

I-DIAG: From Community Discussion to Knowledge Distillation

Mark S. Ackerman^{1,2}, Anne Swenson², Stephen Cotterill², and Kurtis DeMaagd³

¹Department of Electrical Engineering and Computer Science

²School of Information

³Haas School of Business Administration

University of Michigan, USA

{*ackerm, aswenson, scotteri, demaagdk*}@umich.edu

Abstract. I-DIAG is an attempt to understand how to take the collective discussions of a large group of people and distill the messages and documents into more succinct, durable knowledge. I-DIAG is a distributed environment that includes two separate applications, CyberForum and Consolidate. The goals of the project, the architecture of I-DIAG, and the two applications are described. We focus on technical mechanisms to augment social maintenance and social regulation in the system.

Introduction

Imagine the following scenario: The president of a large public university in the US asks a blue-ribbon panel of his highly regarded faculty to reflect upon the future of their university. The president wants to keep their university not only in the forefront of similar universities but also in front of basic societal pressures and opportunities. However, the faculty are also admonished to consider the various often-overlooked stakeholders – the university’s staff, undergraduate students, graduate students, alumni, non-tenured instructors, state legislature members, and local community residents. A large US state university may have several thousand faculty members, and the various concerned stakeholders might include 50 thousand or more people. Of course, the faculty committee could do as a

typical blue-ribbon panel often does, going into their respective rooms to inscribe their already acquired expertise. But if they wished, how might they reach out to these stakeholders, include their perhaps divergent opinions, and search for new and interesting opinions and options?

We know that Internet-scale systems can provide forums for large groups (> 10⁵ people) to gather, discuss, and trade ideas. Within a corporate setting, these systems can be used for brainstorming, new produce ideas, quality circles, and the like. Governments, institutions, and universities can discuss such issues as organizational change and future plans in order to come to a “shared mind”.

Yet all too often problems arise in these attempts. People do not come to the site, or do not stay on topic. More importantly, once use has finished (either by deadline or by neglect), the site is often a bramble of ideas and topics, too large and unwieldy for its information to be successfully reused.

Our system, I-DIAG¹, investigates how to garner and then distill this valuable community knowledge. It is part of a larger project to investigate how to maintain and reuse informal information within organizational and Internet-scale settings.

The paper is arranged as follows: We begin with a description of the research problems under consideration, and follow that with a brief overview of the relevant literatures. We then discuss the architecture of I-DIAG as well as provide a description of the various components of I-DIAG. (I-DIAG consists of a number of applications and distributed services.) This is followed by a discussion of three facilities to augment important social aspects of I-DIAG’s use – social maintenance, social facilitation, and social regulation. We conclude with future work and directions.

Research Overview

We created I-DIAG to consider several general research problems as well as provide a concrete application with which to examine these problems. Overall, we are investigating:

- New models for refinement and distillation. Our primary interest is in finding social and technical mechanisms to facilitate the distillation of knowledge from large amounts of informal information, such as bulletin-board messages, chat messages, e-mail, or quickly written brief documents. Our argument below is that previous mechanisms have failed because of the

¹ The main quad of the University of Michigan campus is called the Diag. I-DIAG is also short for Interactive Diagenesis. Diagenesis is “the recombination or rearrangement of constituents (as of a chemical or mineral) resulting in a new product, or the conversion (as by compaction) of sediment into rock (Webster's 1986)”.

social barriers. Accordingly, our emphasis is less on the technical mechanisms for doing textual summarization or knowledge elicitation than on finding social models with augmentative technical mechanisms to foster the creation of material and then “boiling down” of that material into something that will be subsequently useful to others.

- These “boiled down” repositories are the distilled and refined versions of many people’s thoughts about a subject, mostly likely specific to a particular socio-technical environment. We are also investigating mechanisms to foster the sustainability of this distilled repository over time.
- In any social space, mechanisms must exist to foster social regulation and sustainability over time (as in Ackerman and Palen 1996). While social regulation can have pejorative connotations for computer people, some amount is necessary to continue any collectivity’s activities. It seems as though there are always problem or abusive users in online spaces. We also wish to prevent or ameliorate unproductive or hateful exchanges. As we will see, the duration for I-DIAG is very short. Nonetheless, there are still social regulation and maintenance issues to be resolved; indeed, some may be exacerbated by use assumed to be brief. Through I-DIAG we are investigating collaboration-centric mechanisms to quickly move users into an understanding of the system and its uses, enable productive exchanges, and control potentially unruly users and problematic exchanges.
- Since we hope that use is rapid and the corpus of information is constructed very quickly, we are investigating interface mechanisms to allow users to return to the space and understand what is new quickly and effectively. We hope to produce interface guidelines for these types of spaces.
- Overall, we see ourselves as investigating new forms of knowledge management. I-DIAG forms an interactive or dynamic “book”, where the corpus is constructed iteratively and collaboratively by people with different opinions, types of expertise, and varieties of experience and viewpoints. This “book” is a living document – not only is it constructed by people in terms of their own interests and knowledge, but it can be maintained over time in the same manner.

Our major goal, then, is to understand how to iteratively construct a refined knowledge repository (probably less than completely formalized but more distilled than raw messages). To do so, we must necessarily also investigate what technical *and* social mechanisms we need for sustainability, social regulation and maintenance, navigation and return, and interface metaphors.

In order to examine these broad issues, we have created a particular problem scenario and the computational system to solve it. The scenario in the introduction describes most of the problem we are addressing. It is a “brainstorming system”, a system in which people can come together to offer

ideas and debate them. I-DIAG, then, is the specific testbed we have created to investigate these issues. We have simplified our application and its environment:

- In keeping with the Internet philosophy of utilizing many eyeballs, I-DIAG attempts to harness small amounts of time from users. Motivations for using the system come from everyday activity. Within a given organization or community, we hope to have some small number of core users who will be key contributors, but we expect small contributions from a much larger number of people. At the end, we expect only a handful of people to distill the material.
- In our standard scenario of use, we are assuming the site will be used actively for a brief period of time – two weeks in our current plans. This allows people to have a healthy and vigorous discussion on specific topics, and then the site can close down before the topic becomes obsolete or stale. It also provides us a time to start mining the discussion as a final product – namely the final report and/or a distilled, concise web site of responses and ideas.
- I-DIAG, accordingly, has three sets of users. The first user group consists of the people entering their comments and discussing appropriate topics. In general, these people will be from a specific organization, institution, geopolitical community, or scientific community. The second user group consists of the moderators, editors, and wizards who control the interactive discussion. The final set consists of the people distilling the archived materials, either for an external report or to create a more concise site.
- The precise outcome of any given I-DIAG installation may not be known in advance. Some communities may wish a linear book as their outcome. The distillation process for a linear book would likely be different than when one wishes a concise site as the outcome. In addition, the scope of the distillation might vary – some sites may wish to include every point of view and every significant issue; other sites may wish to merely keep subdiscussions or interesting points.

I-DIAG, then, is an attempt to reconsider knowledge management and knowledge communities. It attempts to create incentives for use and reuse by differing groups of people, all of whom iteratively construct the space and the knowledge through their activities.

Relevant Literatures and Related Systems

Several diverse literatures bring appropriate insights and prior work.

Of direct relevance here are a number of approaches to distillation and summarization. In an older Education literature, one can find descriptions of “advanced organizers,” organization tools for structuring educational lessons or

materials (Jonassen, Beissner, and Yacci 1993). Although over time, the term came to be known as a technique for textual or oral materials (similar to foreshadowing), originally these included visual organizers. These visual organizers included timelines, web of relationships, trees of concepts, and the like. Many of the visual interfaces are directly relevant to our efforts to provide organization tools to users; however, these visual interfaces, we feel, are only part of what is needed.

Similar in intent to the literature on visual organizers is an important research stream on incremental formalization (Shipman and Marshall 1999, Shipman and McCall 1999, Shipman and McCall 1994). Visual organizers allow one to slowly increase the amount of organization in one's material by presenting more conceptually-oriented views on that material. This idea has been generalized in Shipman and colleagues' hypertext work. These papers argue that one should consider how to allow incremental formalization over time: Users can enter free text initially and slowly increase the level of organization and formalisms in their material. By allowing them to choose how and when to formalize their material not only is the system easier to use, users are more motivated to provide material. Incremental formalization is critical to how I-DIAG works.

As well, I-DIAG uses techniques derived from and similar to text summarization. Text summarization (e.g., Radev and Hovy 1999) attempts to consolidate large documents or sets of documents into abstracts or shorter documents. They do this through partial natural language understanding, taking the material in a document or set of documents and creating an abstract or summary. Many of the techniques are relevant to I-DIAG, but again these techniques are only part of what is needed.

I-DIAG is related to a number of different Computer Supported Cooperative Work (CSCW) systems (also called collaborative systems here). I-DIAG is obviously an e-community system. E-communities have been largely studied for their social effects (e.g., Sproull and Kiesler 1991, Wenger 1998). Emphasis has been on the social norms of use (e.g., flaming in Sproull and Kiesler 1991), character formation (e.g., alternative personalities in Turkle 1995), and communications (e.g., shared language in Cherny 1995). Much of this research is summarized in Preece 2000.

These systems do not have a large technical research literature. There is a literature on group communications, which concentrates on low-level distributed protocols or on construction flexibility. There is also a literature on visualizations (Xiong and Donath 1999, Smith, Cadiz, and Burkhalter 2000). There is, however, a considerable practitioner base and understanding, summarized again in Preece 2000.

Even fewer studies detail how the social and technical aspects of e-communities are related. Ackerman and Palen (1996) studied the Zephyr Message System, a shared chat system used at MIT. The study outlined the basic

norm and reward structures for the Help Instance, which is one Zephyr channel. Participants were able to use Zephyr, because of these social structures and a set of reinforcing technical affordances. Still, the Zephyr system was extremely simple, consisting of message scrolling by in a tty window. In the I-DIAG work, we are searching for additional, flexible types of support for e-communities.

I-DIAG also has similarities to a variety of brainstorming systems that have been investigated over the years. Generally, most such systems have been deployed and studied within face-to-face and distributed work meetings. Group support systems (e.g., Nunamaker et al. 1991) and meeting support systems (e.g., Streitz et al. 1994), to our knowledge, have been limited to either single-session meetings or small groups. They suggest, nonetheless, the value in computer support for brainstorming. A number of studies have shown that the use of these systems provides more ideas and more creative insight to a problem (Dennis et al. 1999). However, since the use of these systems has been limited to single-session meetings, little has been studied about the social structures of use over time, or the technology and human-computer interface mechanisms required to support that use over time.

One large-scale brainstorming system reported in the research literature was the White House 's Open Meeting on the National Performance Review (Hurwitz and Mallery 1995). Using the system, users "discussed, evaluated, and critiqued recommendations by linking their comments to points in the evolving policy hypertext." The message were typed according to an ontology, forming the potential basis of a discussion distillation. However, it is not clear, from the paper, that any further work, such as distillation, was done with the messages.

Indeed, evolving discussions of the sort in Hurwitz and Mallory or in I-DIAG could serve as a rudimentary design or decision rationale system (Conklin and Begeman 1988, MacLean et al. 1990, Moran and Carroll 1996). In a decision rationale, users categorize their points according to an explicit ontology concerned with discussion, technical design, or decision-making (e.g., gIBIS in Conklin and Begeman 1988 or QOC in MacLean et al. 1990). This is combined with an implicit social process in order to create a coherent, well-structured argument that can be viewed by others at a later time. The goal is to help future readers understand a decision or design, and perhaps reuse portions of the rationale in their own subsequent design or decision processes. However, as Grudin points out in Moran and Carroll (1996), users must do considerable upfront work for an unclear future payoff. Indeed, most attempts to use rationale systems show that users are reluctant to go to the extra work to construct detailed, formalized rationale arguments. Accordingly, I-DIAG attempts to provide suitable incentives for all of the users of the system by separating the argumentation from the distillation. The message database in an I-DIAG installation is created because users want to discuss a problem; the users do not have to categorize their messages according to an ontology or create overly detailed arguments. Users can

then incrementally formalize the discussion, as will be discussed below, and editors can later distill. I-DIAG discussions will not be as complete as design rationale arguments, but we believe I-DIAG discussions are more likely to appear.

Finally, in our own earlier work, we examined collaborative systems for the distillation process. Answer Garden 2 (Ackerman and McDonald 1996) included the Collaborative Refinery (Co-Refinery), a system to support the refinement of messages and other raw information into frequently-asked questions (FAQs). Co-Refinery followed a process based on libraries' collection management processes (Gardner 1981, Osburn and Atkinson 1991). There were four steps. Collecting was the phase in which information is gathered into a collection, and culling was removing superfluous or redundant material from the collection. Organizing was the phase in which the materials were grouped according to some classification scheme (even an ad-hoc one). Some form of organization was a necessary precursor to distillation and to later retrievability. Distilling was the phase in which existing material was boiled down into shorter or more substantive materials. Each of these phases was considered a separate activity, and each was considered independently valuable. It was also assumed that any of these phases could be done iteratively or in any order. Co-Refinery supported organizing and distilling the materials. I-DIAG takes its beginning point from Co-Refinery and its mechanisms.

In summary, considerable work has been done on creating, fostering, and governing e-communities. Systems have also been created to foster and support brainstorming and decision rationale on-line. However, there has been little work, to date, on distilling informal information, especially group brainstorming results.

Architecture and Services

Differing users and their tasks suggested multiple applications, rather than trying to do everything in one Web-based application. For the discussion portion of I-DIAG, the interface requirements are relatively low. A Web-based interface could handle those requirements, and so we could consider customizing one of many Web-based discussion systems. On the other hand, there are substantial interface requirements for interactively handling sense-making, collaborative, and ad-hoc representations of complex intellectual spaces. As we found in the Co-Refinery (Ackerman and McDonald 1996), Web-based interfaces would likely be marginal.

Therefore, we constructed I-DIAG instead as an environment into which new applications and auxiliary agents can easily be added. The architecture allows a gradation between user-controlled applications and autonomous agents. The architecture is shown in Figure 1. As many Web-based applications have, I-DIAG has a database at its core. For I-DIAG, the database stores largely hypermedia objects as well as meta-data. Applications (discussed immediately

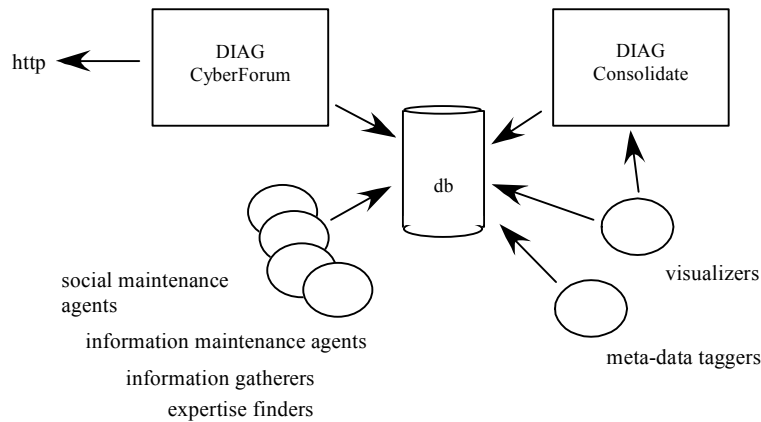


Figure 1: I-DIAG architecture

below) and agents feed to and from the database. As new services are developed, they can be placed into the architecture easily. We expect some of these services and applications to consist of relatively standard software projects; others will consist of research prototypes.

The following two subsections describe the two major applications that provide the basic functionality of I-DIAG.

CyberForum

The front-end, discussion service is called I-DIAG/CyberForum (CyberForum). CyberForum is a typical Web discussion site. This application is absolutely essential to solving the scenario problem described in the introduction, since all discussion occurs within it.

Figure 2 shows the CyberForum home page. The home page shows the most recent posts. On a normal topic page, it would show the messages for that topic. These messages are threaded, as is normal in similar systems, and are shown to a user-settable depth. Figure 3 shows part of a discussion. At the top of the main area is a summary; summaries „roll up“ part of a discussion. (Summaries will be discussed more fully below.) On either side of the page are small boxes that contain information, links, and program actions for the user. The type of boxes and actions are dependent on the user's level, and they can be customized by the user.

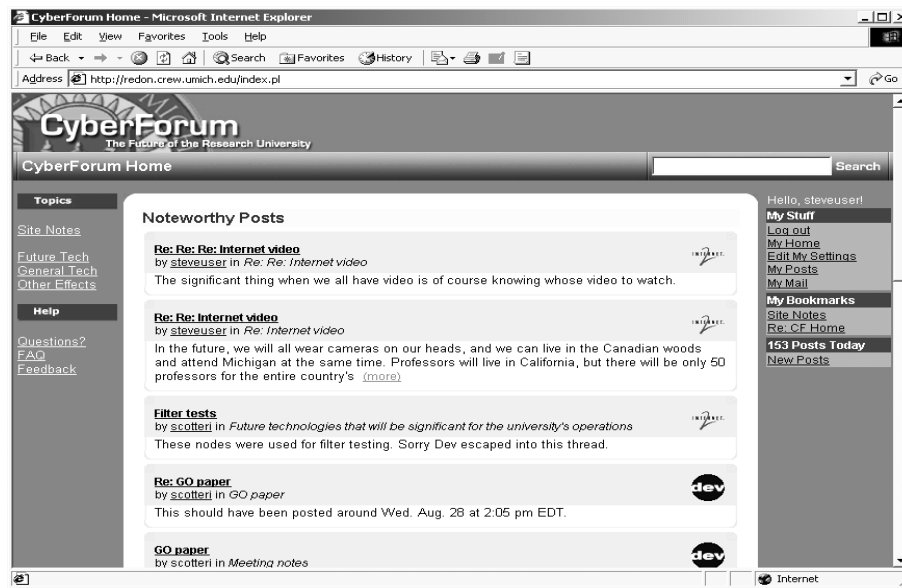


Figure 2: CyberForum home page

In addition to the basic CyberForum engineering, several research problems had to be addressed. As mentioned above, at a social level, we wished to consider collaborative mechanisms to facilitate social interaction and regulation. Because CyberForum is intended for relatively short-term use – a few weeks or a month for a particular site – the system has had to be optimized not only for performance efficiency (as does any Web application) but also for *social maintenance*. Social maintenance includes how to motivate users to come to and continue to participate at the site (*social facilitation*²) as well as how to deal with problem users (*social regulation*). We will discuss the mechanisms to support these requirements in detail below, but briefly, we added:

- Facilities to allow people to easy come in and out of discussions. In order for users to return to the site over time, it is important for them to be able to easily determine the current state of discussions as well as see what is new.
- User facilities to see what messages someone has posted. This not only provides a motivation for users to post, it also allows some pre-processing for later distillation. Moderators can highlight interesting posts for other users. Moreover, they can annotate, discuss, or merely note interesting posts for later examination.

² The social facilitation effect, in social psychology, occurs when people are motivated to act because others are acting similarly. Examples include going to well-populated restaurants or, negatively, not rescuing a crime victim when no one else does. However, we use the term in a more general sense to include all manners of facilitating long-term social interaction in a collectivity.



Figure 3: Summary in CyberForum

- Summaries to close problematic discussions. Summaries can provide a visual consolidation with further discussion allowed, a closing-off of further discussion, or a conclusion to an extended discussion.
- Agent-based mechanisms by which message traffic can be monitored for problem users, spam, robot posters, and the like.

These will be discussed further below. We expect to add additional services to support the social requirements as we use CyberForum in limited field tests. Recently, we have begun to make our rating mechanism more flexible, especially with regard to the visual indicators for a message's rating by other users.

CyberForum is an application based upon the open-source Everything2 engine (essentially the same as that used by the Slashdot site). The Everything2 engine provides CyberForum with the capability for message creation, editing, and storing. The Everything2 engine also provides support for constructing displays, linking, and threading. It should be noted, however, that the effort of constructing the CyberForum application was still substantial; the raw engine provides only the basic underlying services. CyberForum is currently 40,000 lines of code in addition to the Everything2 facilities. (Everything2 is substantially smaller.)

As well, to support our problem scenario, we had to add two major facilities to the Everything2 engine in order to create our computational architecture. In order to have external agents, we added a SOAP interface. Everything2 out of the box does not support communication with external programs. This facility gives us many additional capabilities. To facilitate the social processes around editing and

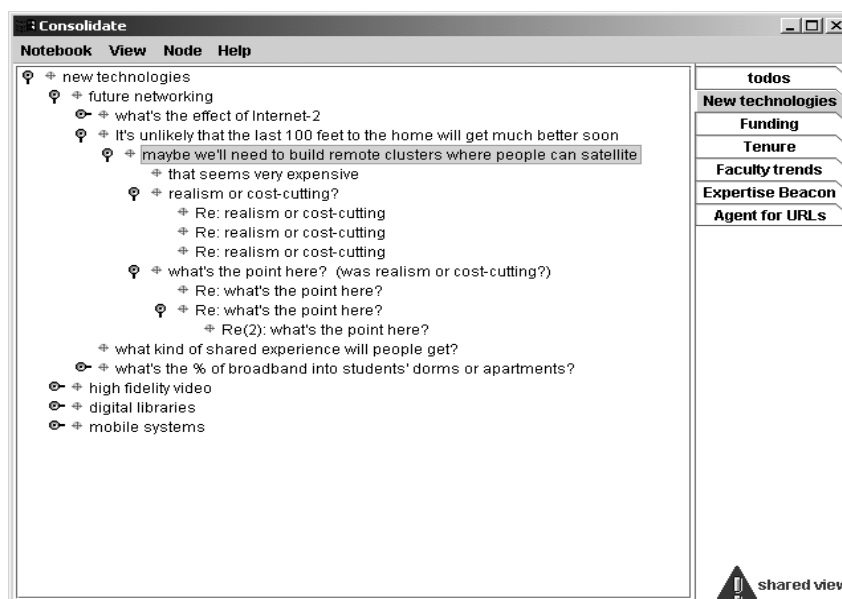


Figure 4: A Consolidate viewsheet

moderating of messages, we added a base layer of process support. This process facility is critical to our efforts at social regulation and maintenance.

Consolidate

The second major application in the I-DIAG environment is I-DIAG/Consolidate (Consolidate). Consolidate, in our scenario of use, will be used by experts to consolidate and distill the messages and organization of the site once people have finished with CyberForum. Consolidate consists of an extremely flexible core system that ties together extensible views, a query service, and visualizations of the information (in this case, messages, threads, topics, and people) and its structure

Consolidate provides for collaborative use through shared views. The data for these shared views can be handled through a variety of replication engines; currently a simple replication scheme is supported. Through the shared views, multiple editors can discuss and consolidate differing organizations of the raw information. Multiple messages, as well as additional information (e.g., editor notes, links to external references), can be consolidated into summary nodes. Figure 4 shows an outline view of a topic; the icon in the lower right corner (which is normally in red) signals that this is a view shared with others.

In Consolidate, editors can place messages into multiple topics or even rearrange the topics themselves. While Everything2 and hence CyberForum requires that all messages have only one parent, Consolidate does not. This is particularly important for knowledge distillation. Nodes can clearly be about

multiple topics. In addition, editors may wish to keep their own lists of interesting nodes, nodes by certain people, and other kinds of working lists.

In addition to views of the information, Consolidate contains a query service used to find new relationships. The query service currently allows users to retrieve based on topic, date, keywords, and author. We believe that a major use will be retrieval by author. Many times one finds an unusually perspicacious or even offbeat author, and wishes to find other postings by the same author. In the future, we plan a “reduced keyword” query based on latent semantic indexing. In this query, both the message space and the query are mapped to an approximately 100 dimensional space; this can improve retrieval, especially for short messages.

Consolidate also contains a number of semi-autonomous agents. Some will be used to crunch visualizations of the messages. Editors must search for outliers, either to eliminate them from a consolidated site or to make them prominent because they have novel or offbeat ideas.

Social Maintenance Services in I-DIAG

As mentioned, our concern in CyberForum is in finding new technical facilities for social maintenance and social regulation. E-communities to date have largely relied on social norms, reward structures, or other social structures to maintain and regulate themselves. We recognize the efficacy of these solutions; yet, efficiency and cost suggest examining potential technical augmentations. The cost of regulating problem users can be prohibitive. Accordingly, we are examining three technical mechanisms to help the people running a CyberForum. They range in level of augmentation. The facilities are summaries, used to control problem discussions; social maintenance agents, to watch for problem users and other social problems; and, governance objects, to radically alter the social structure of a CyberForum if required.

We cover each in turn.

Summaries

The first facility is summary nodes. As described briefly above, summary nodes „roll up“ a subdiscussion. As a summary, they can include straightforward text summaries created by hand or through software or both. Summary nodes serve to visually signal to the user that a block of related messages exist, but they need not be read since the summary is available instead. As such, they are largely visual indicators that a conclusion or partial conclusion has been reached. The internal text can link to specific messages if more detail is required or to serve as citations. Most importantly, summaries allow incremental formalization. Moderators or editors can slowly distill discussions while the discussion is ongoing.

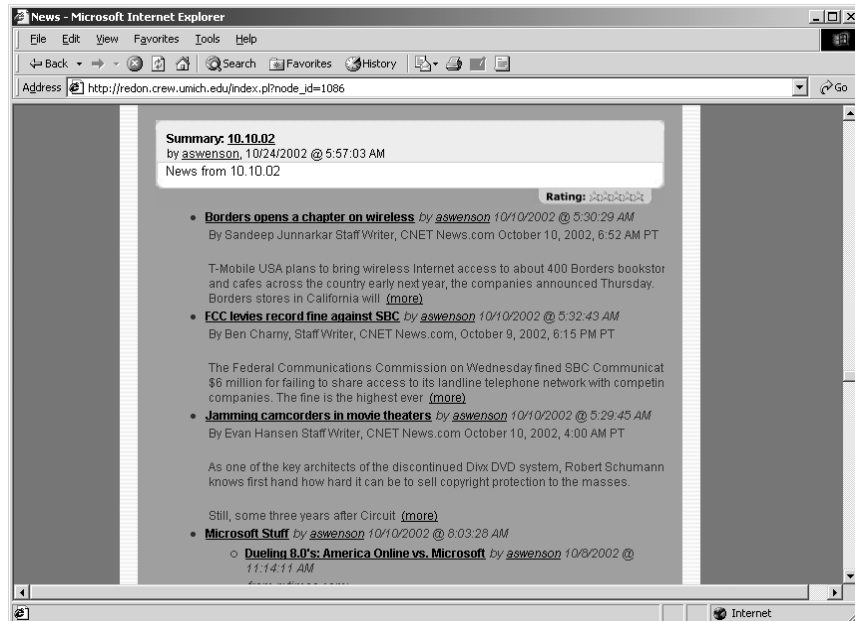


Figure 5: CyberForum summary

In addition, however, we created summary nodes to augment regulating further discussion. We needed to not only provide users with some mechanism to reduce visual overload; CyberForum also required some mechanism to forestall problematic conversations. Some debates are endless. Debates like „is the Macintosh better than the PC?“ can easily erupt in an e-community. (In our problem scenario, a similar debate might concern „what is the best fraternity?“). Summaries provide a mechanism by which a system administrator or a moderator can gently push users along by rolling up this argument, briefly summarizing it, and pointing out the intermittable nature of the discussion.

More importantly, while the endless debates are annoying, some debates can be socially destructive. CyberForum needed some mechanism to close off socially problematic arguments while not becoming overtly censorial. Some people post so-called „flame bait“, messages critical of minorities, women, lifestyles, or nationalities. Messages arguing, for example, that „some minorities get unfair breaks“ will serve their purpose, to draw attention to the writer. They can even create long-running arguments. However, these arguments are often organizationally dysfunctional. In the above example, a minority user will likely feel alienated, no matter how the debate ends. These arguments can create discord throughout the larger community, which is certainly against the purpose of I-DIAG. Accordingly, CyberForum summaries can be closed, disallowing further discussion. When summaries are closed, they serve as a statement by the system administrator or topic moderator that the discussion should be avoided. Figure 5 shows one of these closed summaries.

While simple, summaries are surprisingly useful socially.

Social Maintenance Agents

As mentioned, I-DIAG also includes sets of semi-autonomous agents. We have created a set of agents, and their support environment, to monitor social conditions inside CyberForum. Because of the programming environment of CyberForum, a result of its underlying Everything2 engine, it is necessary to run these agents outside of CyberForum. Nonetheless, we believe this type of facility is generalizable in that it would be useful in any e-community system.

Nodes (i.e., CyberForum messages) are written out as they are created. The nodes are sent asynchronously to avoid locking problems. The nodes are read in by the agent environment, parsed, and placed on an internal blackboard for further processing. (Blackboards are persistent tuple-stores, often used in agent architectures.) This blackboard can be read by any agent, and any agent can write partial results onto the blackboard for other agents. The environment can be periodically snapshot to storage for persistence.

Currently, we have implemented three agents. The first checks for flames by scanning a message for inappropriate words. This agent can also look for problem phrases. When a flame or problematic posting is detected, a message is sent to the appropriate editor or moderator in I-DIAG. Human intervention is necessary since the language may have been appropriate to a particular discussion.

The second agent watches posting rates. With unusually high volumes, the agent signals that a robot attack may be underway, and locks out the user. With moderately high rates over time, the agent signals the system administrators that this user should be acknowledged, since this user is one of the mainstays of I-DIAG. The agent also notes when a low usage-level user has returned to I-DIAG after a hiatus, currently set to two days.

The final prototype of a social maintenance agent notes when discussions are active or inactive. If a discussion is currently active, it sends a message to the appropriate moderator or system administrator. This person can then foster the discussion, adding comments for example. Alternatively, the agent can also note that a particular discussion has been inactive, especially if the system administrator or moderator has indicated that the discussion is not yet closed. Action can then be taken to draw the participants back to the discussion.

These agents only scratch the surface of what will be useful. We will uncover additional agents as we go through our field tests.

Governance Objects

E-communities or other virtual collectivities, like any other sociality, must have norm structures, membership structures, and other ways of governing themselves. At present, little support is provided through technical mechanisms for these social structures. Indeed, technical mechanisms that have been tried, such as floor control (Rein and Ellis 1991), have been too awkward to use effectively.

Elsewhere, we have argued (Ackerman 2000, Ackerman 2001) that there is an inherent gap between what we need to support socially and what we can facilitate technically: These social structures are inherently flexible, nuanced, and tolerant of exceptions, whereas our system mechanisms tend to be brittle, rigid, and standardized. Unfortunately, leaving everything to communication backchannels and people working out norms leads to substantial time loss and social breakdowns.

Governance objects (GOs) are an attempt to find a suitable work-around or approximation. We draw our inspiration from Hollan and Stornetta (1992). They argued that computer-mediated communication (CMC) would never be as good as face-to-face communication, and that comparisons of computer-mediated to face-to-face behavior would always be disappointing. They argued instead that the telephone, although inherently inferior to face-to-face communication, had an important characteristic that face-to-face did not. The telephone could be used for communication over long distances. Despite the telephone's inferiority, people not only tolerated but accepted it, because communication was good enough and because the communication went „beyond being there“.

Similarly, GOs are an attempt to use the advantages of computational mechanisms while acknowledging their limitation. Computational mechanisms to facilitate governance can never be as good as human forms of social interaction and social understanding. However, perhaps computational mechanisms could add something impossible in human interaction. Accordingly, the goal for GOs has been to allow people to quickly instantiate collective forms of governance. While users lose the flexibility and nuance of human interaction, they gain the flexibility of easily changing many rules simultaneously – something impossible with „normal“ social interaction.

GOs work essentially as templates, and users can quickly move among them.

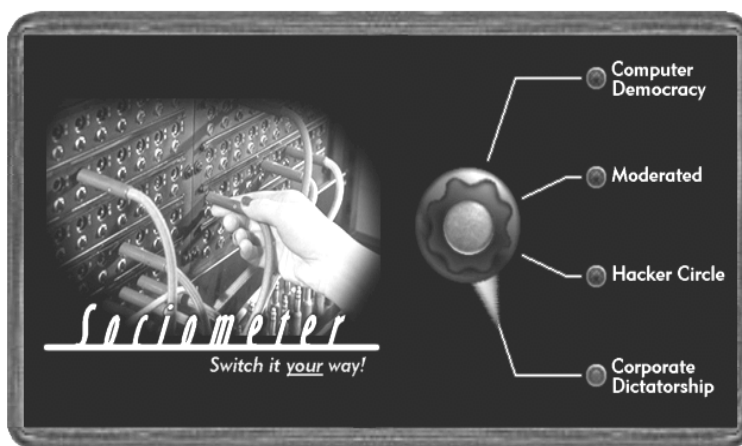


Figure 6: Governance objects as installed in an interface widget

A mockup that best describes our goal for GOs can be seen in Figure 6. In this interface, the user can quickly switch from „computer democracy“ (where anyone can post) to „moderated“ (where a moderator must vet postings first). Governance styles also control membership rules. In a „computer democracy“, anyone can join and post; in a „corporate dictatorship“, the manager can unilaterally decide to let users join or force them to leave; and, in a „hacker circle“, a vote allows people to join, but only in a probationary role. Of course, some GO transforms require additional input. For example, the name of a moderator must be provided for the covered topics.

We have implemented GOs twice in CyberForum. The first implementation assumed the priority of GOs. Each operation for each role in the system had governance-controlled access. The centrality of governance, however, came with a cost: Because of the code arrangements in Everything2, the display code was duplicated extensively. Our second implementation centralized the display code, but now the governance code is heavily duplicated. Since Everything2 is not object-oriented, there is essentially no easy way to conserve code – either governance or display will be duplicated extensively. Despite this, however, GOs (in reality, governance mechanisms rather than governance objects) appear to work. We can quickly change a number of governance rules at both the system level and at the topic level. This will be invaluable for large-scale implementations of I-DIAG, since it will be important to be able to switch among free-form and moderated discussions. In addition, we envision other important uses for GOs; for example, to provide the capability to have scientific-journal editor boards or other forms of oligarchical, expert regulation.

Implementation

Currently, as mentioned, CyberForum consists of over 40,000 lines of Perl code, over and above the base Everything2 engine and our extensions. CyberForum requires the Everything2 open source engine, a MySQL database, an Apache Web server, and Debian Linux. The social maintenance agents and the agent environment are written in Java. The code for each agent is rather minor; the largest is several hundred lines of Java code. The agent environment is approximately 3000 lines.

I-DIAG/Consolidate is constructed in Java and Jython, the Java implementation of the Python language. (Consolidate uses Jython as both an internal scripting language as well as a scripting language for user-created agents and user-modified views.) Consolidate currently consists of approximately 16,000 lines of Java code. Consolidate runs on any Java platform; it assumes the MySQL database or a connection to a Web server for the CyberForum nodes.

Only CyberForum is ready for full deployment. We are currently testing CyberForum in a limited field study consisting of two University of Michigan

classes. Our preliminary results appear encouraging. Students are using the system. (We have limited field data to date, because of privacy controls.) The classes have posed several new requirements. The most important requirement has been the capability to place static documents within discussions. This capability will be useful for our problem scenario as well; it is often useful to have joining a discussion first examine a common set of reference documents or links.

We are planning a larger scale field test in the near future using both CyberForum and Consolidate.

Summary

I-DIAG is an attempt to create a suitable environment for a common knowledge problem – to bring a large group of people or an organization together to discuss and brainstorm a problem, followed by experts distilling the results into something meaningful and succinct. Both steps require technical facilitation and augmentation.

To meet these goals, I-DIAG requires a suite of applications and services. This paper has described I-DIAG's two applications – CyberForum, as the discussion application, and Consolidate, as the distillation application.

More importantly, to meet these goals, I-DIAG also requires a set of technical mechanisms to facilitate social maintenance, social facilitation, and social regulation. We need to motivate people to come to the site and to continue to use the site. We also need to control problem users and situations. The paper has described three technical mechanisms to augment social maintenance. All provide some support for social facilitation and regulation, although the amount varies. Closed summaries provide a basic level of social regulation, closing off problematic discussions, debates, and so-called „flame fests“. Social maintenance agents search for problem users, as well as situations ripe for motivational reinforcement. Finally, governance objects (GOs) provide for quickly switching among sets of social rules, including social maintenance, membership, and authorship control.

Acknowledgements

This work has been funded, in part, by the National Science Foundation (IRI-9702904) and the University of Michigan/School of Information Alliance for Community Technology. Portions of I-DIAG Consolidate were originally funded by the MIT/Laboratory for Computer Science Project Oxygen Partnership. We would like to thank Dan Atkins for his extended help, as well as Cliff Lampe and George Furnas for their insights on these issues.

References

- Ackerman, Mark S. 2000. The Intellectual Challenge of CSCW: The Gap Between Social Requirements and Technical Feasibility. *Human-Computer Interaction*, 15 (2-3) : 179-204.
- Ackerman, Mark S. 2001. The Intellectual Challenge of CSCW: The Gap Between Social Requirements and Technical Feasibility. In *HCI in the New Millennium*. Edited by J. Carroll. New York: Addison-Wesley.
- Ackerman, Mark S., and David W. McDonald. 1996. Answer Garden 2: Merging Organizational Memory with Collective Help. *Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW'96)* : 97-105.
- Ackerman, Mark S., and Leysia Palen. 1996. The Zephyr Help Instance: Promoting Ongoing Activity in a CSCW System. *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'96)* : 268-275.
- Cherny, Lynn. 1995. The Mud Register: Conversational Modes of Action in a Text-Based Virtual Reality. Stanford University, Department of Linguistics, Ph.D. dissertation.
- Conklin, Jeff, and Michael L. Begeman. 1988. gIBIS: A Hypertext Tool for Exploratory Policy Discussion. *Proceedings of the CSCW '88* : 140-152.
- Dennis, Alan R., Jay E. Aronson, William G. Heninger, and Edward D. Walker. 1999. Structuring time and task in electronic brainstorming. *MIS Quarterly*, 23 (1) : 95-108.
- Gardner, Richard K. 1981. *Library Collections: Their Origin, Selection, and Development*. New York: McGraw-Hill.
- Hollan, Jim, and Scott Stornetta. 1992. Beyond Being There. In *Proceedings of ACM CHI'92 Conference on Human Factors in Computing Systems*. 119-125.
- Hurwitz, Roger, and John C. Mallery. 1995. The Open Meeting: A Web-Based System for Conferencing and Collaboration. *Proceedings of the Fourth International Conference on The World-Wide Web*
- Jonassen, David H., Katherine Beissner, and Michael Yacci. 1993. *Structural Knowledge: Techniques for Representing, Conveying, and Acquiring Structural Knowledge*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- MacLean, Allan, Richard Young, Victoria Bellotti, and Thomas Moran. 1990. Questions, Options, and Criteria: Elements of a Design Rationale for User Interfaces. EuroPARC/AMODEUS Report.
- Moran, Thomas P., and John M. Carroll. 1996. *Design Rationale: Concepts, Techniques, and Use*. Lawrence Erlbaum.
- Nunamaker, J. F., Alan R. Dennis, Joseph S. Valacich, Douglas Vogel, and Joey F. George. 1991. Electronic meeting systems. *Communications of the ACM*, 34 (7) : 40-61.

- Osburn, Charles B., and Ross Atkinson. 1991. *Collection Management: A New Treatise*. Greenwich, Connecticut: JAI Press.
- Preece, Jenny. 2000. *Online Communities*. New York: Wiley.
- Radev, Dragomir R., and Eduard Hovy. 1999. Intelligent text summarization. *AI Magazine*, 20 (3).
- Rein, Gail L., and Clarence A. Ellis. 1991. rIBIS: A Real-Time Group Hypertext System. *International Journal of Man-Machine Studies*, 34 (3) : 349-367.
- Shipman, Frank, and Cathy Marshall. 1999. Formality Considered Harmful: Experiences, Emerging Themes, and Directions on the Use of Formal Representations in Interactive Systems. *Computer Supported Cooperative Work*, 8 (4) : 333-352.
- Shipman, Frank, and Ray McCall. 1999. Supporting Incremental Formalization with the Hyper-Object Substrate. *ACM Transactions on Information System* *ACM Transactions on Information System*, 17 (2) : 199-227.
- Shipman, Frank M., III, and Raymond McCall. 1994. Supporting Knowledge-Base Evolution with Incremental Formalization. *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI'94)* : 285-291.
- Smith, Marc, J. J. Cadiz, and Byron Burkhalter. 2000. Conversation Trees and Threaded Chats. *Proceedings of the ACM Conference on Computer-Supported Cooperative Work (CSCW'2000)* : 97-105.
- Sproull, Lee, and Sara Kiesler. 1991. *Connections: New Ways of Working in the Networked Organization*. Cambridge, MA: MIT Press.
- Streitz, Norbert A., Jörg Geißler, Jörg M. Haake, and Jeroen Hol. 1994. DOLPHIN: integrated meeting support across local and remote desktop environments and LiveBoards. *Proceedings of the ACM Conference on Computer supported cooperative work (CSCW'94)* : 345-358.
- Turkle, Sherry. 1995. *Life on the Screen: Identity in the Age of the Internet*. New York: Simon & Schuster.
- Webster's. 1986. *Webster's Ninth New Collegiate Dictionary*. Springfield, MA: Merriam-Webster.
- Wenger, Etienne. 1998. *Communities of practice : learning, meaning, and identity*. New York: Cambridge University Press
- Xiong, Rebecca, and Judith Donath. 1999. PeopleGarden: Creating Data Portraits for Users. *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST'99)* : 37-44.