Journal of Universal Computer Science, vol. 6, no. 12 (2000), 1197-1202
submitted: 5/9/00, accepted: 20/12/00, appeared: 28/12/00 © Springer Pub. Co.

Active Documents: Concept, Implementation and Applications

E. Heinrich
(Massey University, Palmerston North,
New Zealand
E.Heinrich@massey.ac.nz

H. Maurer
(Graz University of Technology, Graz, Austria
hmaurer@iicm.edu)

Abstract: In this paper we present the notion of "active documents". The basic idea is that in the future, users of documents in any networked system should not just be able to communicate with other users, but also with documents. To put it differently, we believe that communication in networks should be understood in a more general sense than it usually is. Although our notion will, at first glance, almost look like science fiction, we will show that good approximations can indeed be implemented. We conclude this short paper by pointing out a number of important applications of our new concept and mention cases where it has already been successfully applied.

Key Words: eLearning, WBT, web based teaching, knowledge management
Categories: H.4, H.5, J.7, K.3

1 The notion of active documents

The idea behind active documents is very simple: whenever a user sees some information on the screen of a networked computer the user can ask (typically by typing) an arbitrary question ... and the document immediately provides the relevant answer.

Putting it this way, the notion sounds impossible to implement (how can a document answer any conceivable question?), or at least sounds like an idea out of science fiction. However, although the idea cannot be fully implemented it is surprising how well it can be approximated in large scale distributed networks. The reason for this is that important documents are viewed by a very large number of persons, hence the same questions (albeit in possibly different wordings) come up over and over again. Indeed, experimental data with one of the first users of the Hyperwave eLearning Suite [2, 5, 6] confirms that after some 500 to 1000 users new questions come up only exceedingly rarely. Thus, the basic implementational idea behind active documents is simple: when new documents are added to a server, questions asked are initially answered by experts, ideally but not necessarily immediately and around the clock. All questions and answers are recorded in a database. As a new question is asked it is checked whether this question is semantically equivalent to an earlier one: if so, the system can provide the answer immediately. As the number of semantically new questions decreases (and indeed approaches almost zero) the experts become superfluous. In the very rare case that a really new question arises the system might answer apologetically: “This is a very good question. We will forward it to our experts and you will receive an answer by xxx the latest”.

The crux of the matter is, of course, how is it possible to determine if two questions that may be worded quite differently are indeed semantically equivalent. How, to just present one example, can any software recognize that the question “Please explain to me how the picture compression techniques GIF, BMP and JPG compare to each other” is really equivalent to “I do not understand the difference between GIF, BMP and JPEG coding”.

In the next section we show that this task, as hopeless as it might seem at first glance, has indeed both pragmatic and systematic solutions.

2 The implementation of active documents

2.1 The heuristic approach

When a question is asked it is compared with some heuristic algorithm to find an earlier question that seems to be similar. Similarity can be determined by using a number of techniques that can become quite sophisticated: from comparing if different words match independently of their order (like GIF and BMP in our example), to the use of synonyms (like JPG and JEPG), to stemming algorithms that take care of the flexion of words, to semantic nets, to syntactic analysis using such nets, a variety of techniques do exist that are surprisingly powerful. A typical example is the “LIKE” operator in the well-known search engine Verity that is able to identify with a high success rate pieces of text that are likely to have the same meaning.

In each case, once the system has determined that the question x being asked is likely to be the same as a question y asked earlier, the system will say “Do you mean y”. It is now up to user to decide whether indeed the answer to this question y is what is desired or not. If not, the system may offer other alternatives, but if none satisfies the user, the apology mentioned earlier: “This is a very good question. We will forward it to our experts and you will receive an answer by xxx the latest” will be shown. As time goes by (in the sense that a document is visited by more and more persons) chances that a question is asked that is not only identical semantically but is also similar in form decreases, thus reducing the amount of time a human expert has to help out, and hence reducing the number of times a question is not answered immediately.

2.2 The iconic approach

Users ask questions by marking (with the cursor) some part of the screen (a formula, an abbreviation, a picture, etc.): their question refers to information in the marked area. When a question has been answered, some icon or highlighting shows to other users that other persons have asked questions concerning this piece of information and that experts have answered them. If another user also has a question concerning the material at issue, one click suffices to show all questions and answers that have occurred so far. Again, after sufficiently many users the chances are good that all questions of interest have indeed been answered.

The iconic approach is clearly particularly easy to implement and has the advantage that semantically equivalent questions that are formulated in different ways will not often arise. Also, the advantage of this (and the other techniques relating to active documents) is that feedback is provided to the authors of documents as to where users have questions. After all, this may often mean that some explanation is not clear enough, information is missing, etc., hence allowing the improvement of the documents. The iconic approach has been sometimes belittled by just saying that this is not more than FAQ's, and in a way it is, of course. However, the FAQ's are not collected in a long unusable list but in a short list directly where the problems occur. If that list gets too long, something is likely to be wrong with the document, and the document should be improved.

2.3 The linguistic approach

The most satisfying approach to handle active documents would be to develop techniques that actually prove the semantic equivalence of questions. Remember, heuristic techniques can only provide guesses whether two questions are equivalent, they cannot prove their equivalence. Remember also that both the heuristic and iconic approaches use the intelligence of the user to determine whether previously asked questions are relevant or not.

To actually prove that two pieces of text are semantically equivalent one would require a complete understanding of natural language, something still quite elusive. However, we can consider a compromise: rather than allowing a full natural language we restrict our attention to a simplified grammar and to a particular domain for which an ontology (semantic network) is developed. Clearly, sufficiently restricting syntactic possibilities and terms to be used will allow to actually prove the equivalence of pieces of text. This approach has been investigated in the first author's Ph.D. thesis (for somewhat different purposes [1]). Attempts to adjust it to the active document situation are currently being carried out [7]**.** At this point in time the restriction on the wording of questions and the domain specificity are serious problems. It seems clear that this technique will not be suitable for the naïve user, yet (as we will mention in the application section) there may well be situations when this linguistic approach is also useful.

3 Futher research

Before turning our attention to the application of active documents let us point out that the concept is being applied successfully right now, yet there is room for much improvement. In the heuristic approach mentioned, algorithms trying to guess the equivalence of textual pieces still can and should be improved by incorporating both more powerful semantic networks and more syntactic analysis. Observe that the efforts in this area are, unfortunately, also quite language dependent! In the linguistic approach there are two somewhat contradictory aims that one still has to go after: on the one hand, the query syntax should be as natural as possible, on the other hand domain dependence should be reduced and the construction of the domain specific ontologies should be simplified.

However, there are also other more intrinsic problems with active documents. We have stated that “really new” questions usually do not arise after a document has been used by some 500 to 1000 users. This figure comes from a large eLearning experiment with a multi-national company with some 200.00 employees, but is quite unrealistic in other contexts. Typically, the WWW or Intranets also contain information that changes over time. What we have discussed so far does not handle this situation at all. Just consider a page about skiing in Austria. At some stage someone has asked “How much snow can I count on” . The answer, whatever it may have been, to exactly the same question is likely to be different as soon as a day later. It thus seems to appear that the notion of active document is only applicable to fairly static information, and question-answer dialogues should be time-stamped so that they disappear automatically when they are invalid. Actually, the situation is better than this: an indirect step can alleviate the problems in some situations in a very elegant way. To be specific, the answer to the question “How much snow can I count on” should not be “30 inches” or such but rather “Find information on current skiing conditions under ‘snow report’ “ , where ‘snow report’ provides a link to another server that has indeed up to date information on snow depth, temperatures, etc.

Another fairly deep problem is that the same question may require different answers depending on the circumstances. Again, an example shows best what can happen. The question “Please explain ISDN” could have as answer “ISDN stands for Integrated Services Digital Networks” and this might be sufficient for some persons. Others might expect much more than the explanation of what the abbreviation stands for! Future active documents should probable reply along the lines: “Do you want to know what the abbreviation stands for (choice a), do you want a short technical explanation (choice b), or a fairly detailed explanation (choice c)”. Clearly this is more work for the experts involved, it tends to blur the border between simple question and answer situations and eLearning on a broader scale, but it also shows that our initial claim that persons should be able to really communicate with documents is not that far fetched.

Overall, the deeper one digs into applications of active documents the more does it become apparent that in a way much of the knowledge that now goes into more clever search engines, into language analysis and intelligent agents will also apply to active documents.

Let us conclude this section by just making sure that two further points are understood: first, active documents are not supposed to be restricted to textual questions and textual answers: it is easy to allow arbitrary multimedia activities as answers to questions, and it is certainly conceivable to allow questions posed in the form of speech input or such, even if this is likely to make the recognition of semantically equivalent questions still more complicated; secondly, we have always said that questions are originally answered by “experts”. Surely also other users may answer questions being posed, yet somehow the level of competence of the person answering a question should be known to users.

4 Applications of the notion of active documents

We have encountered the notion of active documents the first time in connection with eLearning experiments [5, 2] and have found them to be very useful in this connection. Note that the "critical number" of users of the same document (we have quoted 500 - 1000 a number of times) will usually only be reached if the courseware at issue is offered on the WWW for a large audience, or in a substantial intranet, but will usually not be reached when using material for e.g. teaching typical university classes. However, even in this case, the mechanisms described eliminate many duplicate questions and are hence useful even if the "saturation level" is never achieved.

It has become clear over time that the usefulness of active documents goes far beyond eLearning. Indeed we would like to claim that in the future every WWW or intranet system should assure that ALL documents are active. This helps users (who are otherwise frustrated by having to send emails when they have questions, often not knowing whom to send them to), eases the support work for those offering information or services, and provides valuable feed-back to the administrators of sites as to where users have problems.

However, in addition to eLearning there are three other areas where active documents seem to be particularly useful: one is support for software, the other is in connection with help desks, and the third are digital libraries. Let us elaborate this again by means of examples.

Suppose a company releases a new software product to a large number of (pilot) customers, with the corresponding documentation on the WWW. In the past, support staff would always receive a stream of questions of the type "On page x , line z from the top, in volume y of the documentation it says that the SW should act according to w, but it does not. Can you please help?". Support staff would then have to consult volume y, go to page x, count to line z and examine the situation, only to find out that this bug had been pointed out many times before, and that the development team was already in the process of fixing the bug (or documentation). If each page of the documentation is an active document this situation does not occur: after the problem at issue is pointed out the first time, other customers have no need to ask the question any more, helping them and the support staff.

The situation is similar in help-desk situations when customers do not understand a manual or such. There are two interesting additional aspects in this case, however. First, customers may ask the question not via an active document on the WWW but by telephone. Help desk staff may, however, now use the active document to find the answer to this question: after all the question may have been answered by a colleague at some earlier stage. Second, in this case the linguistic approach mentioned could be handy: the domain is limited, and the staff of the support center may well be expected to be able to translate customer queries into queries allowed by the linguistic approach. Thus, even a help desk person not capable of answering a question x, can translate it into x' for the system, and the system might provide the answer that is then communicated to the user.

We have argued in a separate paper [3] that active documents will also play a fundamental role in digital libraries in the future, since those libraries will turn more and more from static information repositories into interactively and collaboratively used centers of human knowledge.

5 Conclusion

In this paper we have described the new notion of "active documents". We believe that already the first applications make it abundantly clear that this notion will play a major role in the intelligent exploitation of information and knowledge in computer networks.

Much research and development remains to be done, yet even with what is available today in e.g. [2] much can be achieved!

References

[1] Heinrich, E., Patrick, J., Kemp, E.: A Flexible Structured Language for the Coding of Human Behaviour – FSCL, Massey University Department of Information Systems Report 1 (1999)

[2] www.hyperwave.com

[3] Maurer, H: Beyond Classical digital Libraries; Proc. NIT conference, Beijing (2001), to appear.

[4] Lennon, J.: Hypermedia Systems and Applications; Springer Pub. Co., Germany (1997)

[5] www.gentle-wbt.com

[6] Maurer, H.: Hyperwave - the Next Generation Web Solution; Addison Wesley Longman (1996)

[7] Maurer, H.: Hyperwave - the Next Generation Web Solution; Addison Wesley Longman (1996)