The other 60% - 80% are contained in the unstructured form of electronic documents[1] (e.g., Microsoft Office files, e-mails, images, multimedia files, or web based content). This is even more dramatic as the annual growth rate of newly created unstructured content is considerably higher than that of structured information.

Historically unstructured information was and in most scenarios still is managed in an application specific way. Due to different business requirements, enterprises employ task-specific applications which leads to the storage and processing of inherently similar information in different places in the enterprise information architecture. A single task or workflow will therefore require the use of multiple applications resulting in different interaction techniques and information representations to support the user [27]. Tungare et al. labeled this situation *Information Fragmentation* where a user's data are tied to different formats, distributed across multiple locations, manipulated by different applications and reside in a generally disconnected manner [34]. Most often information items are stored in separate collections depending on their formats: documents are saved in a documents' folder hierarchy (e.g. in the My Documents folder), e-mails in a separate mailbox hierarchy, and bookmarks to favorite web sites in another browser hierarchy [19]. This hierarchy separation has several negative outcomes: It leads to potential redundancy and an increased required effort to locate document based knowledge. In addition to being time consuming, managing three (or more) different hierarchies generates cognitive load when trying to maintain a certain degree of consistency between the hierarchies and in using multiple different applications with inconsistent interaction designs [20]. Since users associate information objects with their projects and tasks rather than document formats [4] this represents a potential disconnect between information systems and a user's needs.

Such problems multiply in a multi user enterprise environment with numerous competing repositories, document management systems, shared network drives, and local file systems for each distinctive user. On one hand a persons ability to overlook the available number of information

[1]In the following the term document is understood as relating to electronic files containing unstructured information.

10^{th} International Conference on Wirtschaftsinformatik,
$16^{th} - 18^{th}$ February 2011, Zurich, Switzerland

sources diminishes as the amount of documents in numerous different repositories grows. On the other hand a multi user environment with local file system repositories lacks the central coordinating instance of an individual user [29]. As a result individual knowledge workers in such multi user enterprise environments spend an even greater part of their working hours looking for the correct and most up-to-date information needed in workflow steps or tasks.

The research contribution presented in this paper is a method to reduce document based information fragmentation by semantically reconnecting electronic documents to each other without imposing additional training or tagging workload on the user. To this end the actions knowledge workers perform on their desktop are transparently monitored to analyze the user's interaction with his computer system. These action metadata are further clustered by superordinate activities performed by the user. Actions are clustered if their associated desktop operations can be related to each other by combining multiple relationship building algorithms. Finally documents attached to window instances within the identified activity clusters are semantically related to each other reducing the fragmentation of their contained information.

The paper is organized as follows: Section 2 presents a review of current research. This is followed by the description of the presented approach in section 3 consisting of the capturing of semantic user activity context, the subsequent clustering of these activity contexts, and the final inference of semantic interrelations between documents. Section 4 shows the results of evaluating a prototypical implementation of the approach in a small scale testing setup. The paper is concluded with a summary and an outlook to ongoing research and open questions.

2. RELATED WORK

Previous research approaches to establish and utilize relationships between documents can be separated into two groups. The first group makes use of the content of documents to detect similarity or references in documents, while the second group of work relies on analyzing user activity involving documents usage to infer interrelations.

While some techniques applied by the first group of approaches (e.g. Named Entity Recognition) are already quite advanced, more sophisticated parsing of document content (e.g. Natural Language Processing) is still problematic due to high manual training requirements of the applied algorithms [8]. Additionally content based approaches are often limited to text documents (e.g., multimedia files or files in proprietary formats) [6].

The second group of approaches focuses on the activities of a user around individual documents. Relying on tasks as the principle means of document relationship discovery requires identifying the connections of one piece of information in one application to task-related information in the same or other applications. The project *Stuff I've Seen* [12] focuses on document metadata and in part document content to create a context-snapshot of the time when a document was accessed. A user might then at a later time query the *Stuff I've Seen* datastore with arbitrary keywords to be presented with a list of documents he accessed earlier. The research prototype *Phlat* [9] combines keyword and property-value search with faceted browsing functionality. The tagging of documents which serves as the basis for *Phlat*'s superimposed search UI has to be manually performed by the user though. Both *Stuff I've Seen* as well as *Phlat* aim at enhancing information retrieval but do not reconnect previously fragmented information contained in different documents. The task-centered tools *UMEA* [21], *TaskMaster* [2] and *TaskTracer* [11, 33] address this to some degree by capturing the desktop context of project or task related user actions. All three approaches require the user to manually enter the specific task or project on which he is working at any current time. This necessity is avoided by the IBM prototype *Activity Explorer* which makes use of document-centric user activities within a collaborative document management system to add relations between document instances [14]. This approach relies on a specifically customized DMS environment though and does not detect user activity outside this proprietary work context.

The research projects *SWISH* [28], *Smart Desktop* [24], *Dyonipos* [17], *APOSDLE* [23], and *UICO* [31] are all engaged in automated task prediction and task switch detection within personal desktop environments. The first four approaches rely on textual metadata of surveyed desktop resources to distinguish task contexts. On top of that the UICO prototype also takes chronological metadata of contextual activities into account. While all five prototypes show high detection accuracy this comes at the cost of initially training the employed machine learning mechanisms to specific application domains [30]. Besides this initial training UICO requires an additional step to model the ontology classes.

A limiting factor to the adoption of Knowledge Management Systems (KMS) is often seen in lacking user acceptance [26]. Steve Bailey (Senior Adviser for Records Management issues at JISC infoNet) summarizes the issues employees have with the overhead generated by IT systems to improve the management of unstructured content [1]: *"As far as the average user is concerned, the EDRMS [(Electronic Document and Records Management System)] is something they didn't want, don't like and can't use. As such, its no wonder that so few users accept them. As one person once said to me 'making me use an EDRMS is like asking a plasterer to use a hammer!"* This low user acceptance results mostly from knowledge workers feeling burdened by additionally workload caused by e.g., manual tagging or categorization in KMS. Incidentally the described approaches to the management of unstructured information require some initial *knowledge investment* before a critical mass is reached which can be used to endow user benefit (e.g. a DMS only showing relevant search results after a certain amount of manual tagging on contained documents has been performed). The additional effort is felt by the employees to be distracting from the subjectively more important *real work* and is therefore often circumvented or altogether ignored [15].

3. CONSENSE APPROACH

The presented approach is set within the encompassing research context of the ConSense project[2] which aims to gather document related metadata within enterprise environments to deduct semantic relations between documents and business domain specific entities (e.g., documents, persons, projects or products) to allow a semantic exploration of the enterprise information landscape [6]. The presented approach forms one of several pillars to capture semantic metadata on client workstations. While reasoning within the semantic network is a crucial part of the encompassing project it occurs at a later stage after sensor data has been gathered and aggregated[3] and is therefore not in the focus of this paper.

3.1 Semantic Activity Context

According to Kuutti, user activities can be considered as having three hierarchical levels: activity, action, and operation, which correspond to motive, goal, and conditions of the task for which the activity is performed [22]. An activity may be achieved through a variety of actions. Similarly, operations may contribute to a variety of actions. In the following a task is regarded as containing one or more distinctive activities which in turn consists of various actions which all relate to the same object or motive (see figure 1). Actions in turn consist of any number of (desktop) operations with an action always having a particular goal within the context of the task-induced activity motive [35]. As will be shown for the purpose of the presented approach it is sufficient to focus on the lower three levels of activity, action, and operation to establish relations between documents containing fragmented information. The encompassing level of tasks spanning potentially multiple days is therefore not in further scope.

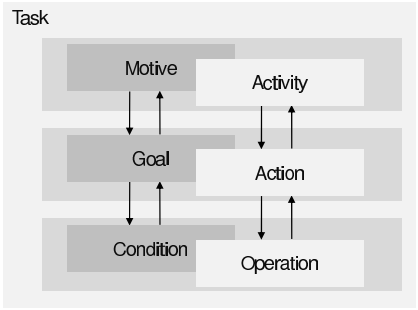

Figure 1: Tasks, activities, actions, and operations [22]

When evaluating user desktop activity each corresponding action has a context consisting of the sum of all other actions taken by a user as well as the respective workplace environment during the user action. The individual elements comprising such context can be separated into two dimensions: time and scope. The dimension of time is split into context elements which happened before, during, and after the user action. The dimension of scope differentiates among user actions targeting a specific document and the general desktop environment visible to the user. This desktop environment in the context of a document access consists of all opened and visible desktop and web applications as well as all content and documents contained in these applications.

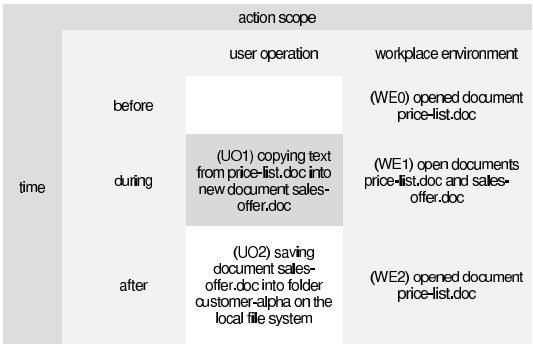

Figure 2: Exemplary context of electronic document access

To give a simple business scenario example: A user in the sales department copies some textual content from document PRICE-LIST.DOC into a new document SALESOFFER.DOC and saves it in the local file system folder CUSTOMER-ALPHA. Figure 2 shows the context of this action. The user operation (in this case copying) allows the assumption (not certain knowledge) of an existing semantic relation between the documents PRICE-LIST.DOC and SALES-OFFER.DOC. The metadata gained from observing user operation UO1 might further be influenced by the corresponding workplace environment WE1. So could for example a window positioning showing both documents next to each other without overlap strengthen the assumption of an existing relationship. Part of the action context is the second user operation UO2. This in turn can be used to establish a new (or strengthen an existing) semantic relation of SALES-OFFER.DOC and (to a lesser extend) PRICE-LIST.DOC to other documents contained in folder CUSTOMER-ALPHA. From a modeling perspective UO2 has its own context, which would then contain UO1 as an operation that happened shortly before UO2.

3.2 Capturing Semantic Activity Context

For the purpose of capturing the semantic context of user actions a client-side sensor plugin is installed on a user's workstation. The sensor takes a snapshot of window metadata whenever changes are occurring within the workspace environment either manually triggered by the user (e.g., opening or moving window instances) or automatically triggered by the system (e.g., a notification window being shown). Events being considered changes to the workplace environment are changes to a window's X/Y/Z position as well as to its size. Additional snapshots are taken upon user operations triggering new documents being opened within these window instances. Table 1 lists the recorded metadata for each window during a monitoring snapshot.

With multi monitor setups becoming more common [10] the sensor detects and logs changes in the workstation's monitor configuration including the monitor's resolution, physical position in relation to other monitors and the start and end

[2] www.consense-project.com

[3] See [6] on the subjects of reasoning as well as dealing with computational complexity in the ConSense project.

Table 1: Extracted basic window metadata

Data type	Metadata
int	X position of the top left corner
int	X position of the top left corner
int	Z axis position
int	Monitor id on which the top left corner is positioned
int	Height
int	Width
string	Unique window identifier
string	Unique parent window identifier
string	Window title
bool	Indicator if the window has UI focus

time of this specific setup. This allows the later calculation of the available screen estate to the user for any given time.

In order to analyze these raw action metadata effectively they are transformed into a hierarchical, chronologically ordered, data structure as depicted in figure 3. The main object *Working Item* contains all workspace related metadata between the start and end of a user session as indicated by the operation system's session start and session end events[4]. Within a *Working Item* window metadata for each snapshot are contained within a *Time Span* describing periods during which no changes to the workplace environment occurred. These *Time Spans* contain the window metadata of all windows visible to the user during the duration of the *Time Span*. Visibility in this regard is determined by calculating the relative square size to which a window is visible to the user using the window's X/Y/Z position and size metadata. Due to study results indicating a user preference to perform a high percentage of window switching on one primary monitor [18] windows being displayed on a secondary monitor as indicated by the operating system have their relative visibility reduced by 50%. For windows stretching across monitor boundaries this reduction is applied according to their partial visibility on secondary monitors. Only metadata for windows having a computed relative visibility higher than 0.2[5] are included in the *Time Stamp* recordings.

The raw window metadata captured in this way offer only limited leverage to identify substantive relationships between fragmented document based information. For this reason it is enriched by metadata relating to the respective application responsible for the window's creation as well as relating to document(s) opened within the window instances. Existing studies regarding the usage of document and application types during typical task execution patterns [12, 9] show e-mail as the most common type opened (~75%), followed by web pages (~15%), Microsoft Word files (~6%), Microsoft Powerpoint files (~3%), and various media types including pictures and audio files (~4%). Over 6% of all

[4]The logical session start and session end events are comprised of the physical start/stop, sleep/hibernate/stand by/wake up, user login/logout and remote connection initiated/terminated events.

[5]According to [18] windows hidden by more than ~75% do not impart any informational value to the user and are therefore treated as having no direct relation to the immediate action context.

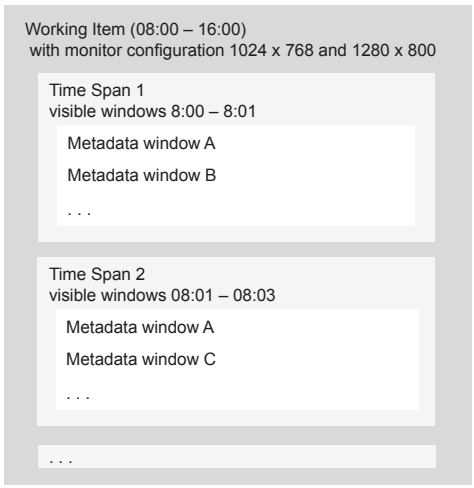

Figure 3: Structure of captured workspace metadata

items were e-mail attachments. Accordingly the sensor plugin's metadata extraction routines focus on the Microsoft Office Suite as well as the major web browser families Microsoft Internet Explorer and Mozilla Firefox. In particular the Microsoft Outlook element types e-mail, calendar entry, contact item, and files attached to these elements are included in the sensor range. Table 2 lists the (upon availability) recorded additional application and document metadata for each window during a monitoring snapshot.

Table 2: Extracted extended window metadata

Data type	Metadata
string	Unique id of the application responsible for the creation of the window
string	Name of the application responsible for the creation of the window
string	System path of the application responsible for the creation of the window
string[]	System path(s) of opened file based document(s)
string[]	Unique id(s) of opened Microsoft Outlook element(s)
string[]	Web page URI(s) opened in web browser tab(s)

Three categories of documents are treated in more depth using different content extraction approaches. These categories are web based HTML documents, elements within Microsoft Office Outlook, and lastly all other documents. In the case of web content it is first checked whether the target is the web address of a HTML file. If this is the case the sensor accesses the web browser tab's HTML source and temporarily stores it on the local machine. In the case of a Microsoft Outlook element the unique Outlook id initially extracted is used to capture further metadata for these elements. Additionally the respective Outlook element is evaluated regarding attached files. If found these are recursively processed in the same way as the originally identified documents. Lastly for the content extraction from other files the indexing service inherently contained within the Microsoft

Windows operation system is used to access and extract document metadata. If the file system in question is of type New Technology File System (NTFS), which is commonplace in modern Microsoft Windows operating systems, additional metadata from the file system are available for extraction by the sensor plugin.

Because the naming conventions for different types of metadata attached to files vary depending on the respective application responsible for the creation of the specific file types, a common semantic as well as syntactic metadata description is necessary. This underlying metadata description is provided by the ConSense ontology store [6] containing both custom ontology descriptions in the Web Ontology Language (OWL) format as well as a subset of the Friend-of-a-Friend (FOAF) and Dublin Core ontologies. In the case of Dublin Core the original RDF Schema notation has been transformed into OWL mapping *rdf:Property* predicate types to *owl:AnnotationProperty* types. This allows the resulting Dublin Core OWL to retain an expressivity of OWL-DL compared to the computationally more complex OWL-Full superset. The combined expressivity of these ontologies stays within OWL-DL corresponding to $\mathcal{SHOIN}^{(\mathcal{D})}$ using description logic naming conventions.

This ontology set forms the ground for a static mapping of raw file attributes to the gathered (RDF encoded) semantic metadata. Additionally the ontology set accessed by the sensor plugin could be further extended to account for e.g., application types beyond the abovely mentioned Microsoft Office suite. The sensor plugin validates gathered context input against the common ontologies (without resorting to expensive inference reasoning) and persists it in the form of semantic networks in a local Resource Description Framework (RDF) triple-store.

3.3 Clustering Semantic Activity Context

Utilizing the semantic metadata described in section 3.2 to relate fragmented document based information to each other is limited by the underlying assumption that the whole captured context belongs to a single user activity. Due to multitasking behavior being predominant within all kinds of office environments and scenarios though this does not hold true for most real world applications. Knowledge workers spend a great deal of time engaged in multiple tasks at the same time [32]. So do for example employees of an information-technology company spent an average of only 3 minutes per task before switching to another task [16]. As Salvucci et al. point out there exist multiple types of multitasking. On one hand there is *concurrent multitasking* where a person changes between different tasks back and forth several times a second giving the impression that they are performing the different tasks simultaneously. On the other hand there exists *sequential multitasking* where a task is carried out over minutes or hours until a secondary task interrupting the first (e.g., a person reading a book while cooking, briefly stirring the sauce, and then reading again for several minutes in the book). The basic procedure of human multitasking is always the same regardless of the kind of multitasking though as shown in figure 4. An important point in changing from one task to the next is the time delay which can take quite long depending on the complexity of the task. In the context of the presented approach a special subset of multitasking is of particular interest, namely *intermittent processing* [3]. Within this scenario several tasks are executed on a computer system in parallel though the user can put his attention only on one activity at any time resulting in a de facto sequential multitasking pattern.

Accordingly the sensor plugin unravels the previously captured semantic window action metadata and clusters them according to their superordinate activities. To this end multiple algorithms are employed to establish semantic relationships between the observed window instances. Initially, the simplest type of relationship is considered, namely whether a window instance can be identified as existing across multiple *Time Spans* using the unique window identifier as well as positioning metadata. The second scenario being resolved by the sensor plugin is the existence of multiple window instances visible within the same *Time Spans* belonging to the same application. Such situations often occur when dialog boxes (e.g. the application asking for confirmation of a specific user action) are displayed by an application. This is confirmed if the captured metadata show one of the window instances containing as parent id the unique handle of the other. In both scenarios a semantic statement of predicate *belongsToSameActivity* interrelating the window instances is added to the semantic network.

In the next step relationships between windows of different applications are uncovered which indicate user operations within the same action and therefore the same superordinate activity. This is initially done for all windows within a *Time Span* using the extracted textual metadata both from the window instances as well as from their associated documents (opened and attached). As the starting point for such a comparison the *title* metadata of windows and documents is utilized which is then enriched by the textual metadata of contained or attached documents.

When employing a basic string comparison algorithm (e.g. Jaro-Winkler distance [37]) different spelling of keywords or the presence of textual clutter can taint the relevance of resulting similarity indices as illustrated in the following two exemplary window titles (1) *The term Business Activity Monitoring (BAM) describes a collection of analyses and presentations - Windows Internet Explorer* and (2) *C:/Interest/Analyses/BAM - Windows Explorer*.

In order to increase the relevance of further processing results, the sensor plugin sanitizes the textual metadata. In a first step the application name is stripped from the window title if it is contained[7]. In the case of more extensive textual metadata as shown in (1) the removal of the application name alone is not sufficient. As the string contains a full English sentence it comes with the usual language specific expletives. To this end the sensor plugin normalizes spelling variations using the Porter Stemming algorithm [36] and separates compound strings into a list of individual words. Subsequently further splitting is performed on dashes to separate system paths into single directory name

[6]Hosted at http://ontology.consense-project.com

[7]Within the Microsoft Windows operation system each application registers its own name within the registry hive.

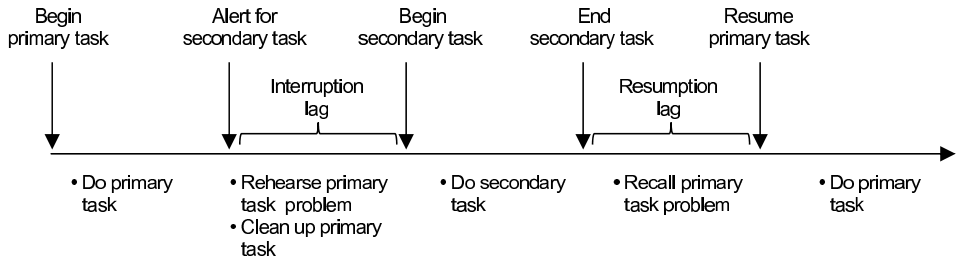

Figure 4: Multitasking flow [32]

parts to account for cases as illustrated in example (2). The resulting list of normalized strings is then cleaned off language specific stop words. As mentioned in section 1 the presented approach is set to avoid task interruption to the knowledge worker making a manual selection of language contexts suboptimal. To this end the sensor plugin employs a generic n-gram comparison which utilizes word frequency classes of the textual metadata to determine the language of a given (partial) text. According to the detected language, stop words are then removed from the word list using provided static stop word lists for the languages Dutch, English, French, Spanish, Italian, and German.

Finally the word lists for two window instances are compared using a Weighted Tag Similarity algorithm [7] resulting in a numeric value in the interval [0,1] indicating the assumed similarity of the two windows. To reduce the necessary amount of background processing window instances are only compared if their respective *Time Spans* lie within 10 minutes of each other.

Any *belongsToSameActivity* relation concluded by this similarity value can only be treated as an assumption with a certain amount of unreliability. To this end a semantic reification statement is added to the *belongsToSameActivity* relation indicating its reliability as calculated from the similarity value using a fuzzifying function (see figure 5):

$$rel = \begin{cases} 2sim^2 & sim < 0.5 \\ 4sim(1 - \frac{sim}{2}) - 1 & sim \geq 0.5 \end{cases}$$

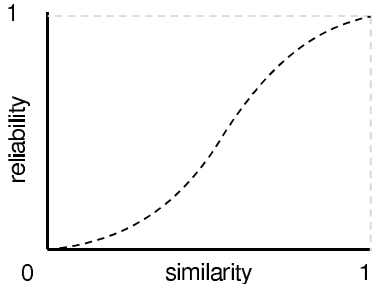

Figure 5: Fuzzification of textual similarity to reliability of statement

Figure 6 shows an exemplary semantic network of activity cluster relations being generated by evaluating window instances across two adjacent *Time Spans*. Lastly the previously generated relationships between window instances for all *Time Spans* within the current *Working Item* are condensed into a single clustering graph as depicted in figure 7 forming the final activity clusters. The reified reliability is shown on the graph edges.

3.4 Relating Documents based on Activity Clusters

Based on the identified activity clusters the sensor plugin relates documents associated to window instances within the same cluster. To this end a weighted degree centrality C for each window instance w is calculated[8] with W being all window instances, and $rel\,(w, w')$ being the attached reliability of the *belongsToSameActivity* relation between windows w and w':

$$W'_w = \{w' \in W : \exists rel\,(w, w')\}$$

$$C_w = \frac{\sum_{w' \in W} (rel\,(w, w'))}{|W| - 1} |W'_w|$$

For every two window instances p and q within the activity cluster a semantic statement with predicate *hasRelationTo* is inserted into the semantic network. Additionally a reification statement signifying the reliability of the *hasRelationTo* statement with value $C(p) * C(q)$ is added.

Finally in addition to the reliability further reification statements are added containing provenance information to the statements relating documents to each other. These consist of the current time stamp, the semantic metadata which were input for the relation inference, and the identifier for the presented approach designating it as the source of the inferred relationship.

As the *ConSense* research project underlying the presented approach [6] is focused on multi user enterprise environments semantic document-relating metadata with relevance beyond that of the local user (that is higher-level metadata relating to domain-specific business entities) is then submitted to an enterprise-wide central semantic triple store. This central metadata repository, forming a virtualized view of assumptions on real-world relations among business entities and documents, can in turn be queried by a user or knowledge manager to visualize and cluster relationship types according to his task-specific discovery needs. So can for example an enterprise legal department query for documents

[8]Relations (edges) between window instances are treated as bidirectional for this purpose.

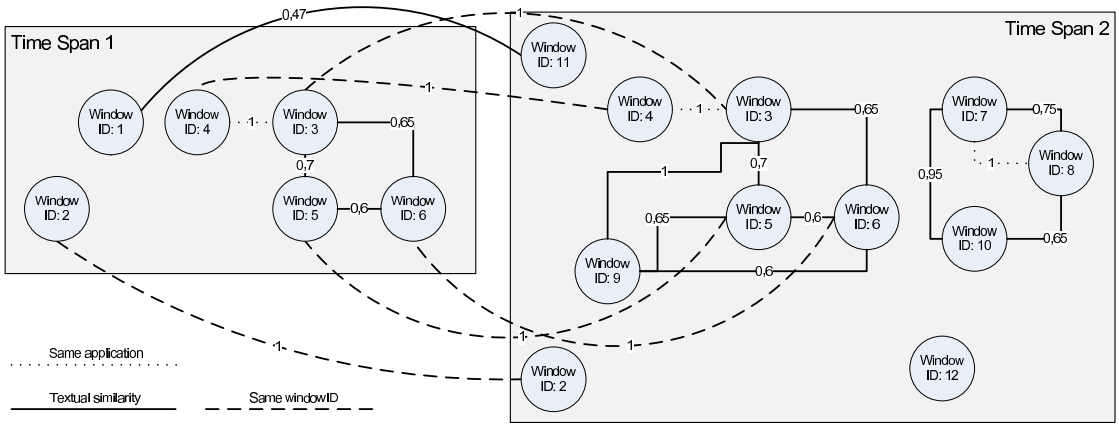

Figure 6: Example of window instances related across time spans

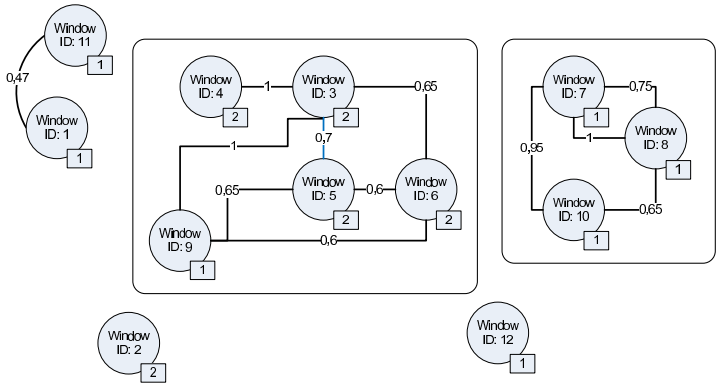

Figure 7: Example of a resulting activity cluster

having a relation to a product line being the subject in a legal low-product-quality complaint by a client. The query parameters could then be narrowed to documents having additional relations to the internal Quality Management process. Alternatively a rule or heuristic based electronic service can directly access the semantic aggregation layer and query or manipulate the semantic network.

4. EVALUATION

To test the viability of the presented approach the prototypically implementated sensor plugin was installed on the client desktops of four knowledge workers. During the evaluation 15 *Working Items* with durations ranging from 5 minutes to 5 hours were generated containing activity patterns of singular desktop activity, sequential desktop activity and multitasking activity as described in section 3.3. The goal of these experiments was mainly to to obtain information on the accuracy of the clustering algorithm both regarding false-positives (actions included in the clusters) as well as false-negatives (actual window activity relationships not detected by the sensor plugin). The test subjects were later asked to confirm or deny each of the individual *belongsToSameActivity* relationships concluded by the sensor plugin, denials resulting in a false-positive result for that match. In a second step the subjects were presented with the unconnected window instances and asked to cluster them according to their own remembrance of the performed activities resulting in false-negative errors where *belongsToSameActivity* relationships stated by the test subjects were missing in the sensor results.

In addition to the results shown in table 3 the document interrelations being identified as false-positives showed a lower average attached reliability (0.31 compared to 0.53) than the correctly identified relationships. Especially when considering that the reliabilities attached to document interrelations are not discarded but made available to further processing within the resulting semantic network a reconnection of the previously fragmented document based information can be achieved using the presented approach. So while the number of participants in this evaluation was limited the error rate lies in an acceptable range.

5. CONCLUSION AND OUTLOOK

It has been shown that the fragmentation of document based information in enterprise environments slows down workflow related information retrieval and poses a significant challenge to knowledge management initiatives. The paper demonstrated a new method to relate such unstructured documents containing similar information to each other without imposing additional tagging or training workload on the individual user. The method was prototypically implemented and evaluated in a small scale testing setup.

Table 3: Evaluation Results

Test	Subject	Window Instances	False Negatives	False Positives	Error Rate
1	A	62	1	6	11.29%
2	A	6	0	0	0.00%
3	A	12	0	1	8.33%
4	A	70	1	0	1.43%
5	B	12	0	1	8.33%
6	B	52	0	3	5.77%
7	B	28	0	0	0.00%
8	B	33	0	1	3.03%
9	B	21	0	2	9.52%
10	B	71	2	0	2.82%
11	C	51	0	0	0.00%
12	C	67	1	1	2.99%
13	D	16	0	0	0.00%
14	D	24	1	1	8.33%
15	D	60	0	1	1.67%
Total		585	6	17	3.93%

The amount of combined errors during the performed evaluation were in a noticeable yet – for the purpose of the presented approach – acceptable range. While the arguably more sophisticated existing task switching detection approaches as described in section 2 show a slightly higher rate of error when detecting task and activity switches this has to be explained by the steeper requirements applied by them. For the purpose of interrelating documents which does not require the detection of long running tasks contexts taking possibly multiple days to complete the presented approach is sufficient though. Furthermore the presented approach has the advantage of not causing user workload by imposing manual tagging or training requirements. Lastly the generated semantic relationships as well as the attached provenance and reliability statements can be further utilized by third party semantic Personal Information Management systems or Enterprise Content Management solutions.

In future research the mentioned heuristics to form activity clusters will have to be further tested and improved to reduce the number of false-positives. To this end domain specific heuristics might prove to be valuable. Also legislative aspects, especially privacy concerns of employees, have to be considered. Semantically white- and/or blacklisting named business entities as well as excluding specific window types or file system paths from the initial monitoring might be feasible to specifically include business process relevant documents only or to exclude documents and communication of sensible parties from the context readings.

6. REFERENCES

[1] S. Bailey. Has EDRMS been a success? The case for the prosecution. In *Opening Statement to the RMS Conference 2008, Edinburgh*, 2008.

[2] V. Bellotti, N. Ducheneaut, M. Howard, and I. Smith. Taking email to task: the design and evaluation of a task management centered email tool. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, page 352. ACM, 2003.

[3] R. Benbunan-Fich, R. Adler, and T. Mavlanova. Towards new metrics for multitasking behavior. In *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems*, pages 4039–4044. ACM, 2009.

[4] O. Bergman, R. Beyth-Marom, and R. Nachmias. The project fragmentation problem in personal information management. In *Proceedings of the SIGCHI conference on Human Factors in computing systems*, page 274. ACM, 2006.

[5] R. Blumberg and S. Atre. The problem with unstructured data. *DM REVIEW*, 13:42–49, 2003.

[6] H. Brügmann. Assisting the Discovery and Reuse of Document- based Knowledge using Semantic Metadata. In M. Schumann, L. M. Kolbe, M. H. Breitner, and A. Frerichs, editors, *Multikonferenz Wirtschaftsinformatik 2010*, pages 27–38, Göttingen, 2010. Universitätsverlag Göttingen.

[7] D. Buttler. A short survey of document structure similarity algorithms. *Proceedings of the 5th international conference on internet computing (IC2004)*, pages 191–208, 2004.

[8] P. Castells, M. Fernandez, and D. Vallet. An Adaptation of the Vector-Space Model for Ontology-Based Information Retrieval. *IEEE Transactions on Knowledge and Data Engineering*, 19(2), 2007.

[9] E. Cutrell, D. C. Robbins, S. T. Dumais, and R. Sarin. Fast, Flexible Filtering with Phlat? Personal Search and Organization Made Easy. *Microsoft Research*, 2006.

[10] M. Czerwinski, G. Smith, T. Regan, B. Meyers, G. Robertson, and G. Starkweather. Toward characterizing the productivity benefits of very large displays. In *Human-computer interaction: INTERACT'03; IFIP TC13 International Conference on Human-Computer Interaction, 1st-5th September 2003, Zurich, Switzerland*, page 9. Ios Pr Inc, 2003.

[11] A. Dragunov, T. Dietterich, K. Johnsrude, M. McLaughlin, L. Li, and J. Herlocker. TaskTracer: a desktop environment to support multi-tasking knowledge workers. In *Proceedings of the 10th international conference on Intelligent user interfaces*, pages 75–82. ACM, 2005.

[12] S. Dumais, E. Cutrell, J. Cadiz, G. Jancke, R. Sarin, and D. C. Robbins. Stuff I've seen: A system for personal information retrieval and re-use. *Annual ACM Conference on Research and Development in Information Retrieval*, 2003.

[13] Ferris Research. Industry Statistics 2009, http://www.ferris.com/research-library/industry-statistics/, 04/07/2010.

[14] W. Geyer, M. J. Muller, M. T. Moore, E. Wilcox, L.-T. Cheng, B. Brownholtz, C. Hill, and D. R. Millen. Activity explorer: activity-centric collaboration from research to product. *IBM Systems Journal*, 45(4):713, 2006.

[15] M. R. Gilbert, K. M. Shegda, D. Logan, K. Chin, T. Bell, L. Latham, R. E. Knox, and J. Lundy. Key Issues for Enterprise Content Management, 2006. Technical Report G00139515, Gartner Inc., 2006.

[16] V. M. Gonzalez and G. Mark. Constant, constant, multi-tasking craziness: managing multiple working spheres. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*, pages 113–120. ACM Press, 2004.

[17] M. Granitzer, G. Granitzer, K. Tochtermann, S. Lindstaedt, A. Rath, and W. Groiß. Automating Knowledge Transfer and Creation in Knowledge Intensive Business Processes. In *Proceedings of the First Workshop on Business Process Management and Social Software BPMS208 in conjunction with 6th International Conference on Business Process Management*, pages 1–10. Springer Berlin / Heidelberg, 2008.

[18] D. R. Hutchings, G. Smith, B. Meyers, M. Czerwinski, and G. Robertson. *Display space usage and window management operation comparisons between single monitor and multiple monitor users*. ACM Press, New York, New York, USA, 2004.

[19] W. Jones and H. Bruce. A Report on the NSF-Sponsored Workshop on Personal Information Management, 2005.

[20] W. Jones, S. Dumais, and H. Bruce. Once found, what then? A study of keeping behaviors in the personal use of Web information. *Society for Information*, 2002.

[21] V. Kaptelinin. UMEA: User-monitoring environment for activities. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, volume 4, page 1, Ft. Lauderdale, 2002. ACM Press.

[22] K. Kuutti. *Activity theory as a potential framework for human-computer interaction research*, chapter 2, pages 17–44. Context and consciousness: Activity theory and human-computer interaction. MIT Press, 1996.

[23] R. Lokaiczyk and M. Goertz. Extending Low Level Context Events by Data Aggregation. *Proceedings of IKNOW Š08 and IMEDIA 08*, pages 118–125, 2008.

[24] D. Lowd and N. Kushmerick. Using salience to segment desktop activity into projects. In *IUI '09: Proceedings of the 14th International Conference on Intelligent User Interfaces*, page 463, New York, New York, USA, 2009. ACM Press.

[25] P. Lyman and H. R. Varian. How Much Information 2003. Technical report, University of California - Berkeley's School of Information Management and Systems, 2003.

[26] W. Money and A. Turner. Assessing knowledge management system user acceptance with the Technology Acceptance Model. *International Journal of Knowledge Management*, 1(1):8–26, 2005.

[27] D. Oard and J. Kim. Modeling information content using observable behavior. In *Proceedings of the 64th Annual Conference of the American Society for Information Science and Technology*, pages 481–488, 2001.

[28] N. Oliver, G. Smith, C. Thakkar, and A. C. Surendran. SWISH: semantic analysis of window titles and switching history. In *Proceedings of the 11th international conference on Intelligent user interfaces*, volume 06, page 201. ACM, 2006.

[29] M. Perez-Quinones, M. Tungare, P. Pyla, and M. Kurdziolek. A Special Topics Course on Personal Information Management, 2006.

[30] A. S. Rath. *User Interaction Context - Studying and Enhancing Automatic User Task Detection on the ComputerDesktop via an Ontology-based User Interaction Context Model*. Dissertation, Graz University of Technology, 2009.

[31] A. S. Rath, D. Devaurs, and S. N. Lindstaedt. UICO: An Ontology-Based User Interaction Context Model for Automatic Task Detection on the Computer Desktop. In *Proceedings of the 1st Workshop on Context, Information and Ontologies - CIAO '09*, pages 1–10, New York, New York, USA, 2009. ACM Press.

[32] D. D. Salvucci, N. A. Taatgen, and J. P. Borst. *Toward a unified theory of the multitasking continuum*. ACM Press, New York, New York, USA, 2009.

[33] J. Shen, J. Irvine, X. Bao, M. Goodman, S. Kolibaba, A. Tran, F. Carl, B. Kirschner, S. Stumpf, and T. Dietterich. Detecting and correcting user activity switches: Algorithms and interfaces. In *Proceedings of the 13th international conference on Intelligent user interfaces*, pages 117–126. ACM, 2009.

[34] M. Tungare, P. S. Pyla, M. Sampat, and P.-Q. nones. M.: Syncables: A framework to support seamless data migration across multiple platforms. In *IEEE International Conference on Portable Information Devices*, 2007.

[35] L. Uden, P. Valderas, and O. Pastor. An activity-theory-based model to analyse Web application requirements. *Information Research*, 13(2):1, 2008.

[36] C. J. Van Rijsbergen, S. E. Robertson, and M. F. Porter. New models in probabilistic information retrieval. In *British Library Research and Development Report no 5587*, 1980.

[37] W. E. Winkler. The state of record linkage and current research problems. *Statistical Research Division US Bureau of the Census Wachington DC*, 1999.

[38] E. Zadok, J. Osborn, A. Shater, C. Wright, and K.-K. Muniswamy-Reddy. Reducing Storage Management Costs via Informed User-Based Policies, 2004.