An Experiment Using Document Annotations in Education

Dr. Francis Lapique

Institute for computer Communication and Applications, EPFL-DI-ICA, Swiss Federal Institute of Technology CH-1015 Lausanne Switzerland, Tel: +41-21-6936749, Fax: +41-21-6936610, E-mail: francis.lapique@epfl.ch

Gil Regev

Institute for computer Communication and Applications, EPFL-DI-ICA, Swiss Federal Institute of Technology CH-1015 Lausanne Switzerland, Tel: +41-21-6936790, Fax: +41-21-6936610, E-mail: gil.regev@epfl.ch

Abstract: This paper describes an on-going experiment at the Swiss Federal Institute of Technology in Lausanne regarding the use of Web based document annotations for educational purposes. An annotation tool called Medium was built, merging database and Web technologies that will be used as a companion for classical university courses and collaborative learning experiments.

1   Introduction

Document annotations are a very old concept. Much business and academia work is accomplished through annotation of documents. The usual scenario is that of a member of a group producing a document, which is then distributed to the group members for review. Each member reviews their own copy of the document and later returns the annotated document to the author. The author merges the comments into the document and produces a new document for review. After a number of iterations the document is declared final and is typically published outside of the group.
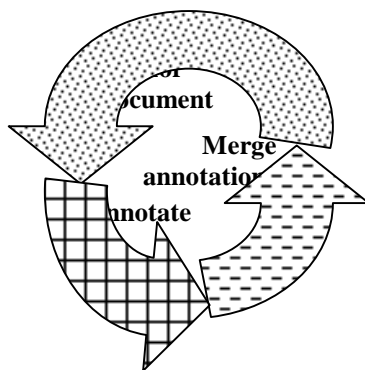

Figure 1: The document refinement process

As the document is refined during these iterations, the knowledge of the group and its understanding of the domain grow. So this is in fact a group learning process centered on the document and its annotations. This process is typically made more complicated when the document is authored by several members of the group simultaneously.
Annotations are mainly used in education for the purpose of producing teaching material which boils down to the same process as outlined above. Our intent is to use annotations within the teaching or learning process itself, e.g. as a companion for the classroom experience.

2   Annotations Versus News Groups

It is our belief that news groups lack the contextual binding that exist between a document and its annotations. News groups threads tend to diverge after a few iterations making them incomprehensible for the newcomer. Many times in order for the reader to understand a certain message of interest they

need to follow the thread the first message. Annotations, on the other hand, share a very strong context, which is the document itself. This is both a good thing and a bad thing. It is good because it helps users to understand the annotation and comment on it. It is bad because it may hinder people from annotating for fear of saying wrong things thereby showing their misunderstanding of the issue at hand.

3   Other Annotation Tools

Since annotations are not a new concept, one could expect to find many annotations tools available. The mainstream tools that support annotations are word processors such as Microsoft Word and groupware products such as Lotus Notes. In the past few years we've seen the emergence of Web based annotation tools. Some examples we found are the following:

- The ComMentor project at Stanford University is an early effort intended mainly as a proof of concept of the feasibility of Web annotations [Röscheisen et al. 1994].
- FEED Magazine (http://feedmag.com) operates a forum where readers can comment articles posted by authors. This is, however, not a fine-grained annotation tool [Drexler 1987] because annotations can only be placed within hooks (called loops) provided by the author.
- CritLink Mediator (http://crit.org) is a direct descendent of the ideas exposed by Drexler. It is aimed at enabling people to annotate any Web page using a proxy-based architecture.

4   Design Constraints

From the outset of the project, we formulated the following design constraints:

- Comments can be made on any portion of a text, fine grained annotation as defined by Drexler [Drexler 1987]
- Users will have to log into the server. This help us identify them and insures some filtering since non-authorized users can not contribute comments
- The user interface should be designed so that it operates well with small screen resolutions e.g. 640x480
- Medium should operate well when used with a standard 28K modem connection
- Medium should be usable with any Web browser supporting HTML frames and JavaScript. Users should not be required to install extra client software such as a plug-in
- The user interface should be as modeless as possible

5   Architecture

Medium's core technology is an Object-relational database engine (Informix Universal Server) and its extensions in the form of DataBlade modules which extend the capabilities of the server with new user-defined data types and related methods. Of the multitude of existing datablades we use the following 2:

1. The WebDataBlade, interfaces the database with Web the server and browsers
2. The TextDataBlade, implements basic text search

The front-end can be any browser that supports frames and JavaScript. The user interface is implemented through HTML pages with JavaScript code. The HTML pages themselves are generated dynamically from the database. JavaScript allows to automate the HTML forms so that a Submit button is not required in order to execute an action. For example, when the user selects a document from the document list Medium immediately loads the selected document.

Since we are using a central database, documents have to be imported before they can be annotated. We will discuss later what documents can be imported and from where.

6   Development Process

We chose to use an incremental development process. The intents was to have the basic system up and running in a matter of a few weeks and from than on refine by taking into consideration user input and extending the functionality by taking into consideration users and developers wish list. Thus, we chose to use low tech. technologies, which allowed us to have fast turnaround cycles.

7 Features and Restrictions

Medium offers the following features:
- Annotations of different types: Text, Java applets, VRML scenes, Audio clips
- Text Annotations can HTML tags. For less advanced users, an optional link field is provided allowing them to include a reference to a Web page that is attached to the annotation.
- Named annotations, an optional name can be specified when an annotation is added
- Annotations can be updated or deleted only by their author
- Annotations are public, anybody who connects to the server even as a guest can see them
- Annotations carry a timestamp and the name of the annotator
- Original documents are never altered. Annotations appear as a layer on top of the original
- Contributions can be viewed by document (called contribution) and by contributor

Medium supports importing of HTML and Tex documents. HTML documents can be imported from the local disk or directly from the Web by supplying their URL. The latter is the preferred method since documents can than keep their images, applets etc, which are loaded at runtime from the server from which the document was imported. This of course requires that these images and applets will not be deleted from the original server, which is not always in the hands of the contributor.

Medium has the following restrictions and limitations:
- URL's within a document can not be annotated
- Images can not be annotated
- HTML documents loaded from a local drive will lose the images and applets that they contain

8 User Interface

The user interface of Medium has been optimized from the beginning to encourage people to annotate. Thus, there are only 2 required fields for a text annotation:
1. The text to annotate
2. The text of annotation itself

All other fields are optional. They take default values if not used. For instance, the name field, which is used to give an explicit name to the annotation, defaults to a number if left empty. Medium will automatically assign a number, increasing from 1 for the first annotation, and apply it to a non-named annotation. This makes the process of adding an annotation very efficient, enabling users to concentrate on their annotations rather than on the mechanism of annotating.

9 Evolution

Up to this date, we can clearly identify 3 stages in the evolution of Medium issued from the incremental process described above. In the following figures, the left pane shows the original document whereas the right one represents the document with an annotation.

9.1 First Design

The first design involved four frames [Fig. 2], frame 4 for the document and three others to manage projects, to enter and view annotations and a frame that enabled users to select documents and annotations within a project. For each contributor we collected in the database the original document and annotations. Major disadvantages: The area allocated for viewing the document was too small, users were lost in all these frames, information was duplicated between frames. The annotation process required the user to select the text to annotate and copy it into the annotation form, then enter the annotation text itself and any optional fields and submit the form.
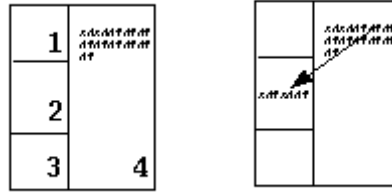
Figure 2: First design

9.2 Second Design

The second design involved only two frames [Fig. 3]: The largest frame was allocated for the document and the horizontal frame underneath was used for handling all the interactions with the user. Using the getSelection method of JavaScript we were able to capture the selected text in the document frame and insert it automatically into the text to be annotated when the user chose to annotate, thus reducing the process of annotating by one step. This however only works with Netscape browsers version 4.0 and higher. Curiously, it does not work with the latest Microsoft browsers. Annotations are displayed directly in the document when the user selects one.

Figure 3: Second design

From the database design point of view, the advantages are that we don't duplicate information, one table named "medium" and complex data types are introduced (doc_t and note_t). Following are SQL code snippets, which show the usage of these data types:

```
create table medium (
document              doc_t,
note                  note_t,
data_txt              html,
data_blob             blob) put data_blob in (sblobspce),data_txt in (sblobspce);
```

note_t is a named row type as follows:

```
create row type note_t (
owner                 user_t,
name                  varchar(10),
where                 varchar(50),
abstract              varchar(40),
count                 integer,
type                  integer,
update                datetime YEAR to second,
language              varchar(10));
```

We assemble the original document and annotations with JavaScript. Major disadvantages:
- For large sized documents (> 100K), annotation access time is too long
- The JavaScript program consumes too many resources on the client side
- Each time an annotation is opened or closed, the whole document needs to be reloaded which takes a long time and interrupts the thought process of the user.

9.3 Third Design

The third and current design keeps the two frames concept explored in the previous design but there is no more reloading of the document when the user requests an annotation. Instead, annotations of type text, are displayed in the bottom frame, other types (VRML scenes or Java Applets) are displayed in a new browser window. Audio annotations are simply sent to the audio player with no display involved. The search for the text to be annotated which was based on simple substring matching is now performed through the text search DataBlade module which enables Medium to search information in ways that are more sophisticated such as exact, fuzzy searches and synonym matching.
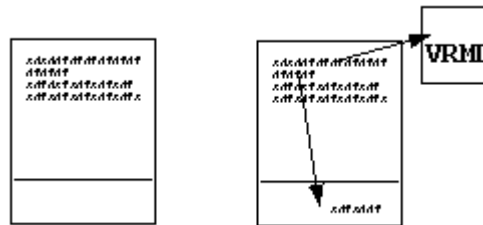

Figure 4: Third design

10   Extending Medium

We have realized that annotations lead to discussions so that people want to use Medium as a groupware tool. In order to support discussions we are considering adding the following functionality:

- Contextual chat, users will be able to start a conversation with other users connected to Medium at the same time. Our aim is to enable users to not only see that other people are present in the system but that they may be in the same context. For now, we define the context as being the document. This will enable the system to alert user A to the fact that user B may be reading or annotating the same document as the one user A is reading.

- Concept map description of document, concept maps are very useful to graphically describe a document. We want to explore the feasibility and impact of enabling people to annotate a document with a concept map which is linked with the document so that clicking on symbols in the concept map trigger the display of the corresponding area or keyword within the document.

- Annotations within annotations, users should be able to annotate an annotation thus sparking of a discussion. This, in essence, will make the document be the basis of a groupware process.

- Annotation scope, since people may feel uncomfortable with the public nature of annotations, we are going to introduce a scoping property which will enable users to add annotations which are private, visible to a group of users or public.

- Cross referencing documents, Using simple text analysis tools it is possible to take advantage of the fact that all documents are in the same database and that we can use a common dictionary and thesaurus to link keywords to their definition and to their references in other documents [Fig. 5]. The whole process should be automatic so that when a user views a document, the keywords that are in the dictionary appear as links which, when selected, display the list of definitions and references. Selecting a reference displays the related occurrence of the keyword in the referenced document.
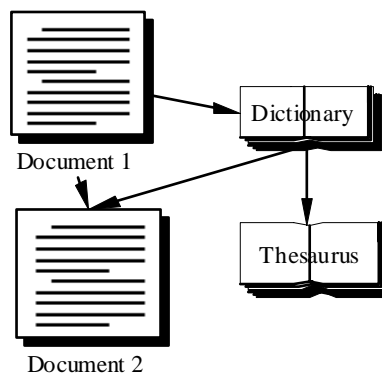

Document 1

Dictionary

Document 2

Thesaurus

Figure 5: Cross-referencing documents

11 Medium as an Education Tool

As a companion for a classical university course, Medium could be used by the professor and assistants in order to illustrate the course with examples, provide errata and even ask students to supply their own examples as annotations of the basic course. A typical scenario involves the professor making their course material available for annotation. Students then read the material at their leisure and annotate it with their questions and comments. The professor and assistants periodically browse through the annotations, answer questions and take action on comments by inserting their own annotations or through email if necessary. Since annotations can be other than text, it is possible for users (professors, assistants and students) to illustrate the course with examples, which can be graphical and even interactive.

12 Conclusions

A basic version of Medium is now available with which we are satisfied from both the user interface, performance and stability points of view. We have had less luck with participation in Medium. Most people who were granted access rights never quite made it to adding comments. The instability of the tool from the availability, performance and user interface points of view may have prevented some users from adding their comments, but we also believe that people are drawn back by the public nature of Medium and the strong context that exists between the annotation and the document as explained in [Annotations Versus News Groups]. with the solution being the scoping of annotations as explained in [Extending Medium]. This will be particularly important for the education field where students may feel uncomfortable with making their annotations public at first. Students may feel more comfortable adding their annotations while keeping them private and only make them public once they feel comfortable enough with the course material.

We hope to start several collaborations in order to evaluate the adequacy of Medium for higher education and for desktop publishing in a research institute. A public version of Medium is available on the World Wide Web at the following URL: http://sgwww.epfl.ch/medium

With hindsight, we think that the real potential of an annotation tool will be to offer it as a service. By service we mean a module with no user interface that offers annotation services to Web applications by exposing an API. By API we mean an HTML form based API.

13 References

[Drexler 1987] Drexler Eric (1987), *Hypertext Publishing and the Evolution of Knowledge*. Social Intelligence, 1 (2), 87-120. Web version: http://www.foresight.org/WebEnhance/HPEK1.html.

[Röscheisen et al. 1994] Martin Röscheisen, Christian Mogensen, and Terry Winograd (1994). *Shared Web Annotations as a Platform for Third-Party Value-Added Information Providers: Architecture, Protocols, and Usage Examples*. Technical Report STAN-CS-TR-97-1582, Stanford Integrated Digital Library Project, Computer Science Dept., Stanford University. November 1994, updated April 1995.

[Shipman et al. 1995] Frank M, Shipman III, and Catherine C. Marshall (1995), *Formality Considered Harmful: Experiences, Emerging Themes, and Directions*. Xerox PARC.

[Gaines et al. 1995] Gaines, B.R., and Shaw, M.L.G. (1995), *Concept maps as hyperdmedia components*. Int. J. Human-Computer Studies 43, 323-361.