Bridging the Gap Between Real and Electronic Documents

Renata Marin Tiago Guerreiro Daniel Gonçalves Joaquim A. Jorge
Dep. Engª. Informática, IST
Av. Rovisco Pais, 1000 Lisboa
renata.marin@gmail.com, {tjvg,daniel.goncalves,jaj}@inesc-id.pt

Abstract

In the area of Personal Information Management, researchers try to cope with problems arising from the large amount of personal electronic information users have to deal with nowadays. Part of that information are the users' documents. It is difficult to retrieve documents that have been written weeks or months ago, as existing systems provide little or no clues to their whereabouts. However, it is often the case when a printed version of a document is available, and the users want to find the corresponding electronic document. That printed version should be enough to make the retrieval possible.

We describe how RFID tags were used to solve that problem, by allowing an automatic association between electronic and paper versions of the same document. Our solution is part of the Quill system, a narrative-based document retrieval interface. Quill allows stories about documents to be told and used to find the documents they describe. With the use of RFID technology, it is now possible to mention information about real-world documents in those stories. Furthermore, Quill maintains a knowledge base in which information about the users and their activities is stored. The synergies between that information and the one collected with the help of RFID tags allow new use scenarios to be supported.

Keywords

RFID, document retrieval, narrative-based user interfaces, personal information management.

1. INTRODUCTION

Retrieving electronic documents is not an easy task. The hierarchic nature of common filesystems does not properly reflect how users remember their documents. To store a document in a filesystem it must first be classified into the hierarchy. This leads to cognitive problems, since the users are aware that from a good classification depends an easy retrieval at a later time.

Unfortunately, the classification task is fraught with problems. It might not be evident in which category to place a document. More than one might seem adequate (how to choose?) or none might. In the latter case, should a new category be created? If so, their number will grow, making the classification of future documents increasingly difficult. If not, the document will be placed into some kind of "miscellaneous" category, almost the same as not having been classified at all. Furthermore, even if users choose what seems to be the best classification at a given point in time, it is not certain that classification will be remembered later on, or even if it would still seem the correct choice, given that new documents might influence the classification criteria. These problems are so serious that several studies have shown that users will preferably keep their documents in unsorted "piles", rather than in organized "files", relying on clues such as special memory to find them [Malone83][Rodden99].

To make matters worse, computers, in their different forms, are pervasive in our society. Most users now have access to more than one machine. The computers at home and at the office are an example of this, but also are mobile phones or PDAs, that nowadays allow users to perform several tasks that before required a "traditional" machine. In particular, personal information and documents are now created, edited and stored in those devices. The problem of finding a document became more than just looking for it in a computer's filesystem. It now requires users to remember *in which computer* it might reside [Nielsen02].

The retrieval problems we just described arise, to a large extent, from the fact that while the computer forces users to think in terms of *files*, they are indeed dealing with *documents*. The filesystem and files are metaphors created to cope with the fairly basic storage capabilities of early computing systems. Users, on the other hand, remember their documents not only in relation to an *ad-hoc* category or a filename, but instead reporting to the wider context in which those documents were handled. As mentioned by Jeff Raskin, the content of a document is its best filename [Raskin00]. Users remember what is in their documents, when they were handled or why they were read or written. Using this autobiographic information to describe and retrieve documents would be helpful, as it mimics how users tend to remember them.

Whittaker's work confirms this, showing that many users resort to their email clients to store documents, sometimes inside "fake" email messages [Whittaker96]. Even if those tools don't directly support such tasks, all messages have associated to them information that will make the retrieval of documents easier: a subject, a date, etc.

While using autobiographic information to retrieve documents is a promising idea, just asking users for that information would yield poor results. It is necessary to somehow elicit it from them, and for the computer to understand it. To solve that problem we developed a narrative-based interface for document retrieval. Users will tell stories describing the documents they seek, and those stories will be used to identify them. All humans tell stories all their lives, from an early age. This is an innate human ability, as it was found that across cultures, even those of remote tribes, stories are told [Brown91]. It is, thus, a natural way for users to express themselves. Furthermore, the different story elements that compose a narrative appear not isolated from each other, but integrated into a coherent whole. Telling a story will enable users to more easily recall relevant information, instead of just trying to remember independent factoids about a document.

While developing Quill, our narrative-based document retrieval system, and choosing the shape its interface would take (described in Section 4), we identified the different elements that might appear in document-describing stories. Among them are references to related documents and to the document itself in printed form. It soon became evident that some way of taking printed documents into account when trying to retrieve a document was necessary. The work we present in this paper will complement Quill, establishing a relationship between virtual documents and their real world replicas.

We resorted to RFID technology to bridge the gap between electronic and paper documents. RFID tags are very cheap. RFID readers are also becoming cheaper and smaller. Thus, this technology can be used to identify the different paper documents handled by users with little cost. Our approach consists on the semi-automatic association of paper documents to their electronic counterparts whenever they are printed, with the help of an RFID tag. This allows users to employ one to find the other. Furthermore, by integrating this feature into Quill, data about real documents can be considered at the same time as other autobiographic information about the users, supporting different interaction scenarios rather than just allowing the blind retrieval of a document.

In the following section we will describe other approaches that try to use real documents for information management and retrieval. After identifying their shortcomings, we'll describe what a narrative-based interface is, and how such an interface can be used for document retrieval. This will lead us to a description of the Quill system, into which our solution was integrated. We'll then present our approach, detailing the different supported interaction scenarios, and mention some preliminary results. Finally, we'll conclude pointing to interesting possible future work.

2. RELATED WORK

As the sizes of our Personal Document Spaces increased, it became a priority to improve and ease the document-retrieval process. However, even with fast virtual document retrieval techniques, there is still a gap between the virtual space and the real printed documents. In our everyday life we can easily find situations like this:

> *"Isn't that the paper the teacher talked about yesterday? Can you send it to me, please?" asks Michael.*
>
> *"Sure", answers Charles.*
>
> *Half an hour later, Charles arrives at home and wants to send the pdf version of the paper to Michael. But … Where is the file? Charles has the printed version but doesn't find the pdf it comes from.*
>
> *"- Dammed!" says Charles, -"I've got here the printed version but I can't do anything with it…"*

The problem here is that while both documents, real and virtual, are the same, when the pdf file was printed the connection between them was lost. They still are just different facets of the same document, albeit in different media. As such, we should strive to maintain a relationship between them.

Some previous works try to accomplish this in different ways. *Video-Based Document Tracking [Kim04]* is an interesting work that tries to solve the physical/virtual document association problem. In the solution it proposes, an overhead camera is used to track physical documents on a desktop and link them to the corresponding virtual documents. The system detects changes in a document stack, and uses this information to establish the desk's contents. The movement of documents is tracked with a video-camera. The video is analyzed with computer vision techniques to connect the document with his virtual copy on the disk. Although this approach can relate some documents with their virtual replicas it isn't as very accurate, as should happen in a retrieval system. It relies on the visual pattern of the first page of a document to identify it, which is clearly not enough when several documents with similar first pages are present at the same time. Also this technique can only recognize documents if their first page is facing upwards and not occluded by some other document or object..

Other solutions, aiming at a more accurate linking between virtual and real documents are based on RFID (Radio Frequency Identification) technology.

Historically, the roots of the RFID technology can be traced back to the World War II. It was used by the British to distinguish their aircrafts from the enemy's. RFID is a generic term for technologies that use waves to automatically identify objects. Each tag is identified with a unique serial number. A microchip is attached to an antenna that enables the RFID to transmit the identification

to a reader. This reader converts the radio waves reflected back from the RFID tag into digital information that can be passed on to computers [RFID].

One of the earliest works that tried to bridge physical and virtual worlds through the use of RFID technology was *Want et al.* [Want99]. It tries to connect the physical objects with its virtual representation using various types of tags.

More recently, the bridge between virtual representations and real objects has been extended to personal items that accompany the users on their everyday tasks [Borriello04]. This work tries to help people not to forget important objects. The different relevant objects are tagged. RFID readers are present at each of the locations where the user usually spends some time at. Users are given a mini personal server that communicates with their watches. Whenever a user passes close to a reader without an important object (that he could have forgotten) the watch reminds the user of if. This work shows how RFID technology can be employed not only to help users to initiate the retrieval of objects they remember, but also how the computer itself can identify a user's needs and act proactively to provide the required objects.

Abu Safiya et al. [AbuSafiya04] developed a project that more directly addresses the information retrieval problem. The concept of Document Database (where all electronic documents are indexed) is extended to allow the representation of printed documents and their physical location. With this, a company can manage all its documents, becoming possible to know at all times where a specific document is or which documents are in a specific room. The RFID readers must be located in strategic positions, places where documents usually accumulate. While interesting, this work is a large scale project, trying to deal with an organization's entire document collection. Thus, only important documents are tagged identified. It is mostly a localization-based project and the retrieved information is quite poor. Also, the retrieval process is unidirectional: a user can't retrieve a virtual document from its real replica.

Our goal is similar to that of the systems we just described: to use RFID technology to manage and relate virtual and real documents uniformly and efficiently. However, our approach strictly focuses on personal document spaces having in mind the retrieval of virtual documents from their real-world counterparts and vice-versa. Since we're trying to help users manage their personal documents, our solution, unlike those above, instead of simply maintaining the link between real and virtual documents, uses a wealth of autobiographic information to enrich the documents' descriptions and further enhance the users' ability to find them.

3. NARRATIVE-BASED INTERFACES

Previous studies had shown that autobiographic information might be useful to organize and retrieve documents. The use of narratives as a way for users to convey that information to the computer seemed promising. Thus, we began to design a narrative-based interface for document retrieval. Rather than using stories to annotate documents, something few users would even consider doing in a consistent way for all their documents, we focused on using narratives as a way to describe documents at retrieval time.

To validate the approach, we conducted a set of interviews in which users were asked to tell stories about their documents [Gonçalves03] [Gonçalves04]. We collected 60 such stories, told by 20 different users. Each user told a story about a Recent document (written by the user less that two weeks earlier), an Old document (written by the user more than 6 months ago), and a document with Other authors. The interviews were recorded, transcribed, and contents and relational analysis were performed on the transcripts [Huberman91].

Overall, we found stories to be composed of 17 different elements: Time, Place, Co-Authors, Purpose, Author, Subject, Other Documents, Personal Life Events, World in General, Exchanges, Type, Tasks, Storage Location, Versions, Contents, Events occurring when handling the document, and the document's Name. Each element occurs as a semantically significant sentence or phrase in the stories.

We tried to find differences in stories arising from (among other factors) the different document types, the elapsed time since the document had been handled and user gender and age. No noteworthy differences were found, with one exception: the length of stories about documents written by the users is slightly greater than that of those they didn't write (17 vs. 11 elements, on average). The different story elements also appeared with different frequencies. Most notably, Place, Co-Authors, Purpose, Author, and Versions appear less frequently. Statistical tests (with 95% confidence) confirmed these results.

In short, it was confirmed that narratives about documents convey lots of information about those documents, and can be easily told by all, regardless of gender and age. We also got an extensive description of what information to expect in stories, and in what order.

Based on those results, we were able to create some guidelines for the design of narrative-based document retrieval interfaces. The most important are:

- In general, no user customization will be necessary in relation to what to expect from a story.
- It is important to determine the kind of document being described early in the narrative, to correctly form expectations about what can be found ahead in the story.
- The typical story structures we found should be used to help understand what element is being described by the users at any given time, and to help understand the information therein.
- Users tend to digress. As such, it is important to establish dialogues with them in order to keep them on

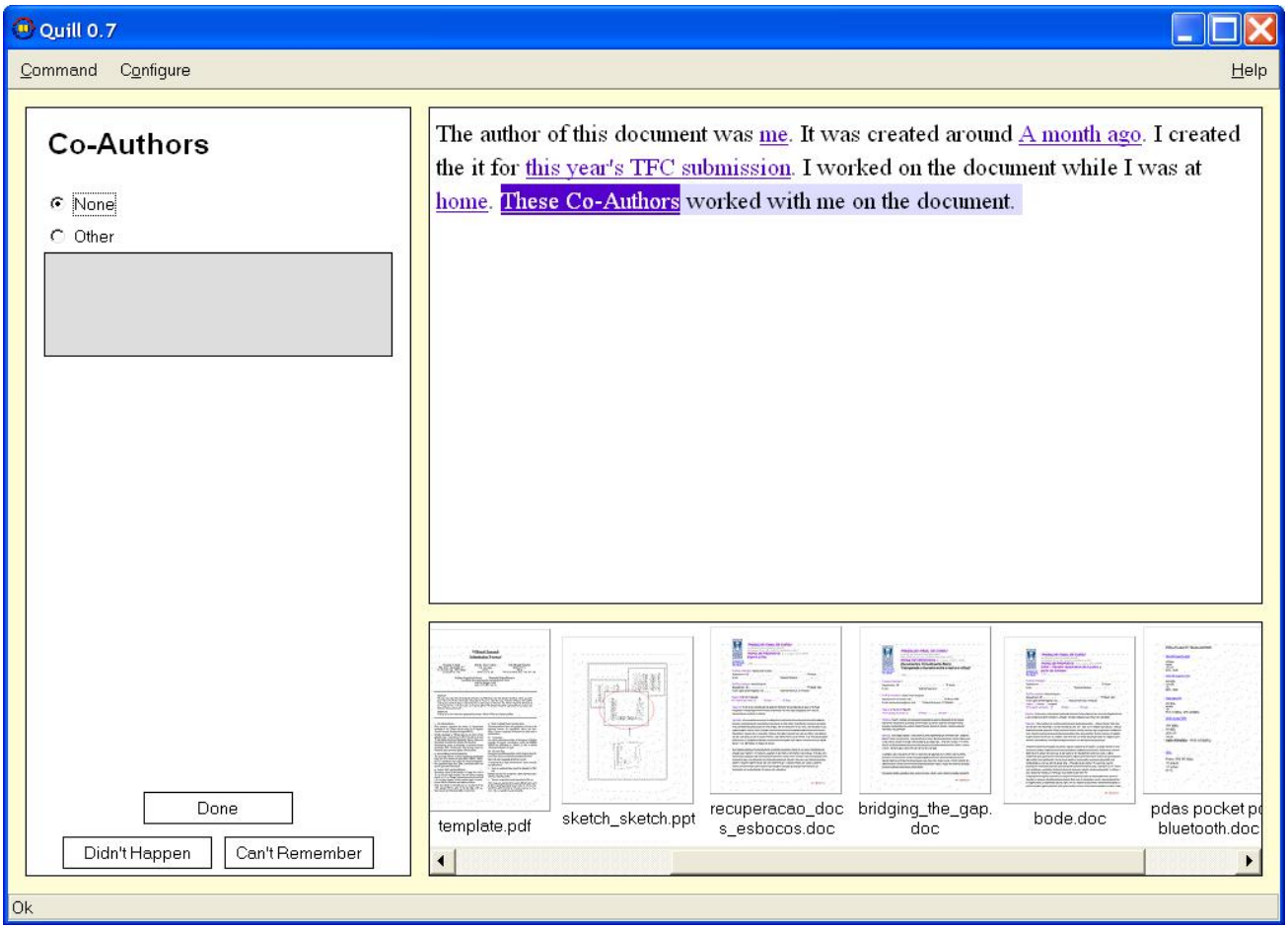

Figure 1 – The Quill Interface

track and to obtain all information they can actually remember.

- We found that the users' visual and special memory play important roles in stories. Some technique that identifies the overall structure or visual appearance of a document and can use that information to differentiate among several documents would be useful.
- Related documents are often mentioned in "stories within the story". The interface should be able to capture those stories and, at the same time, lead the users back to the description of the target document.
- Actions such as printing a document are mentioned fairly often. We should take those actions into account. Printing and printed documents, in particular, should be taken into consideration.

In possession of these guidelines, we knew what to expect form stories, and had an idea of the features the interface to which those stories are told should take. The actual shape of that interface remained to be established.

To do so, we created and evaluated two low-fidelity prototypes of narrative-based interfaces for document retrieval [Gonçalves04a]. Both embodied the design guidelines. The first was based on the direct manipulation of the different story elements, depicted as boxes that could be arranged on screen to create the stories. The second represented the story as text. Each element corresponded to a sentence. Incomplete sentences were presented to the user that only had to fill in the missing information. In both prototypes the story elements were entered with the help of specialized dialogues, and suggested to the user in the order found to be the most likely in the interviews. In a actual system, those stories would then be used to identify the document being sought by the users.

Again, we collected 60 stories, from 20 users, using both prototypes. The second prototype was undoubtedly the better one. Stories told using it were similar to those told to humans, both in terms of content and structure. The users' subjective satisfaction was also clearly better for that prototype.

Based on these results, we implemented the Quill interface, described below. Quill allows the users to tell their stories and uses them to identify, from an index, the document that best matches it. With Quill, we performed a third and final study regarding the validity of stories as a way to convey information about documents. In that study, the information in stories was evaluated regarding its accuracy [Gonçalves05]. We found that on average, we can expect between 81% and 91% of the information in each story to be correct. This means that only between 1 and 3 story elements might be wrong (out of 17). Furthermore, we found that, for the most part, the informa-

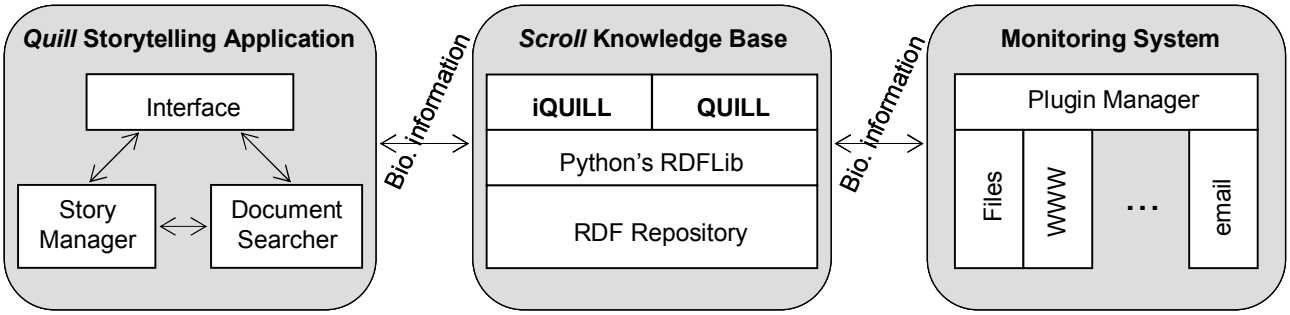

Figure 2 – The Quill System Architecture

tion isn't completely incorrect. For instance, when mentioning a document's name, only part of it is usually wrong, or the words in the name might be mentioned in an erroneous order ("report200" instead of "2000report", for instance). Thus, if the information is not taken as an absolute truth, it might still help identify the desired document. Surprisingly, stories about Recent and Old documents are equally accurate. The only thing that varies is the precision with which the different elements are mentioned. For instance, we find time ranges of a month instead of a few days, but correct nevertheless.

Narrative-based interfaces were, thus, confirmed as being a valid and sound approach for users to convey information for document retrieval.

4. THE QUILL SYSTEM

The Quill interface is depicted in Figure 1. The story is created in the top-right area. A sentence representing each possible story element is suggested, in turn, to the user. In that sentence, the information to be completed is highlighted. The user must then enter it using the dialogue box in the right. There is a different dialogue for each story element, and we ensure the appropriate dialogue is visible. Additionally, the user can state that something didn't happen, or that he can't remember a particular story element. This is done with the help of two buttons at the bottom of the dialogue boxes. It allows the interface to tell the difference between something the users don't know, and something they know not to have happened. Also, while Quill suggests each story element to the user in the order found to be the most likely in previous studies, the user remains in control, being able to choose any element to mention at any point in time. Quill adapts to those changes over time, fine-tuning its suggestion order with the help of hidden-markov models.

The sentences are adapted whenever new information is entered by the user, to make them coherent with that information. We took care not to change them too much, to prevent confusing the users.

As the story grows, promising documents are found. A thumbnail of those documents is displayed in the bottom of the window. This takes advantage of the fact that users often remember the overall aspect of a document (visual memory plays an important role). This way of presenting the probable matches is non intrusive and does not unduly distract the users from the storytelling process. Often, just by quickly scanning the list, it is possible to tell if the target document has been found or not.

Limited natural-language understanding is provided. A full parser tries to understand the phrases entered by the users and automatically generate their semantics. If this fails, a tag parser tries to extract as much meaning as possible from the phrases.

A sample story, collected by Quill, is presented below:

```
The author of this document was me.
It was created around 10 of May of
2004. I created it for PCM Report.
I worked on the document while I
was at home and the workplace and
At my colleague's home, in college.
André Martins worked with me on the
document. The document is about
CGEMS Advanced Search Engine. This
document reminds me of no other. I
sent it to André Martins using
email and LAN (shared folders,
etc.). It's a PDF document. The
document contains the words or ex-
pressions "Search Engines, CGEMS,
Java, SIGGRAPH" and looks like a
two-column with lots of images and
a little text. The document is
stored in Laptop and Other com-
puter. To write it, I had to devel-
oped a prototype for PCM, Search
the Web, Read many related papers.
It had different versions. Its
filename was something like "pcm
final".
```

As can be seen, while the syntax isn't perfect, it is sufficiently good to allow it to be read and understood by the users.

4.1 Behind the Scenes

Underlying Quill is the Scroll Knowledge Base (KB). This knowledge base uses RDF as its knowledge repre-

sentation formalism, and stores knowledge about the users, their documents, and their actions (see Figure 2). A special-purpose library called *Scroll* provides high level methods to deal with the RDF store, and allows path- and node-based inference to be performed. Also, several case-frames were defined. The Quill Schema allows us to represent all knowledge mentioned above, and the iQuill Schema makes it possible to represent first-order logic-like formulae and use them to perform inferences.

Whenever new information is entered into the story, inference rules are generated and evaluated in the KB. From that evaluation, relevant documents are identified. To each is given a score. The scores from resulting from all inference rules are added and the resulting value is used to establish a document ranking. The best placed documents are suggested to the user as possible matches.

4.2 Monitoring the User

To feed knowledge into the KB we created a monitoring system. This system continuously examines what is going on in the user's computer and selects relevant knowledge to be stored in the KB. It is a plugin-based system. Currently, we are able to monitor the following:

- All documents present in the users' computer, keeping track of their whereabouts, and of when they are created, modified, copied or deleted.
- All web pages visited by the user.
- All email messages sent and received by the user, as well as the attachments therein.
- All programs the user runs.
- The user's agenda (appointments, contacts, etc.)
- Real documents (with the help of RFID tags, as described in the following section).

While independent, the different plugins generate a coherent body of knowledge. For instance, if the email plugin finds an attachment that has been saved somewhere in the hard drive (as identified by the document plugin), no new document will be recorded. Rather, it will store in the KB that the existing document was sent by email to someone, enriching the information gathered on that document.

Finally, the plugins try to extract as much information as possible from the documents. This includes meta-information stored in them (ID3 tags for music files, for instance, or META tags in HTML documents). For textual documents, relevant keywords are also extracted. First, the text is tokenized and stemmed. Then the TFIDF algorithm is used to select the most relevant keywords for each document.

5. BRIDGING THE GAP

From the user's perspective, a document is often connected with a real physical copy of itself. It became evident in our user studies that the relationship between virtual and real documents is often important in the document retrieval process. They are seen as different facets of the same document, rather than separate entities. Hence, as the relation is bi-directional we can augment our retrieval process to find both virtual and real copies of a certain document.

To bridge this gap and relate both worlds, we built our new plugin based on an RFID infrastructure. It is composed by a fixed RFID reader (with a range of 30-40cm) that is capable of reading multiple tags simultaneously, connected to a Personal Computer. It is important that it has a relatively short range, since we want to know precisely where our documents are (a desk or a shelf, for instance). The short range prevents overlapping readings if multiple readers are to be used. Our RFID tags are passive (with no power source), as we only need to receive the tag ID. Each printed document will have its own tag, which identifies it. The plugin associates the tag's ID with the corresponding virtual document (Figure 3).

We used RFID tags rather than other ways to identify documents, most notably barcodes, because they require little attention from the user. A barcode could be automatically generated and printed with the document. With RFID tags, on the other hand, the users must attach a tag to each document they print. Also, since the computer doesn't know beforehand what tag will identify a document being printed, a semi-automatic way to perform the association between an electronic document and the tag had to be devised (described in section 5.2). Nevertheless, this occurs only once, when the document is printed. Afterwards, the users have only to place the documents wherever they choose (a desk, a cabinet, etc.), and if an RFID reader is nearby, the document's location will be known to the system. Using barcodes, each time a document is moved it must be swept by a barcode reader. This might distract the user from the current task, and is liable to be forgotten.

The trade-of between having to remember to record the document position every time its moved and having to explicitly identify it once at print time led us to opt for RFID technology.

To accomplish our goals, we dealt with two different but related problems while developing the plugin: Abstracting from the low-level communication with the RFID reader operation into meaningful, document-related tasks, and discovering which documents were printed.

5.1 Reader Data Acquisition.

The plugin periodically receives data from the reader. Whenever a tag is detected, the plugin processes it, classifying it as a new or already known tag. This process can be executed over a set of tags making it possible to detect changes in a document pile or another location where several documents are stored.

If the tag is new, we face a new paper document for the first time and must determine to which virtual document it corresponds. If not, it means that a previously known document has changed place. In both cases, the information in the KB will be updated accordingly.

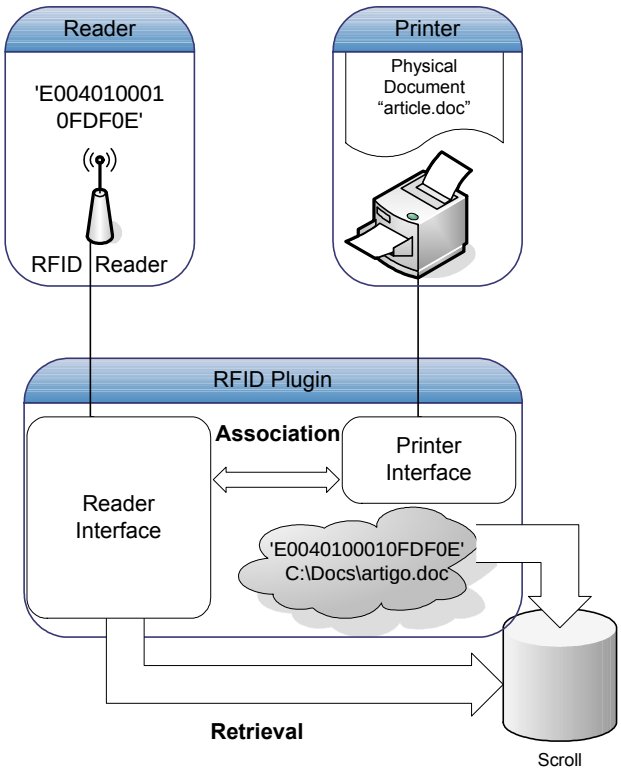

Figure 3 – Real and Virtual Document Association

5.2 Automatic documents printing detection

To associate a printed document to its virtual counterpart, we need to know when a document passes from the virtual world to the real one. By other words, when a document is printed. We assume that when a document is printed an RFID tag will be attached to it by the user, as described before. As a result, the document will be univocally identified by the hexadecimal key of the RFID. Establishing the association with a low effort from the part of the user must then be accomplished.

To accomplish this in the most automatic and less cumbersome possible manner, we follow five steps:

- Intercept the printing event;
- get the name of the file that was printed;
- read the printed document's ID key;
- establish a relationship between the file and the key;
- store the relevant knowledge in our KB.

To detect when a document is printed, we have to capture the operating system's printing events. This allows us to detect when a new document is being printed and needs to be associated to the respective file. We can get the file name from the print job event, but not the entire path. To determine exactly which file is being printed, we use our Scroll knowledge base, where all the users' documents are indexed, to identify all files with that name. If there is only one such file, the association is automatic. On the other hand, if the query returns more than one result, several different files with the same filename, the users are asked to choose between them. All different possibilities will be displayed in a list, sorted by modification date, since it is likely that a file being printed has been recently modified.

The different candidate files are shown to the user in a dialogue box that shows up when a document finishes printing (Figure 4). There, the users have the option not to associate an RFID tag to their document. If they accept, the information about the document in the knowledge base is updated with the respective RFID key. The bridge between both representations is thus created and both virtual and real copies are enriched with extra-information.

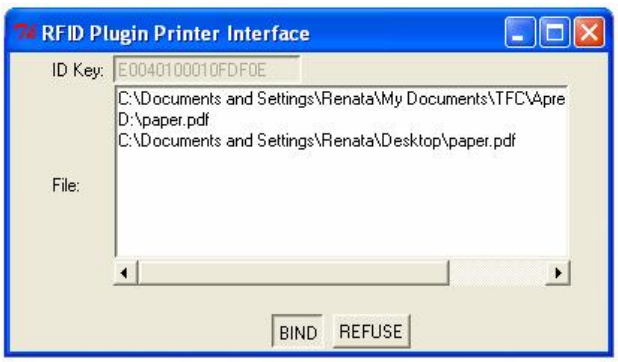

Figure 4 – Prototype of the RFID Printer Plugin

This relationship between the knowledge base representation of our virtual documents and their physical replicas enables us to retrieve documents in different ways. Considering the narrative document retrieval system, Quill, it is now possible to expand the narratives to the real documents context, information that is frequently mentioned by users. Thus, any user can mention in his story the printing occurrence, the real placement of a replica ("*and there is a real copy of the document in that shelve*") or, with the real document in his hands, retrieve the virtual copy ("it's *this* document"). Besides, when a document is retrieved, and we don't know the physical copy's whereabouts, we can ask Quill.

Figure 5 – Multiple Tagged Documents

Also, we developed another application that uses the knowledge base but is independent from our retrieval GUI. It is focused on the history of a real document. For example, if we have a printed document we may want to know where it came from, who did we send it to, and obviously, where is the document in our virtual space. In short, as the bidirectional bridge between both worlds (real and virtual) is established we are now able to retrieve documents, either physical or virtual, and information about them, in a direct, seamless way.

5.3 Evaluating the Association Accuracy

Our approach makes the establishing of a relationship between paper and electronic documents as seamless and effortless as possible. However, it is based on the assumption that most printed documents have a unique filename, allowing the RFID plugin to correctly guess the file being printed. To verify if this is, indeed, the case, we performed a user study in which the documents several users were considered.

A special-purpose program (for Windows machines) that traverses the users' hard drives and analyses the files therein was created. For each user, the program recorded the number of times each filename was found. The program was made available online and a request for participation was sent to our research group's mailing list. Participants needed only to download the program and run it on their machines. It would automatically look at the contents of the "My Documents" and "Desktop" folders and index them. A blacklist / whitelist mechanism (to allow other folders to be analyzed) was available and used by some participants that had most of their documents outside the default folders.

The program considered only files that were established to be documents, based on their extension. For instance, ".exe" files were not considered, but ".doc", ".pdf" or ".jpg" files were.

It was important to make the program as easy to use and understand as possible, to alleviate possible privacy concerns. Thus, its results would be written to a file in text format, so that the users could examine it before submitting the results to us by email.

Overall, 23 participants sent in their results. All participants were engineering graduate or undergraduate students, or faculty. Some ran the program on more than one machine, resulting in 27 datasets.

As can be seen in Figure 6, most (54.4%) of the 192019 files analyzed by the program have unique filenames. These will be automatically identified by our RFID plugin. For the remaining files, several candidates must be suggested to the user. However, for the most part, the number of candidates won't be superior to seven, within the number of elements that can be held and processed by the users' short-term memories. Furthermore, since those possibilities will be presented sorted by modification date, the document they seek will most likely be the first in the list. Only for 12.67% of files will there be more than 7 candidate files.

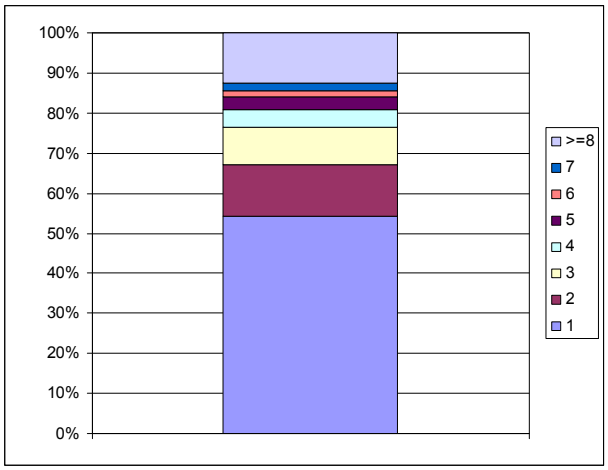

Figure 6 – % of files by filename repetition

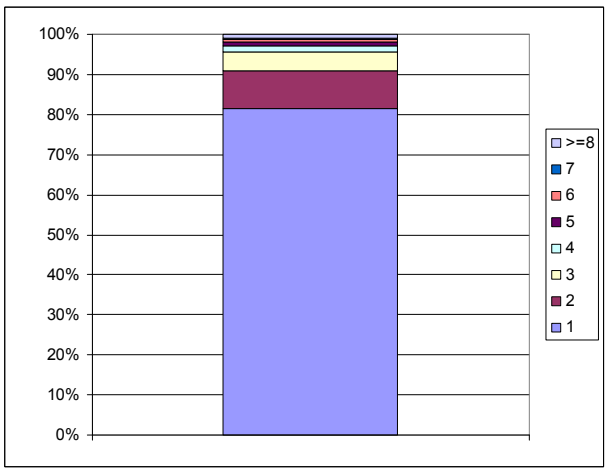

Figure 7 – % of names by filename repetition

Figure 7 shows us the percentages of repeated names, instead of files. There, we can see that 95.74% of names are used for three files at the most. Indeed, 81.42% unequivocally identify a document. This shows that the results where only the numbers of files are taken into account can be misleading. A single name occurring 465 times on a user's computer (and other similar cases) can somewhat distort the results. What is more, a closer look at those files whose names occur more than seven times shows that, for the most part, they are not documents that the users will normally print (no user would normally give the same name to 465 files…). For instance, it is often the case where they are text files accompanying software packages (the classic "readme.txt" files). For that reason, the files that users might want to print will likely share their name with up to three or four other files, well within the limit of what the users can manage.

6. CONCLUSIONS

Retrieving personal electronic documents is not an easy task. One way to make it easier is to resort not only to the hints provided by the filesystem, but also to a wealth of autobiographic information that users often remember

about their documents. References to printed documents are common. However, once a document is printed, its relationship with the electronic version is lost.

We developed a way in which it is possible to associate printed documents to their electronic counterparts with low effort from the part of the user. Our solution makes it possible to automatically detect when and what documents are printed. The identification of the printed file is not perfect, but we've shown it will automatically succeed for nearly 55% of all files, requiring little effort from the user if that is not the case. Simply by sticking an RFID tag to the printed document, the relationship is established. This supports a variety of interaction scenarios, in which not only electronic documents can be retrieved from their physical counterpart, but also where a printed copy of an electronic document can be located. Furthermore, since our solution was integrated into Quill, a narrative-based document retrieval system, the synergies between the printed-electronic document association and knowledge about the users and their activities could be explored.

In the near future, we'd like to test our approach with more than one reader, to extend the usage scenarios by allowing users to find printed documents that might be located in different places. The placement of a reader near the office door and the use of RFID tags by those who enter and leave it would extend those scenarios further, by allowing the user to know who took a document and when.

As equipment prices continue to drop, the user interaction could be made more seamless, by using printers that are able to print not only the document but also the tag, making it unnecessary for the user to stick the tags themselves.

7. ACKNOWLEDGEMENTS

The authors would like to thank all users that participated in the studies described in this paper. This work was funded by Project BIRD, FCT POSI/EIA/59022/2004.

8. REFERENCES

[AbuSafiya04] AbuSafiya, M. and Mazumdar, S.: *Accommodating Paper in Document Databases*, DocEng'04.

[Borriello04] Borriello, G., Brunette, B., Hall, M., Hartung, C., Tangney, C.: *Reminding about Tagged Objects using Passive RFIDs,* Sixth International Conference on Ubiquitous Computing, UbiComp 2004.

[Brown91] Brown, D. E.. *Human Universals*. New York: McGraw-Hill 1991. ISBN: 0-07-008209-X.

[Gonçalves03] Gonçalves, D. *Telling Stories About Documents*, Technical Report, Instituto Superior Técnico, 2003 (http://narrative.shorturl.com/files/telling_stories.zip)

[Gonçalves04] Gonçalves, D. and Jorge, J. *Telling Stories to Computers*. In Proceedings CHI2004, ACM Press, 27-29 April 2004, Vienna, Austria.

[Gonçalves04a] Gonçalves, D. and Jorge, J., *"Tell Me a Story": Issues on the Design of Document Retrieval Systems*. In Proceedings DSV-IS'04, Lecture Notes on Computer Science, Springer-Verlag, July 2004, Hamburg, Germany.

[Gonçalves05] Gonçalves, D. *Real Stories about Real Documents: Evaluating the Trustworthiness of Document-Describing Stories*. Technical Report, IST/UTL. March 2005, (at http://immi.inesc.pt/~djvg/phd/files/real_stories.zip)

[Huberman91] Huberman, M. and Miles, M. *Analyse des données qualitatives. Recueil de nouvelles méthodes*. Bruxelles, De Boeck, 1991.

[Kim04] Kim, J., M.Seitz, S., Agrawala, M.: *Video-Based Document Tracking: Unifying Your Physical and Electronic Desktops,* UIST '04.

[Malone83] Malone, T. *How do People Organize their Desks? Implications for the Design of Office Information Systems*, ACM Transactions on Office Information Systems, 1(1), pp 99-112, ACM Press 1983.

[Nielsen02] Nielsen, J. *Supporting multiple-location users*. Jakob Nielsen's Alerbox, May 26, 2002. http://www.useit.com/alertbox/20020526.html

[Raskin00] Jef Raskin. *The Humane Interface*. Addison-Wesley, 2000. ISBN: 0201379376.

[RFID] RFID Journal – The History of RFID Technology http://www.rfidjournal.com/article/articleview/1338/1/129.

[Rodden99] Rodden, K.. *How do people organise their Photographs*. In Proceedings of the BCS IRSG 21st Annual Colloquium on Information Retrieval Research. 1999.

[Want99] Want, R., Fishkin, K., Gujar, A., Harrison, B.: *Bridging Physical and Virtual Worlds with Electronic Tags*. ACM Conference on Human Factors in Computing Systems, Pittsburgh, PA, May 1999, 370-377.

[Whittaker96] Whittaker, S and Sidner, C. *Email overload exploring personal information management of email*. In Conference proceedings on Human factors in computing systems, pages 276-283. ACM Press, 1996. ISBN 0-89791-777-4.