

PDForum
A Threaded Interface for Collaborative Annotation of PDF Documents
Aditya Kalyanpur, Amir Khella
Department of Computer Science
University Of Maryland
College park, MD 20742
adityak@glue.umd.edu , akhella@cs.umd.edu

Abstract

The Portable Document Format (PDF) is currently becoming the standard for digital documents. One of the tasks regularly performed on digital documents is annotating and reviewing annotated documents. In collaborative annotation, the document is a heterogeneous collection of annotations by several persons, which at some level includes annotating previous annotations done by other persons.

Primarily, two different mechanisms exist for collaboratively annotating digital documents: the first method is using sticky notes, which are a collection of expandable layered notes drawn over the text using transparency. The second method is packing the annotations in a message-thread interface where they are grouped by the context of the discussion thread.

Our hypothesis is that a threaded interface will help users understand and critique collaboratively annotated documents better than layered interface. Our work was based on the Joanna Wolfe's studies which revealed three essential requirements for supporting effective collaborative annotation, namely, the need to allow authors to categorize annotations by positive or negative content, a means to provide information about annotators including their general position about a particular topic or text, and a mechanism that helps readers sort and filter annotations systematically

In this paper, we present PDFORUM, a tool for supporting collaborative annotation of PDF documents as a plug-in for Adobe Acrobat software. The tool provides the ability to annotate (using a type-based classification scheme) as well as rate (positively or negatively) documents contents and previous annotations. Finally the tool provides the ability to preview the annotations using color coding to provide a compilation of all the ratings specified by the annotators.

Keywords:

Digital documents, annotations, message thread interface, information visualization, PDF.

Introduction:

PDF and HTML are becoming the standards for formatted digital documents. More and more people use these documents everyday to augment their paper reading activity. The Portable Document Format is becoming a standard in the academic domain. Scientific papers, lecture notes, books are migrating into digital form in PDF format.

One of the frequent tasks performed on digital documents is annotating the contents and reviewing these annotations later for several purposes (skimming through, summarizing, overview, critique, reference, etc...). An experimental comparison of deep reading online and on paper found numerous differences in how subjects worked with the text: in general, working on paper seemed far easier [8]. This explains why information gatherers print documents to read them. Due to the high affordance and tangibility of papers as a medium, this is proven to be

efficient in non-collaborative tasks. Once the document migrates from one user to another, every person adds his own annotations and reads other people's reviews and annotations, some problems start to arise for the paper format: Reading other persons' handwritings, figuring out the chronological order of the comments¹ people scribbled in the margins or on the text, and most importantly, figuring out how the comments and annotations relate to each other. Adobe Acrobat comes with numerous tools for annotating PDF documents such as sticky notes, free form text, highlighters, free ink and others. Figure 1 is a screenshots showing some of these annotations on a PDF page.

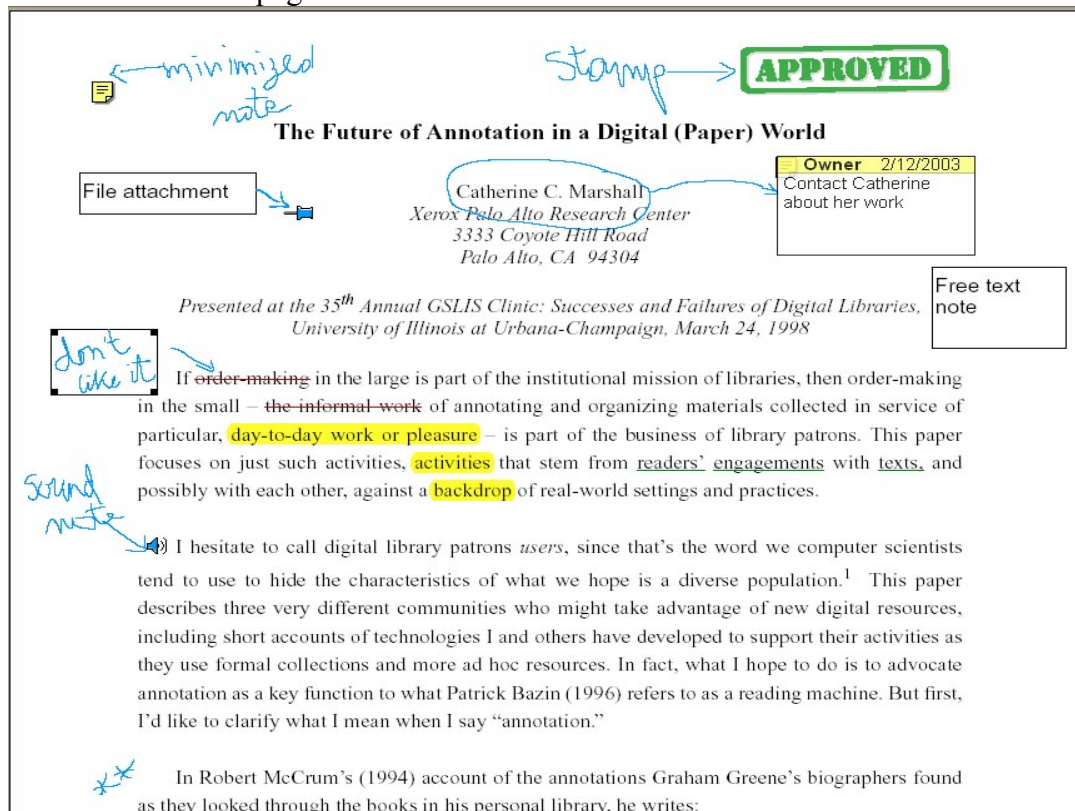


Figure 1: An annotated PDF page using several standard Acrobat tools such as ink, highlighter, sticky notes, free form text, voice, attachments and stamps.

The figure shows that there is a good variety of tools at the users' command to annotate the document. One of the powerful annotation tools is the sticky notes since they are expandable and they don't take enough space when they are minimized which gives the users the full unaltered view of the original document. Acrobat also provides users with several functionalities for filtering and sorting annotations such as grouping by users or pages, as well as hiding and showing specific annotation types.

Meanwhile, reviewing a document annotated using sticky notes becomes harder as the number of annotation increases. Figure 2 shows a page with 9 annotations on a specific paragraph. It is obvious that navigating these annotations requires a good deal of interaction to read through them: bringing to the front, minimizing and expanding, etc...

¹ In this paper, we will use the notion of comments and annotations interchangeably.

It is also obvious by looking at the figure below that figuring out the logical sequence and dependency relations between annotations is hard since the tool does not provide a view that supports this particular task.

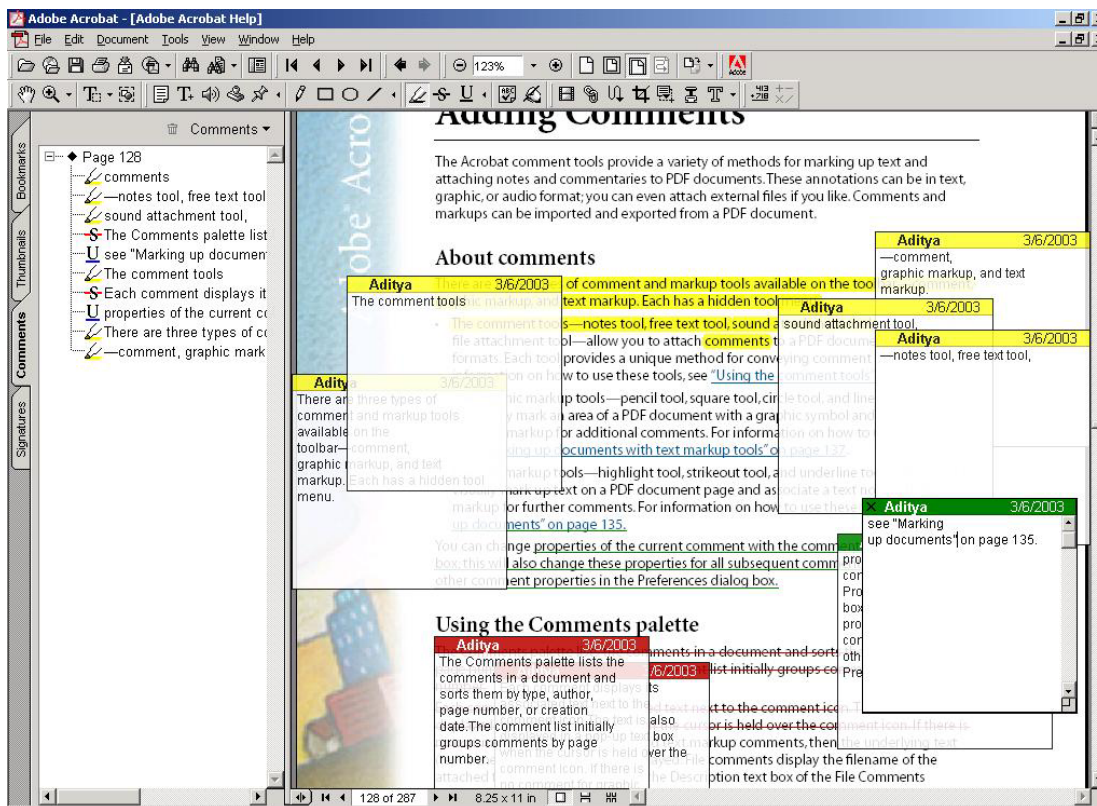


Figure 2: A page in Adobe Acrobat showing 9 annotations using sticky notes.

Related work:

The importance of annotations integrated within the textual content as opposed to response essays appended to the text was demonstrated by a study conducted by Joanna Wolfe [1] in which students were asked to read the material (along with the annotations/essays) and answer a questionnaire on their composing plans, goals and strategies. The study revealed that annotations helped the readers construct complex and integrated representations of the issue's problem space by increasing reflection and elaboration while reading.

Increasingly, however, documents exist in digital form. Supporting annotation of digital documents both for personal use and asynchronous collaboration has many challenges. In this paper, we focus on the challenges involved in the latter task. Joanna Wolfe, in her dissertation on the 'Pedagogical Uses of Annotations and Annotation Technologies' [2], studied the key attributes of the annotation process in order to arrive at a theoretically and practically sound annotation tool. Her study revealed three essential requirements for supporting effective collaborative annotation, namely, the need to allow authors to categorize annotations by positive or negative content, a means to provide information about annotators including their general position about a particular topic or text, and a mechanism that helps readers sort and filter

annotations systematically. Furthermore, various other studies ([3], [4]) reveal that reviewers doing collaborative annotations can greatly benefit from a discussion-thread like approach in which the context-based hierarchical organization of the annotations makes them easy to follow and process. Asynchronous threaded discussions have been used in a variety of University courses (Benbunan-Fich & Hiltz 1999, Chong 1998, DeLoach & Greenlaw, 2002) demonstrating that it is a powerful medium that provides both the interactive features and reflective qualities that aid collaborative discussion.

In order to support digital annotations, various technologies have arisen that mimic paper's flexibility, portability and seamless annotation capabilities as much as possible. Tools such as XLibris [5] and Notable [6], developed by Xerox, are examples of such technologies that allow users to make free-form annotations with a digital pen. Among the two, XLibris has gained popularity among lawyers who use its 'Reader's Notebook' functionality to review and organize the documents along with their ideas and comments, employing techniques such as using different colored pens to rate different types of annotations. However, these tools focus mainly on personal annotations and not much support is provided for shared annotations.

Similarly, Adobe Acrobat provides a wealth of annotation features that range from traditional techniques such as adding comments, highlights and sticky notes to specific parts of a PDF document, to more advanced tools that allow addition of voice comments and embedding files. Currently, users can share annotations globally using a network connection. However, the interface has no support for threaded annotations; text annotations are laid out in layers using transparencies. Furthermore, all the annotations are generic with no classification scheme. Hence, during collaborative review of a document, determining the correlation between the annotations with regard to their creator and context is inherently difficult because of the lack of a firm structural and logical relationship among the various annotations.

Annotea [7] is a web-based shared annotation framework that addresses some of the problems mentioned above for annotating web pages. It employs a discussion-thread like approach to layout annotations and classifies annotations into various types (questions/notes/explanations etc). However there exists no provision for the authors to rate the document content positively or negatively. Moreover, overall compilation of all the annotations to provide a final perspective cannot be performed easily.

Design issues:

The design of the tool was based on the facts provided by previous studies and literature about the use and presentation of annotations on a digital document. The main facts that shaped our design were:

- 1- Annotations should not be generic. Users usually have a defined purpose for typing an annotation, which can be a question, a comment, an example, an advice, etc...
- 2- Annotators should be able to provide a rating (negative, positive, or neutral) for the paragraph they are annotating or the previous annotation they are replying to.
- 3- Annotations on each paragraph should be shown in a message thread interface with the ability to create new annotations, reply to previous annotations and finally sort and search effectively.

- 4- Users should be able to get an overview about the overall rating of the annotations on paragraphs using a consistent and comprehensible color-coding scheme.

Initial design:

Based on previous work and studies mentioned earlier, our goal was to provide a tool that supports our main hypothesis requirements: the need to allow authors to categorize annotations by positive or negative content, and a mechanism that helps readers sort and filter annotations systematically.

Three main design issues came into focus concerning:

- 1- How users would enter their annotations for a particular paragraph
- 2- How users would review and reply to each other's annotations
- 3- How the ratings would compile into color(s) that represented its overall perspective

The initial design tackled the three previous issues by using two separate dialogs: one for entering comments and another one for reviewing comments. The dialogs give the user the following functionalities:

- Enter a new annotation with annotator's name, title, type, rating and text.
- Review current annotations, reply to an annotation or initiate a new thread of annotations
- View annotations in both hierarchical (threaded) view and linear view.
- Filter annotations by text in specific fields.
- Sort annotations by several criteria

We also compiled the ratings using color coding. Each thread contains a number of negative, neutral and positive annotations. Our initial formula compiled these three numbers into a position on the *green-to-yellow-to-red* color spectrum.

Initial design flaws:

The initial design implemented the required functionality but introduced some usability issues. Iterations using design standards and experiments performed on the initial prototype showed us some major flaws in the initial design:

- The use of two dialogs introduced an overload on the user's part. The interface needs to be compact, present as a floating single dialog and not hiding the document's text.
- Color coding was not intuitive. Humans can not pre-attentively process the color and estimate the position on the color spectrum. The color should reflect the ratio of neutral, negative and positive annotations in an efficient way and still easy to process quickly.
- The dialogs contained several interface widgets that were not necessary to the tasks.

Final design:

To solve the flaws in the initial design, we iterated through the design and came up with the final design shown in figure 3. The main modifications to the initial design were:

- Merging the two dialogs used to enter and review annotations into a floating dialog that provides users with the same functionality and yet requiring the minimum amount of interaction using keyboard shortcuts, automated fields filling and default focus switching. Our initial goal was to implement a tabbed pane that complements the existing floating tabbed panes in Adobe Acrobat. Due to some restrictions in the Acrobat SDK, we implemented the interface as a minimizable floating dialog.

- Removing unnecessary widgets from the dialogs such as comment title in order to be consistent with the standard annotation structure in Acrobat. The annotation's title in the threaded view is set to be the first 'n' characters of the annotation's text.
- Devising a new way for color coding: a bar at the left side of the rectangle surrounding the annotated text. The bar shows the three colors (yellow, green and blue) scaled to show the ratio of the corresponding rating to the total number of comments on this paragraph. The color coding counts only the root level annotations, and not the ones that are replies to previous annotations. As an example, if the root level contains 10 annotations, three of which are neutral, five are positive and two are negative, the color bar will be divided into three vertical sections colored: green, yellow and red occupying 50%, 30% and 20% respectively of the vertical scale.

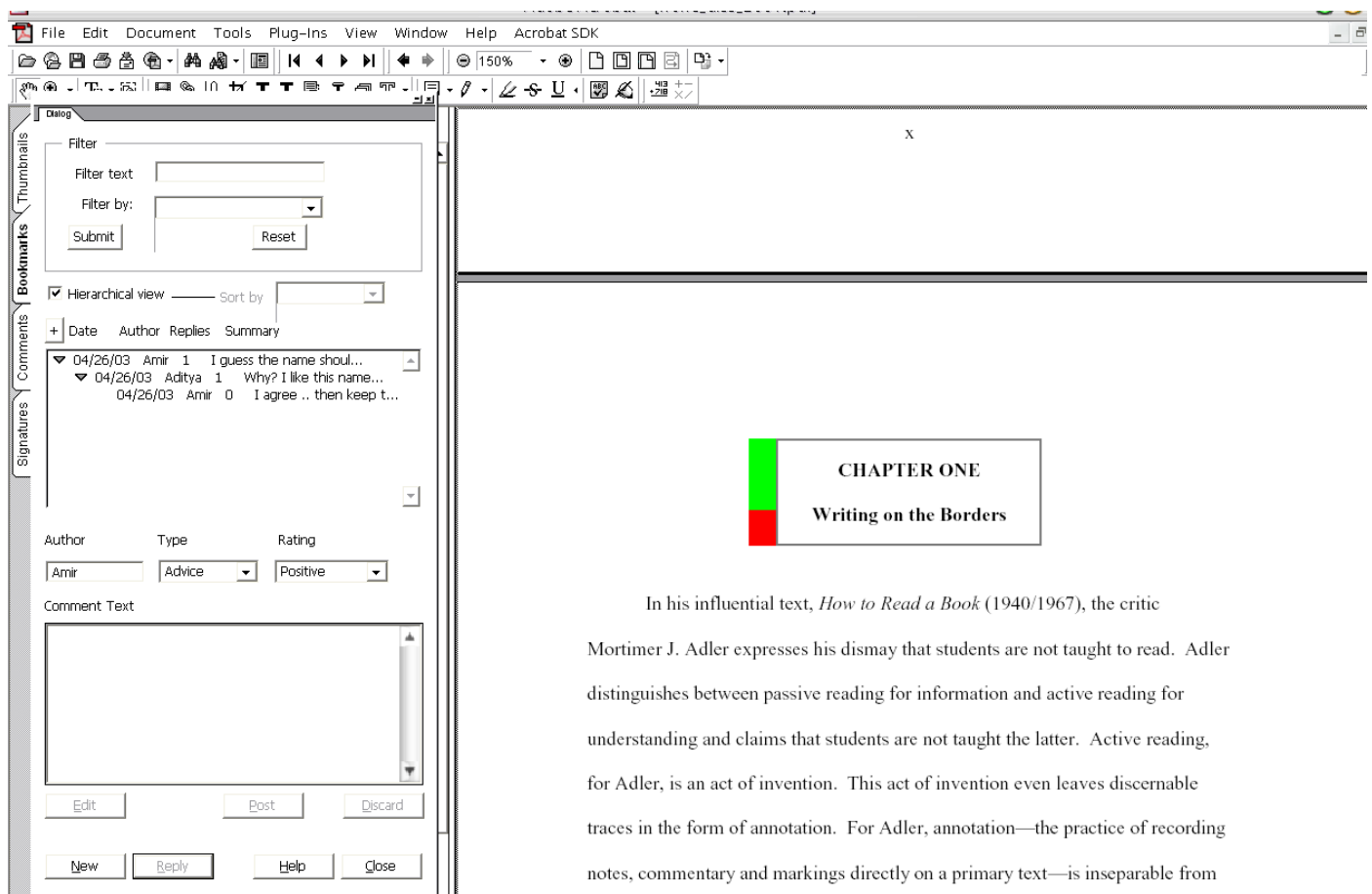


Figure 3: Screenshot showing the final design. The panel on the left appears when the user first selects a paragraph with the annotation tool and is always floating inside the left pane. The screenshot also shows the current color compilation of two positive and one negative annotation on the chapter's title.

A typical usage scenario:²

This scenario highlights an author writing the draft for his new book. Upon completion of his write-up, he uploads his book to his account on the document sharing server, and selects a list of people who will be reviewing his draft. The system sends notifications to these persons informing them that a new document is available for check-out and review.

These reviewers start interacting with the document by checking out the chapter of interest, making the necessary annotations and checking it in again. As they review the document, they start analyzing each others' annotations, reading what other reviewers have to say about a particular paragraph and posting replies and comments about what they agree or disagree with regard to the other comments. The author interacts with the document as well, posting his own annotations and replies to fellow reviewers to explain his point of view if and when clarification is needed.

After the reviewers finish their work, the author checks out the book and starts reviewing it chapter by chapter. For each chapter, he starts by getting an overview about the overall rating of each paragraph by compiling the annotations into color codes. He focuses on the paragraphs that have the most negative annotations since these are the ones which will involve considerable rework. Then he goes through each annotated paragraph and reads through the annotations, reviewing what reviewers said about it and how they reacted to each others comments, sorting messages by number of replies, authors, or ratings and performing some form of history replay mentally to figure out the flow of collaborative interaction with this paragraph over time. The author's main aim is to arrive at an overall consensus of the reviewer's opinion of his work in order to make the necessary suitable modifications.

Pilot User Study 1: Testing Interface Usability

We conducted a pilot user-study to test the usability of our Adobe Acrobat plug-in interface. Two subjects were selected for the experiment – a male and a female. Both had experience using Adobe Acrobat Reader (without the annotation features). Subjects were given a tutorial on the usage of our Adobe Acrobat plug-in. Each subject was then assigned two tasks that involved reading a document titled '*Debate Surrounding Human Cloning*' and using our plug-in to annotate it. Finally, each subject was given a questionnaire to fill based on the tasks performed.

The first task involved annotating a document from scratch i.e. the subject had to use our plug-in to create new annotations and threads and subsequently rate different sections of the document separately. The purpose of this task was to test the efficiency with which the subject could use the features of our plug-in. The second task involved analyzing an annotated document i.e. the subject was given an annotated version of the same document that contained message threads by four fictitious authors. The author's annotations were made to reflect their nature and overall opinion of the document. For example, one of the authors was inquisitive; another was passive; while the remaining two agreed/disagreed with the concept described in the document respectively. The subject was asked to correctly identify the overall opinion (impression) of the four authors and write a short review based on their annotations. The purpose of this task was the test the usefulness of the various features of our plug-in.

² In our scenario, we assume the existence of a document sharing server that allows users checking in and out documents of interests for the purpose of annotations with the necessary sharing semantics implemented.

We made several observations during the experiment, such as noting how well the subject understood the working of our interface, which tasks seemed complicated and confusing, and how satisfied the subject seemed with the functionality of our interface (based on his/her behavior and facial expressions).

Finally, based on these observations and the subsequent questionnaire that the subject filled out, we drew the following conclusions:

- *There were certain bugs present in our interface* (such as invoking the dialog did not clear the annotation attribute fields, the no. of replies in a message thread were displayed wrongly etc)
- *Overall, the subjects were satisfied with the layout and functionality of our interface:*
 - Average rating for overall feel and impression of the interface: 7.5/10
 - Average rating for usefulness of the features: 8/10
 - Favorite feature: Color-coding
 - Both subjects could correctly analyze the overall opinion of the four authors in the second task
- *Various suggestions were made to improve the interface* (such as the need for a horizontal scroll bar in the threaded message pane, a priority setting mechanism to rate annotations separately from content etc)

We used the results of our pilot study in future modifications to the plug-in interface.

Pilot User Study 2: Comparing our Plug-in with Adobe Acrobat's current Annotation Framework

We conducted a second user study to determine how our plug-in fared against the existing annotation framework of Adobe Acrobat. The scenario we chose to simulate was that of an author who had received an annotated version (annotated by six peers/reviewers) of his book's rough draft. The author had to review and assess the annotations made by the other authors and summarize their overall opinion.

As in our previous study, two subjects were selected for the experiment, one male and one female. Each subject was given two tasks, however the sequence in which the tasks were assigned was different for both the subjects. The first task involved using the existing Adobe Acrobat annotation framework to review the annotated document, while the second task involved using our plug-in on the same annotated document. In order to keep an unbiased view, the subjects were not informed about the creators of the interface. In both cases, the annotated document contained annotations and ratings made by six fictitious authors, and the subject had to read and assimilate the annotations in order to arrive at a consensus. Finally the subjects were asked to fill out a questionnaire (a sort of comparative analysis sheet) to ascertain their opinion on the ease of use and functionality of the two interfaces presented to them.

The results of the experiment firmly established our hypothesis that employing a discussion-thread like interface (with ability to classify annotations by type and rate text specifically) was

more beneficial to collaborative discussion than a simple layered-transparency approach. Both the subjects noticed striking differences between the two interfaces and rated our plug-in higher than the current Adobe Acrobat plug-in in three key respects – look & feel of the interface, ease of use and functionality. In their opinion, the main advantages of our plug-in in supporting the task given, was the ability to:

- a. Easily identify the nature of the discussion (using the classification criteria)
- b. Determine individual opinions/sides related to a discussion quickly (using the color-coding scheme)
- c. Logically follow the discussion to arrive at an overall assessment (using the threaded interface)

Future work:

We plan to conduct a full-scale user-study on a more diverse set of tasks and scenarios involving collaborative annotation of digital documents to further validate our hypothesis.

Conclusion:

In this paper, we presented a new collaborative annotations framework for Adobe Acrobat based on findings in related studies on annotations. Our main contribution was providing a user friendly, portable, server independent threaded interface for annotations on PDF documents. The findings from our user studies supported our initial hypothesis that threaded interfaces are more suitable than layered interfaces for reviewing collaborative annotations.

Acknowledgments:

We would like to thank Professor Francois Guimbretiere for his continuous help and feedback throughout the project. We would also like to thank the subjects of our pilot experiments who helped us get timely and valuable feedback during the design process.

References:

- [1] Joanna Wolfe, *Arguing across the margins: How text layout affects reading-to-write strategies*
- [2] Joanna Wolfe (2001), *Pedagogical Uses of Annotations and Annotation Technologies*
- [3] Donald Weasenforth et al, (2002), *Realizing Constructivist Objectives through Collaborative Technologies: Threaded Discussions*
- [4] A J Bernheim Brush et al (2002), *Supporting Interaction Outside of Class: Anchored Discussion vs. Discussion Boards*
- [5] Bill Schilit et al (1999), *XLibris: The Reading Appliance Revolution*
- [6] Michelle Baldonado, Steve Cousins et al (2000), *Notable: At the Intersection of Annotations and Handheld Technology*
- [7] Jose Kahan et al (2001), *Annotea: An Open RDF Infrastructure for Shared Web Annotations*
- [8] K O'Hara. and A Sellen (1997) A Comparison of Reading Paper and On-Line Documents. *Proceedings of CHI97* (Atlanta, GA, March 1997), ACM Press, pp. 335-342.