Available at
www.ElsevierComputerScience.com
POWERED BY SCIENCE DIRECT

Interacting with Computers 16 (2004) 579–610

Interacting with Computers

www.elsevier.com/locate/intcom

A framework for analyzing and understanding online communities

Clarisse Sieckenius de Souza[a,b], Jenny Preece[b,*]

[a]*Departamento de Informática, PUC-Rio Rua Marquês de São Vicente, 22522453-900 Rio de Janeiro, RJ, Brazil*
[b]*Information Systems Department, UMBC, 1000 Hilltop Circle, Baltimore, MD 21250, USA*

Received 28 May 2003; revised 13 December 2003; accepted 21 December 2003
Available online 5 March 2004

Abstract

Social interactions in online communities are varied and often complex, as are the communities themselves. The characteristics of the people, the range of purposes they pursue, the type of governance policies they develop, and the design of the software supporting a community, vary from community to community. These characteristics determine a community's sociability. Thus, the availability of powerful analytic tools to help designers understand existing technology-supported social activity online can broaden the spectrum of design knowledge and promote new insights for designing computer applications of this sort. In this paper, we present one such analytic tool—a theoretically-based online community framework (OCF). In order to demonstrate the efficacy of the framework we elaborate on its communication constituent using semiotic theory to help us. This constituent is particularly important in the OCF because it addresses computer-mediated communication between community members, and also communication from interactive software designers to users via the software they design. This latter kind of communication can shape the community's experience to a considerable extent, as our analysis shows. The paper ends with an agenda for future research.

Keywords: Framework; Online community; Semiotic engineering; Sociability; Usability

1. Introduction

Online communities can vary hugely in their social and technical structure. Intuitively everyone seems to understand the concept of 'online community' but so far there is no

* Corresponding author. Fax: +1-410-455-3795.
E-mail addresses: preece@umbc.edu (J. Preece), clarisse@inf.puc-rio.br (C.S. de Souza).

0953-5438/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.intcom.2003.12.006

agreed upon definition. This is due to the strong multidisciplinary interest that this topic inspires. Sociologists tend to focus on networks of social relations, ethnographers on the roles and activities of small groups of individuals, and technologists on the structure of the underlying software supporting the community. Our definition is broad. An online community is 'a group of people, who come together for a purpose online, and who are governed by norms and policies' (Preece, 2000). The benefits of this broad definition are: (1) it encourages a balanced view of both social and technical issues; and (2) it is widely applicable to a range of communities. For example, it applies to communities that exist only online as well as communities that also have physical presence.

Online communities are dynamic, evolving and constantly change. Understanding what makes such systems successful is therefore complicated. What is quite clear from the number of vacant community spaces on the Internet is that technology alone, even state-of-the-art technology, does not guarantee a successful online community. Success is determined by social factors (i.e. sociability) as well as software functionality and usability. In fact, in some communities sophisticated software design seems to have little impact (Maloney-Krichmar and Preece, 2003). However, well-designed software can make a successful community even more successful. Understanding the impacts of software design on the evolution of online communities is therefore an important part of building technology to support social activity online. The focus of our research is on how this particular kind of human experience is enabled and affected by technology. The purpose of this paper is to present a framework that can be used to leverage knowledge about this focal point. We demonstrate its reach by means of illustrative examples, and conclude by suggesting items for a future research agenda.

In Fig. 1 we represent the key components of an online community—people, purposes, policies and software—and the key qualitative factors that impact its success—sociability and usability (Preece, 2000). Together they form the basis for our online community framework (OCF). *Sociability* is concerned with *social interactions* in the online community while *usability* is concerned mostly with what happens at the human–computer interface. Usability is a more established and better-understood concept than sociability. By addressing sociability first designers are encouraged to focus on the social needs of users *before* deciding on the software design. Furthermore, online communities evolve and change, and thus many design decisions will need revisiting regularly.

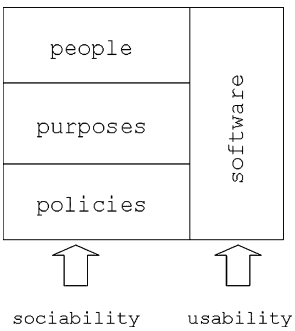

Fig. 1. Key components and factors on the basis of the online community framework.

For example, policies and software that support a new community may need to be changed as the community becomes established. Evaluating sociability and usability in online communities therefore requires a different approach from evaluating software for a single user application that, by comparison, becomes relatively fixed once it is shipped. Subtle changes to sociability (e.g. a moderator's style) or usability (e.g. the way a policy is described and presented at the interface) can have profound effects. The goal of the OCF is therefore to support evaluators, designers, moderators and users in identifying and understanding mainly *sociability* problems and related usability problems in online communities and other software that supports social interaction online. Because this approach focuses on sociability and social interaction it is different from the focus of most other methods, such as user-centered system design (Norman and Draper, 1986), which was proposed primarily to address design issues in single user systems. Another key difference is that our framework is based on principles of semiotic theory (Eco, 1976) and semiotic engineering (de Souza, 2004), whereas user-centered design is based on cognitive theories.

In the OCF the range of human activity is *structured* along three of our four key components, namely: people, purposes and policies. Each component concurs to forming the central online community constituent, which can be further decomposed to reveal additional detail, in the form of sub-constituents. To be *structured* according to these components means to be necessarily understood, communicated and achieved in terms of such central concepts. Therefore, all OCF-based conceptualizations, decisions, and product design to support human activity online will necessarily involve people, purposes and policies. Moreover, human activity is *shaped* by our fourth component, software (the collection of programs and systems that online community members can use), and *affected* by sociability and usability factors. In other words, technology shapes human experience, and a substantial portion of the overall quality of human experience in technology-enabled environments depends on how well the software used by community members matches the sociability and usability requirements of the whole community. Even the policies that are devised by community managers or the community themselves must be represented in software. They are sociability components manifest through software. Decisions such as where to position a policy, how to present it typographically, and what kinds of navigation or interaction events to associate with it are all usability decisions related to sociability factors. Even conversation content[1] has both sociability and usability dimensions to it. And because all such meaningful dimensions of online communities involve software *signs* of various kinds (e.g. web pages and navigation links, action-triggering buttons and menu choices, toolbars, text and graphics canvases and elements that can be used for self or other representation and communication), we chose to use Semiotics[2] as a foundational theoretic support for our study.

[1] For example, see Negretti (1999) for a discussion on how native and non-native speakers of English deal with restrictions on sense-making and meaning negotiation imposed by technology, especially through packaging actions, expressing paralinguistic meanings, conveying their identities and roles, and making lexical choices.

[2] Semiotics is the discipline devoted to investigating sign systems and how they are used in communication (Eco, 1976).

The value of working with the OCF compared to working with best practices and design guidelines is that, because the OCF is derived from an overall (semiotic) theory, it organizes the knowledge space into components, relations and functions that can *themselves* be inspected for consistency and soundness. Since best practices and design guidelines are usually derived from heuristic knowledge without further commitments with explaining why they are what they are, further questioning about the nature and mutual influence of practices and guidelines is difficult or altogether impossible without superimposing a theory on them. The OCF, by comparison, invites and provokes a deeper understanding of online communities and a broader conceptualization of issues.

Knowing how to support social interaction in online communities is a timely research issue. As Mark Ackerman pointed out, research in computer supported cooperative work (CSCW) has progressed well on the technical side but considerable effort is needed to understand social interaction with and via technology (Ackerman, 2000). Some important steps are being made in a new field called 'social computing' (Dourish, 2001), but greater attention to sociability is still needed. Furthermore, there are differences between CSCW, which as the name indicates is work-oriented, and online communities, which generally have less pronounced hierarchies and may not be work-oriented. These differences include:

- Many online communities exist mainly for social interaction as well as or rather than work.
- Online communities can involve large groups whereas most work groups are much smaller.
- Many online communities develop in an ad hoc way by a group of people coming together online for a particular purpose. This group then attracts others and the community evolves in an organic way rather than being planned in advance.
- Schedules and timeliness tend not to be a focal issue for most online communities as in most work-oriented communities.
- Participants in online communities are often widely distributed and may cross cultural and geographical divides. While multinational companies are on the increase, the members of many work groups are still geographically co-located, although this trend is changing.
- Many online communities exist on the Internet and are open to a wide variety of people whereas many CSCW communities are on an Intranet and membership is restricted.
- The skills and knowledge of members may be very broad in some online communities, which is often not the case in CSCW groups, which are more homogenous or deliberately selected to be complementary.

The OCF has the potential to enhance interpretation and decision-making processes in many CSCW systems but in this paper we focus on online communities. At an abstract and generic level, the OCF can support an analysis of breakdowns in existing online communities as a whole or in some of their specific constituents. It can also support a comparison of different communities, or a comparison of design alternatives when planning new online communities. To compare communities, for example, the framework can be usefully instantiated into OCF-based models (with a specification of known

entities, attributes, relations, functions and behavior) for each community of interest. Valuable insights might then be gained from comparing snapshots of: the same community at different stages of development; different communities at the same stage of development; or even the same community at the same stage of development, but using different technological support.

The semiotic foundation of the OCF follows the principles of semiotic engineering (de Souza, 1993, 2001, 2004). In this perspective, the perceived quality of the technology that supports online communities is not only a function of how well systems designers understand and interpret the users' *usability* and *sociability* requirements, but also and very importantly of how well they communicate this back to the users through their designs. This is why in the subsequent layers of OCF decomposition (see Section 3) we illustrate a particularly elaborate communication sub-constituent, for which we propose a systematic method of inspection, the Communicative Adequacy Test (CAT). We demonstrate how CAT can be used to reveal aspects of online community design that have not so far been addressed in such a systematic way.

In Section 2, we briefly discuss the main aspects of the semiotic theory that underpins our framework. Section 3 presents a detailed description of the framework. Section 4 illustrates the results of OCF-based analyses on messages from an existing online community. Finally, Section 5 discusses how the framework can be further developed and proposes a research agenda for future work.

2. Semiotic foundations

Semiotics is concerned with sign systems and communication (Eco, 1976). The fundamental semiotic entity is the *sign*, anything that can be taken as 'significantly substituting for something else', whether this *something else* exists or not, is true or false, known or unknown[3] (op. cit. p. 7). In the context of online communities, the reason for taking semiotics as a foundational theory is almost self-evident. Firstly, communication (through signs) is the means through which communities seek to achieve their purposes and manifest their policies. Secondly, the many varieties of signs that people can use to express themselves have different degrees of efficacy and effectiveness in different contexts and *media*. For example, gesture signs are particularly effective to convey emotions (e.g. the body language of someone who is nervous), whereas natural language is particularly effective to convey thoughts and reasoning. But lightweight computer technology affects the effectiveness of communication using body language, whereas heavyweight computer technology affects the reach of such communication (to only those who have access to this kind of technology). Substitutes for body language, like emoticons, have become an artificial visual *code* for some culturally selected messages conveyed in this language (e.g. happiness or unhappiness, pleasure or anger, etc.). This code is still unstable, as can be seen from the variety of emoticons that are offered by the various chat systems online. However, some of them are getting closer to becoming

[3] Eco's definition is a paraphrase of the most widely adopted among the original Peircean definitions for sign (Peirce, 1931–1958).

a stable conventional symbol that can be universally understood by the computer literate population.

Thirdly, communication is also the fundamental means by which software designers manifest the value and the logic of their products to users. The difference between this kind of communication and the more widely recognized kind of computer-mediated communication we just mentioned is that *direct* communication between designers and users (such as through online help or FAQ pages, for example) is secondary to the prevalent *indirect* form—a communication through the computer artifact itself, where the interface tells the designer's mind. This concept is the fulcrum of Semiotic Engineering (de Souza, 1993, 2001), a theory that views interactive software as a one-shot message from designers to users about how users must interact with the system in order to achieve a certain range of goals and experiences. This message is constructed taking into consideration all the designers have learned about who the users are, which preferences and abilities they have, which needs or ends they would expect to meet by using the software, and so on. It also takes into consideration the design choices (technical or other) and the kinds of signs that can be conveyed to and from users through the systems interface. These signs will have to *communicate* to users the design vision, and the particular code that the system is prepared to interpret and react to, its interface language. In the context of online communities, we can broadly paraphrase the designer's ideal[4] one-shot message to software users as this (de Souza, 2004):

> Here is my (i.e. the designer's) understanding of who you (i.e. the users) are, what I've learned you want or need to do, in which preferred ways, and why. And this is the system that I have therefore designed for you, and the way you can or should use it to fulfill a range of purposes that fall within this vision. You can communicate and interact with other users through the system. During communication, the system will help you check:

i. Who is speaking? To whom?
ii. What is the speaker saying? Using which code and medium? Are code and medium appropriate for the situation? Are there alternatives?
iii. Is(are) the listener(s) receiving the message? What if not?
iv. How can the listener(s) respond to the speaker?
v. Is there recourse if the speaker realizes the listener(s)misunderstood the message? What is it?In order to illustrate how the designer-to-user communication goes, let us examine how MSN Messenger® (MSN Messenger) achieves this. In Fig. 2 we see a snapshot of its interface. A superficial analysis of the signs in it helps us interpret the designer's message. The screen is composed of visual and textual signs. There is a clear *conversational* metaphor suggested for user–system interaction, marked by the presence of the first and second persons of discourse—I and You—in such signs as 'My status', 'I want to', or '[You] Click here to join the Customer Experience

[4] We say that this message is *ideal* because, as can be seen in many existing CMC applications, many of the questions designers *should* help the users to answer are not dealt with by designers. An interesting use of semiotic engineering is to analyze which aspects of the ideal message are taken care of by design, and how well.

Fig. 2. A snapshot of MSN Messenger®.

Improvement Program'. The user switches between being the first and the second person of discourse, since in all of the preceding signs it is the user who is the referent for the pronouns I or You. Furthermore, whereas all the textual signs are in principle known to speakers of English, the visual signs are not as highly codified into a system of signification. For instance, the plane and the car images on the vertical left-hand side tabs can be both taken to mean 'travel', but the specific differentiating component that establishes a different meaning in each case is not obvious (the plane is paraphrased in a tip as 'Travel with Expedia.com', whereas the car is paraphrased as 'See your local Traffic Forecast and Gas Prices'). The bell sign stands for 'alerts' in general.

By walking through some of the questions that the designer is telling the users that the system should allow them to answer, we see that:

- for the conversational expression 'My status'

I Who is speaking?
The user ('my' refers to 'Clarisse', who is the user)
To whom?

Actually, to everybody else whom 'I' have allowed to see me in MSN Messenger. These people are actually the ones listed under the 'online' and 'not online' headings (which we have omitted for the sake of privacy).

- but for the conversational expression 'I want to send a file or photo'

I Who is speaking?
The user ('I' refers to 'Clarisse', who is the user)
To whom?
Actually, to the system, not always to the person to whom 'I' want to send the file or photo. This action engages the user in different kinds of conversational patterns, depending on whether the user is actively exchanging messages with one or more people. If only one person is chatting with the user, the system automatically hands the role of the listener to the person with whom 'I' am talking to. But if 'I' am talking to more than one person, the system suddenly claims back the role of the listener and asks 'me' to tell it to whom I want to send the file or photo.

This example is interesting because the conversational rules that are so clearly marked as a leading sign for conveying the things that MSN Messenger can do and the way users should interact with it in order to do it are sometimes bypassed. The user cannot predict when the system will become inconspicuous and let the user experience 'direct' communication with other members of his community, and when it will not. It can also not ask for direct mediation of the system using the most effective conversational marker for this end, which is a combination of verbal inflection (for the 2nd person singular, like 'Call my friend Jenny') and vocative expressions (like 'System, since you have all my data, register me in the Customer Experience Improvement Program'). Likewise, the 'person' or 'entity' that is speaking to the user when he or she is the 2nd person of discourse (such as in 'Click here to join…' or 'Select the name of the person you want to send a file to…' which 'I' am told when 'I' want to send a file or photo) is not clear to 'me'. If 'I' respond to the 'Click here to join…' invitation, the follow-up conversation tells 'me' that 'I' am speaking to Microsoft Corporation, whereas if 'I' respond to the 'Select the name of the person…', the follow-up conversation tells 'me' that 'I' am talking to MSN Messenger. So, although 'I' am always Clarisse, sometimes I think I am speaking to my friend Jenny but I am actually speaking to MSN Messenger, and some other times I think I am speaking to MSN Messenger but I am actually speaking to Microsoft.

In short, the conversational 'signs' that are completely familiar to any capable speaker of English have been used by the designers of MSN Messenger in a slightly different way compared to how they are culturally encoded. The costs and benefits of this variation are precisely the type of analysis that semiotic engineering can help designers achieve. It will allow them to compare and reflect upon their reasons for such choices, by taking account of potential sign deconstructions that users will have to cope with in order to interact productively with the application.

Another illuminating example of technological signs, and how they are learned and used by a community of users is the meaning of the status 'online'. 'Online' means that certain channels of communication are enabled, but not that the person 'is there'. In fact, the person may be anywhere (having forgotten to sign out of her session, for example).

Likewise, 'Away' does not mean that the person is away—it simply means that she is listed as being away (no matter if she is or not), and may not respond to a call without looking rude to the caller. The meanings that such signs take on in computer-mediated communication are novel and surprising. They can trigger new rules of politeness and netiquette, which if unknown may get the users into trouble. And this is why designers should be encouraged to analyze their choices in further detail, so that they do not inadvertently contribute to a heightened degree of conflict or dissension in a community simply because the users do not understand the meanings (or side effects) of the signs they are using to communicate with each other.

In the context of our OCF, semiotic engineering is particularly useful if it can help designers understand *their* communicative challenges better. In other words, if the OCF can help them analyze online communities, identify the relevant dimensions for design, understand the factors that impact all the complex processes of communication that take place in online communities, then we can proceed with our research efforts towards even more powerful tools. The complexity of communicative processes also covers computer-mediated communication among community members and the designers' own communication to each and every user about how the software they have designed operates and can potentially enable them to have pleasurable and productive experiences online. In Section 3, we present the OCF in detail.

3. The online community framework

The purpose of the OCF is to help designers understand online communities, keeping in mind: the structure of their constitutive entities and the relationships among them; and the communicative aspects of computer-mediated human interaction that affect such communities. The ease with which community members interact with each other and with the technology will depend on how well designers support sociability and usability. Thus, by analyzing existing communities or candidate design solutions using the OCF, designers will be able to make their understanding about how their design is intended to meet community needs more explicit and potentially more perspicuous. Since design is a knowledge-intensive activity, any tool that improves the level of the designer's knowledge contributes to the outcome of the process.

The overall constituent structure of the OCF is shown in Fig. 3. Each one of the three constituents (in white boxes) will be described in more detail in the sub-sections that follow. The *online community constituent* is embedded in a software environment and thus achieves computer-mediated human experience. The *usability* and *sociability constituent* provides the goals (requirements and inspiration) for designing the online community. Since sociability is concerned with the social interactions in the community, whereas usability describes human–computer interaction, sociability and usability directly affect the online community. Finally, the *interpretive constituent* matches the usability and sociability design goals with how they affect the community's experiences, and explains how technology enables the achievement of computer-mediated experiences.

The *online community constituent* of the OCF comprises an idealized abstraction of an online community structured in terms of people, purposes and policies, which recursively

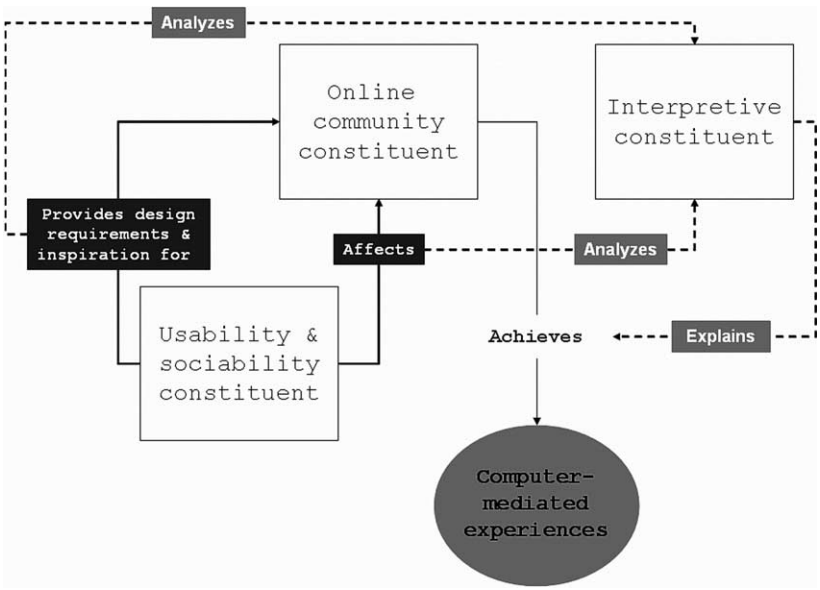

Fig. 3. The three basic constituents of the OCF and the relations among them.

unfolds into sub-constituents, functions, and semantic relations among them. It contains: a set of entities (e.g. people, individuals and actions); the main relations between them (e.g. share, constitute, influence); and attributes (e.g. name, role, goal) of both entities and relations. A graphic representation of the community structuring is shown in Fig. 4 (attributes of entities and relations are not shown).

The *interpretive constituent* enables designers to evaluate if and how computer processing affects human interaction. This evaluation refers not only to the communication between community members themselves, but also (following our semiotic engineering perspective) to the communication from designers to users. As seen in Fig. 5, the *interpretive constituent* is structured around a central interpretation process that receives input from the *usability and sociability constituent*, from culturally-determined sign systems, and from the types of communication going on to, from, and within an online community. An important part of the *interpretive constituent* is the CAT, which helps designers evaluate communication and formulate explanations about computer-mediated communication experiences. CAT will be explained in detail in a subsequent sub-section.

Finally, the *usability* and *sociability constituent* refers to the goals and needs of online communities. The usability factors that matter for online communities refer to both the individual and the community levels. At the individual level, the usability constituent examines infrastructure attributes—e.g. Media type, Network capacity, and Computer capacity, as well as Software attributes—e.g. Software navigation and Community information. At the community level, this constituent examines attributes such as Conviviality, and the Efficiency and Effectiveness of actions. The sociability factors also refer to individually perceived attributes and to community attributes. The sociability constituent examines such individual attributes as Topicality, Reciprocity, Empathy,

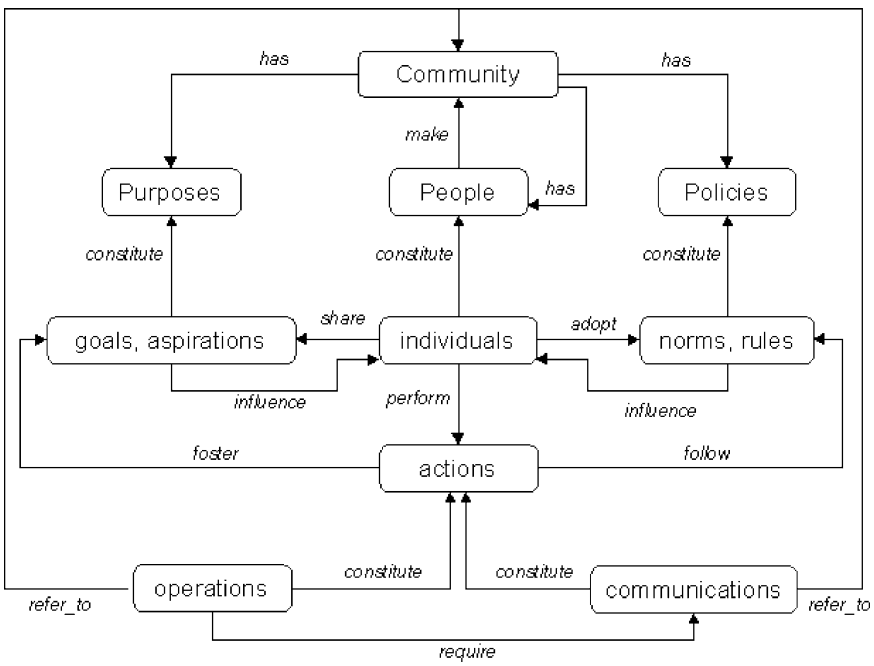

Fig. 4. The *community constituent* of the OCF (its four key ontological elements are capitalized).

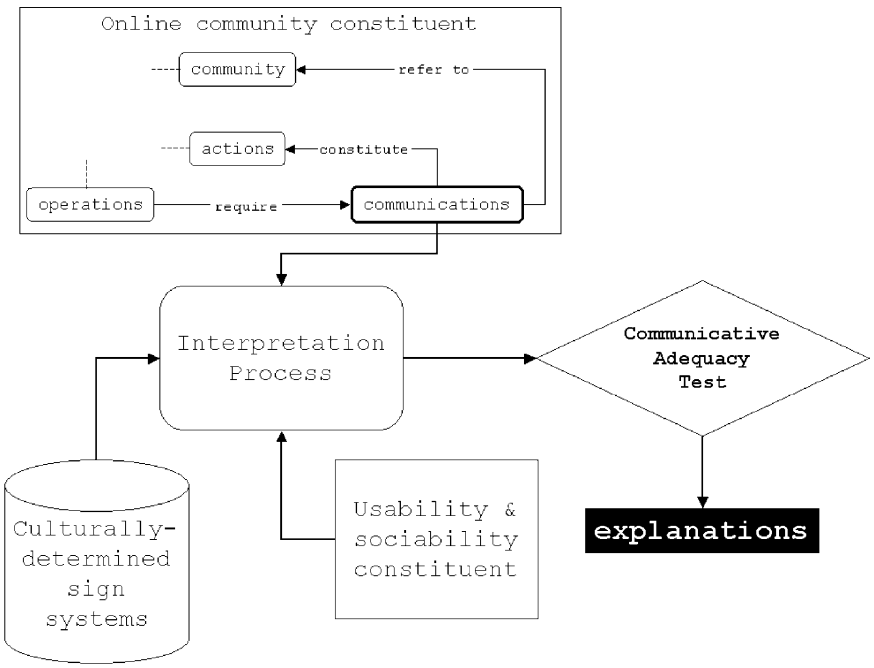

Fig. 5. The interpretive constituent of the OCF.

An original email message	Excerpt from the archived digest

Fig. 6. An unsigned message, whose sender is identified in the heading of the email on the left-hand side, becomes anonymous in the compiled archive copy.

Trust, Identifiability, Common Ground, Politeness and Privacy. The collective attributes examined by the sociability constituent are, for example, the Type of the Community, its Evolutionary Stage, its Size, and its Culture.

3.1. What an analytical framework can do

A designer who wants to use the OCF as an analytical tool can take the *online community constituent* (entirely or partially) and instantiate it to describe the community under study. For example, once instantiated, the *online community constituent* can reveal that the relation ⟨actions *follow* norms,rules⟩ does not hold in the community under study. Or, perhaps, that the relation ⟨communications *refer to* community⟩ does not hold either. To illustrate the analysis, let us suppose that in the community under study members have decided that no messages exchanged among them can be anonymous. However, on closer examination, the designer-analyst realizes that in the archival copies of past Bulletin Board digests the headings of posted messages are suppressed—only the body of messages is replicated and classified under broader themes of information (e.g. 'parties', 'help needed', 'incoming donations'). Thus, although in some cases the original messages may not have been anonymous (because the sender's identity was included in the heading), in the digests they may have *become* anonymous because of heading suppression. This happens to messages from people who do not usually sign their messages (see Fig. 6).

What we see in this example is that the technology is inadvertently undermining the norm. In the original email from Magnus Nielsen,[5] the message is not anonymous, and it has 10 lines of heading and only 5 lines of body. So, although the heading suppression techniques used to compile Bulletin Board archives were certainly justifiable for improving the usability of archival data, in this case technology itself introduced a violation of the community norms.

Regarding problems with the relation 'communications *refer to* community', the designer-analyst may realize that in another online community messages posted on an unmoderated Bulletin Board are not always about central matters that refer to the community. For instance, they may contain SPAM or announcements that are not related to the community's purposes, policies or people. The analysis will then indicate the need or opportunity to enhance technology so as to filter postings from outsiders, for instance, or to add more structure to the community, including moderators or norms with clearer statements about its purposes.

The purpose of OCF-based analyses is to have more insights about how technology *may* or *must* be used to improve usability and sociability, and prevent problems in computer-mediated communication and social interaction. Another fictional but

[5] All names and emails are fictitious in this paper.

X-Sender: troll@mail.this.community

X-Mailer: QUALCOMM Windows Eudora Version 5.1

Date: Mon, 23 Sep 2002 20:10:15 –0300

To: bboard@this.community

From: Magnus Nielsen <troll@this.community>

Subject: Impersonal messages

Cc:

X-Status:

X-Keywords:

X-UID: 6583

Hi,

Last week we had a newcomer asking us the old question about what she should do to find more about this, and that, and the other... All newcomers ask the same questions. And I thought the FAQ thing was a great step forward in helping us deal with having to say the same thing over and over again. But I felt that telling this person to "have a look at the FAQ session" was simply wrong. So, I wrote the whole thing again, and now I wonder what this FAQ is for... Let's not turn these newcomers off by telling them "to find it in the FAQ's", shall we?

Fig. 7. A sign of problems with technology.

plausible example of problems inadvertently introduced by technology concerns a software function for automatically generating frequently asked questions (FAQs) from Bulletin Board digests. In Fig. 7, an exemplary message warns community members against the misuse of FAQs.

3.2. The online community constituent

The online community constituent can be described at various levels of abstraction. The description that follows highlights key features of the entities and relations shown in Fig. 4. In order to use this constituent it is important to understand what the representations in it mean, and the interpretive inferences that can be drawn from them. The constituent is

represented as a directed graph in which:

- nodes (i.e. the rounded shapes) stand for *entities* (e.g. community, people, purposes, policies, etc.)
- arcs (i.e. the uni-directional lines linking nodes) stand for *binary relations* between entities (e.g. community has purposes, has policies etc.); and
- nodes and arcs have attributes (e.g. all entities and relations have a *name*, entities may be *types*, like for instance communications of type 'request').

Entities, the relations between them, and their attributes are subject to two types of rules:

- *structuring rules* generate particular sub-graphs (i.e. compound entities) under conditions determined by the entities, relations and/or attributes of the framework; and
- *inference rules* generate implicit relations between conceptual entities or place constraints on the values of the attributes (including establishing a unique value for an attribute) in the framework.

Entities may be simple or compound:

- *simple entities must* be explicitly represented as nodes in the graph;
- *compound* entities *may* be explicitly represented as nodes in the graph, but *must* all be derivable from the set of structuring or inference rules stated in the framework.

Relations may be direct or indirect:

- *direct relations must* be represented explicitly as arcs in the graph; and
- *indirect relations may* be explicitly represented as arcs in the graph, but *must* be derivable from the set of structuring or inference rules stated in the framework.

Structuring rules are represented in *if-then* format. A brief illustration of structuring rules is the one that derives a community from lower-level structures in the framework and the one that derives the structure of communication:

If the following relations hold:	*If* a structure of type community exists *then* a structure of type communication with attributes
Individuals *share* goals and aspirations	Identifier = ⟨identifier⟩
Individuals *adopt* norms and rules	Speaker = ⟨speaker⟩
Norms and rules *influence* individuals	Listener = ⟨listener⟩
Goals and aspirations *influence* individuals	Topic = ⟨topic⟩
Individuals *perform* actions	Content = ⟨message content⟩
Actions *follow* norms and rules	Form = ⟨message form⟩
Actions *foster* goals and aspirations	Appropriateness = ⟨agreement with politeness and cooperation principles⟩

then a structure of type community	Listener_understanding = ⟨expected
with attributes	linguistic ability of the listener⟩
identifier = ⟨identifier⟩,	Listener_response = ⟨expected action of the listener⟩
purposes = ⟨set of purposes⟩,	Pre_conditions = ⟨pre-conditions applicable to world and interlocutors⟩
policies = ⟨set of policies⟩,	Post_conditions = ⟨post-conditions applicable to world and interlocutors⟩ exists
people = ⟨group of individuals⟩ exists	

Inferencing rules are also represented in *if-then* format. They differ from structuring rules because they do not create new structures—they only generate new relations. A brief illustration of an inferencing rule is:

If a community exists
then
it *has* purposes;
it *has* policies;
and it *has* people; and
goals and aspirations *constitute* its purposes;
norms and rules *constitute* its policies;
individuals *constitute* its people; and
people *make* the community

Depending on the availability of knowledge from theory and practice different entities can be decomposed to reveal more detail, and their attributes can be set to reflect adherence to particular theories and approaches. Take, for example, the case of actions that can be performed by individuals:

- Individuals' actions *are constituted by* two main types: operations and communications.
  - Operations, as the name implies, are associated with achieving task-oriented goals. Communities of practice and CSCW teams, for example, tend to be work-oriented (Wenger, 1998). For these communities achieving the work goal is the prime driving force. The OCF may complement analysts who want to explore such operational aspects. Thus, for example, they can turn to McGrath's taxonomy of tasks, which defines four broad categories of operations that contribute to final goals: generate, choose, negotiate and execute (McGrath, 1984).[6]
  - Communication also *constitutes* action and *is required* by operations. Communication attributes can be established according to Bale's social psychology taxonomy that distinguishes between communication for informational purposes and communication for social–emotional purposes (Bales, 1958).

[6] McGrath includes communication but it contributes to achieving the goal and is not an end in itself.

Alternatively, communication can be investigated from a linguistics perspective, by resorting to Speech Act theory (Searle, 1979; Austin, 1962; Clark, 1996), as well as conversational principles of cooperation (Grice, 1975) and/or politeness (Leech, 1983).

All other relations in Fig. 4 may be enriched by associating specialized knowledge with the elements of the OCF online community constituent. For example:

- Communities *are made* of people, and individuals *constitute* the people in the community. These are obvious but important relations between OCF entities because they draw our attention to the existence of two perspectives: the collective and the individual perspective. These perspectives always exist and at times there may be tension between them. For example, Axelrod (1984) points out that social dilemmas can occur when what would benefit a particular individual is detrimental for the community. An individual may, for instance, try to benefit from group work without making a fair contribution to the effort. Another may contribute a small, inexpensive dish to a communal meal and then eat the expensive delicacies heartily. Online, some individuals may frequently request information but never answer queries from others. Online community developers have to strive to support both individuals and the community and protect each from exploitation. These associations are strengthened by other OCF relations, namely that both individuals and the community *have* goals and aspirations that *are fostered* by actions. These goals and aspirations *constitute* the purposes of the community. The community thrives when individuals' goals and aspirations are in line with those of the community.
- The community *has* people, purposes and policies. These three entities have been identified as important factors for community developers to consider (Preece, 2000). Others have taken a more process-oriented and integrated approach in describing their role in communities of practice (Wenger, 1998).
- Norms and rules *constitute* the policies of the community. They can be either informal or formal. Individuals *adopt* these norms and rules, which in turn have an *influence* on them. Different types of governance structures could be investigated depending on the interests of researchers in selecting the type of policies to be framed. For example, Lewin (Lewin and Lippett, 1938; Lewin, 1948) was influential in defining authoritarian, democratic and *laissez-faire* governance. Other political scientists have identified different categories including anarchic.

3.3. The interpretive constituent

The OCF's interpretive constituent reflects the principles of semiotic engineering overviewed in Section 2. Its central process—*interpretation*–uses a set of resources to explain the kinds of experiences enabled by technology. Following interpretation, the *Communicative Adequacy Test*—CAT—is performed, and its outcome is an evaluation of computer-mediated communication with respect to three types of communication: designers-to-users, users-to-users and users-to-system communication.

The external inputs to the central interpretation process come from communication part of the online community constituent and all elements of the usability and sociability constituent. Its internal resources are basically a set of sign systems, which amounts to all the different coding systems through which designers and users produce and interpret meanings. For example, natural language is a prime sign system in computer-mediated communication. Widgets commonly used to convey interactive opportunities like buttons, menus, hyperlinks, also constitute a sign system or code, although not as deeply established in culture, and nor as rich as natural language. Popular emoticons, like smileys, also constitute a code in which to convey meaning. Technically, widgets or visual signs, like emoticons, are a convention not as widely known and as competently mastered by users as other 'natural' codes (such as language, and culturally accepted behavior, values, and the like), but they play the same role as the latter in supporting computer-mediated communication.

CAT refers to the one-shot designer-to-users communication prescribed by semiotic engineering, and evaluates the signs exchanged through the computer system. Its scope extends from the system's ability to reproduce aural or typographic signs sent from one user to the other during communication, to its ability to provide useful error messages or instructions to help users with interface dialog. CAT is performed by an evaluator who inspects various types and levels of communication online and answers the following fixed set of questions:

I. do I *understand* this message? (e.g. yes, more or less, no)
II. can I detect whom the message is from? [this question unfolds into two further questions:]
    IIa. who *means* it? (e.g. system, user)
    IIb. who *has written* it? (e.g. system, user)
        III. is the system *interfering* with the communication ? (e.g. 5 = yes definitely, 4 = yes, 3 = maybe, 2 = no, 1 = definitely not)

Formally, the interpretive constituent of the OCF may be defined as a function that computes the communicative adequacy of CMC in the online community. It can be represented as:

$f(U\&S,SigSys,OnCom)$[7] $= CATresult$;
$I =$ *answer to question I*;
$IIa =$ *answer to question IIa*;
$IIb =$ *answer to question IIb*;
$III =$ *answer to question III*;
$CATresult = interpretation\_of(I \times IIa \times Iib \times III)$.

Within the scope of online communities, CAT can be used to differentiate various types of system and user behavior including behavior that is causing problems. In order to

[7] U&S = Usability and Sociability Component; SigSys = Set of Signification Systems; OnCom = Computer-Mediated Communication taking place in an Online Community; I × IIa × Iib × III = Cartesian Product of values I, IIa, IIb, and III.

illustrate how the interpretive constituent works, let us assume that the answer values for question I are {yes,no}, those for questions IIa,b are {user,system}, and those for question III are {yes,no}. CAT evaluation varies depending on three factors: (a) which values are used to answer I, IIa,b and III; (b) the usability and sociability principles or guidelines that are invoked by the evaluator; and (c) the sign systems that he or she adopts as a reference. Some of the basic semiotic principles and simple sociability guidelines that evaluators can use are:

1. Good computer-mediated communication in online communities is the result of verbal and non-verbal exchanges that can be understood and responded to by their intended receivers, no matter with what level of system intervention (e.g. the system may intervene to remind users of when to send a message or post a piece of news or the system may leave this decision completely in the hands of the users).
2. Problems arise if:
   - receivers cannot understand the message; or
   - if the system is interfering (negatively) with the communicative process (e.g. by preventing the users from responding appropriately to their understanding of the message).

In the case of a problem:

2.1 If the problematic message is intended and phrased by the system (e.g. if the system automatically detects the opportunity or need for a user to communicate with someone else and generates a verbal or non-verbal message to the targeted recipient on behalf of the sender—a situation that is more frequent in collaborative organizational environments, for instance), this is probably both a usability and a sociability problem;

2.2 If the problematic message is intended by a user but phrased by the system (e.g. via some automatic message generation only after a button is pressed by the user), this is probably a usability problem (that can be solved by rephrasing the message, for example);

2.3 If the problematic message is intended by the system but phrased by a user (e.g. when the system forces a user to send a message to an interlocutor—again a situation that often arises in organizational workflow applications, for instance), this is probably a sociability problem; and

2.4 If the problematic message is intended and phrased by the user, then designers can include some communicative resources online (such as word or phrase dictionaries, netiquette guidelines, and so on) that users can resort to if they realize they may be prone to unsuccessful communication.

In order to illustrate how CAT can be used, we will provide four scenarios.

Scenario I

Let us suppose that the User is busy talking to a colleague using an Instant Messaging (IM) tool, and that for the sake of comfort and practicality the IM window is maximized on her screen. She can then have a better view of previous parts of her conversation and also

a wider view of her input text box at the bottom of the screen. Let us further suppose that this IM tool automatically notifies a user when one of his or her *friends* (registered co-users of the tool) logs on. It does so by means of a pop-up window saying '*Your friend ⟨Name⟩ has just logged on. Would you like to say hello*?' with two buttons labeled *Yes* and *No*. So, while talking to her colleague, the User is bothered by messages from the system asking her whether she would like to socialize with her friends. This points to a usability problem. When 'the system' mediates human-to-human communication (for example, by suggesting that users take the opportunity to say hello to each other) the designer must ponder about the attribute 'appropriateness' for the element 'communication' in the online community constituent of the OCF (see Section 3.1). A polite person knows that asking if one is available for an interruption is in itself an interruption. So, before this person does it, he tries to interpret 'signs of availability' emitted by his intended interlocutor (e.g. she is not talking to somebody else). So, the system should be polite too, especially if it is acting on behalf of somebody else. This is why most Instant Messengers have less perspicuous ways of signaling the opportunity for socializing with friend online.

Scenario II

Now, let us suppose that the User receives the following message: 'ACB78@this. community wants permission to place an outside link to your page. [Y/N]' The user does not know who ACB78@this.community is, and is not sure what an outside link means. If he says 'yes', will he allow this person to navigate throughout the entire community space, or just on his particular page? Here again 'the system' is mediating human communication. The designer must ponder about the attributes 'speaker' (i.e. who is talking?), 'form' (i.e. how is the message phrased or presented?) and 'post-conditions' (i.e. what can or must be done in response to the message?) appearing in the online community constituent, in association with the communication entity. The 'speaker' of this communication is the system, but the 'form' of this communication is 'reported speech'. If the 'form' of this report is inappropriate (as it is for this particular user, who does not fully understand the words in the message), then the system's mediation is probably causing more trouble than direct contact would. Also, if the only two alternative responses the user can give are Yes (permit) or No (forbid), then the user must take action without knowing what he is doing. This is a usability problem, which can be easily corrected by giving access to help information and by elaborating more on the possibilities of dialogue (e.g. allowing users to say 'Yes', 'No', 'Tell me more', 'Let me talk to him/her', and so on).

Scenario III

CAT results can also inform design in more subtle ways. Suppose that a member of an online support community receives the following email message:

> Your disk quota has been exceeded. Please delete some of your files or contact the system administrator.

The user is likely to guess that the message was written by the system because of its terse phrasing. Alternatively, it might have also been written by the system's administrator—i.e. by a person—who assumes incorrectly that users understand the jargon associated with being a computer specialist. The important points are whether the user can understand what the message means and whether he feels that it is interfering

negatively with communication. If the user is having problems with the message, the message should be made clearer. It could be changed to:

> Hi,
> It looks like you have too many files stored in your system area, or that your files are too large. There is unfortunately no more space for you to store new files. You can either delete some of your files and make room for new ones, or contact us to see what we can do.
> Your system administrator

The new phrasing is not only more helpful for inexperienced users, but also more 'sociable'. For the system, it does not matter which message is issued it is treated the same way. For the designers, issuing the second message requires finer knowledge of how conversational protocols vary according to their social purposes. This in turn requires the designer to understand the user population and maybe to go to users and test out the alternatives. Inexperienced or insecure users may prefer the second message. Users who have worked with computers for a long time probably would not mind or might even prefer the first terse message. So, for the designer it is a question of knowing about the expertise of the user, the social context, and what the user wants.

In this example, CAT encourages designers to reflect on whether the meanings associated with 'system's administration' (as a process) should be more mechanical or more human in the context of a particular online community. It contributes to the social ambiance of the community. For example, the terse version of the message may be more in keeping with the ambiance of a technical support community whereas a longer, friendly, verbose explanation might be more appropriate for a patient support community. While usability specialists might claim that they have always tried to improve such messages to make them more understandable and acceptable to users we are now arguing that these messages have an obvious social impact. This is important because much of the software now in use, including that which supports online communities, facilitates social interaction between users in addition to individual user's human–computer interaction.

Scenario IV

In some situations, the system's role is only that of a carrier. If so, asking whether a message is comprehensible or whether it is having a negative impact on communication directs the designer to resources that the community would benefit from having online. If we examine the following Instant Messenger (IM) message, CAT is likely to return results classified as situation G or H in Table 2.

> Hi, Laura
> We are going to have a fund-raising performance next Sunday, and I was wondering if you would like to join us. Peter will not be able to come, and it would be nice to have you help us compensate for his absence.
> Cheers, Oscar

If Laura does not see herself as participating in any performance, and does not have a clue about who Peter is, she cannot understand the message, although the phrasing is OK

in English, in which she is fluent. One of the frequent causes for this kind of breakdown is that people sometimes have the same name, and others get confused with their email addresses. In this fictitious situation, this Laura (say Laura de Souza) is not the one the message was intended to (Laura da Silva). But Oscar thought laura@this.community was Laura da Silva. Laura de Souza is already used to getting messages meant for Laura da Silva, and she realizes that this is a problem with the technology. The system allowed her to register her email address as just 'laura', and now many people think she is the other Laura. This characterizes a 'usability problem' which designers can improve by adopting a certain pattern for login names (e.g. first initial + last name).

Scenario V

Now, suppose Laura is a Brazilian who has just joined this community and that the message is meant for her and correctly addressed to her, but she cannot fully understand what it says because her English is faulty and she does not understand what 'fund-raising performance' means. If designers know that their community is likely to gather members from different countries and cultures throughout the world, they would be well advised to consider incorporating some linguistic and pragmatic scaffolds for users. For example, Laura would benefit from having a handy link to a bilingual dictionary where she could be able to check what this expression means in Portuguese. Likewise, because her English is not too good, she is embarrassed that she must reply in *writing* (which automatically introduces some kind of formality in communication and allows the message to be archived with all the mistakes it may contain). She must use the right words and phrasing when replying to Oscar, who is a *senior* member of the community and a celebrity whom she looks up to. He signed just as 'Oscar', but she would like to show respect and cordiality in replying especially because she will have to tell him that she will not be able to make it on the date he proposes. The designers of this community could help users such as Laura by providing links to repositories of 'tips' about how to be polite, formal, or semi-formal, when talking to people online in English.

Supporting people like Laura with such linguistic *scaffolds* could improve the sociability of the community, and could help participants feel more confident when sending email, posting messages, or putting up homepages, for instance. This sort of analysis is backed up by CAT and can help designers finesse their product along usability and sociability scales. The gist of CAT is summarized in Tables 1 and 2. They highlight the categories corresponding to the five illustrative scenarios.

3.4. The usability and sociability constituent

The third constituent of the framework, shown in Fig. 3, summarizes the usability and sociability considerations that need to be taken into account when developing online communities. As was seen in Section 3.3, this constituent provides input for the CAT, which focuses on communication. However, it can be used to test the adequacy of any of the other entities in the online community constituent. In other words different usability and sociability considerations can be invoked during design depending on the focused entity or the usability theory chosen. The latter point is important because while usability is generally well understood, different experts emphasize different principles at different levels of abstraction. See for example, Shneiderman's eight golden rules (Shneiderman,

Table 1
Tabular representation of situations A–D in CAT

	Situation A (Scenario I)	Situation B (Scenario II)	Situation C	Situation D (Scenario III)
I. do I *understand* this message?	Yes	No	Yes	No
IIa. has it been *meant* by the system or by another user?	System	User	User	System
IIb. has it been *written* by the system or by another user?	System	System	System	System
III. is the system interfering with human communication?	{yes/no}	{yes/no}	{yes/no}	{yes/no}
Result (results are only illustrative, for the specific values of I, IIa,b and III, and for specific sets of Usability and Sociability principles and Sign Systems)	If III = Yes then check usability. A user receives an *understandable system message*. If it interferes with human communication then there may be usability and sociability problems	Check usability regardless of III. A user receives an *incomprehensible system message*, but supposedly it expresses another user's intent. This is very probably a usability problem	If III = Yes then check sociability. A user receives an *understandable system message*, but it expresses another user's intention. If there is a problem then check sociability aspects	Check usability regardless of III. A user receives an *incomprehensible system message*. This is very probably a usability problem

Table 2
Tabular representation of situations E–H in CAT

	Situation E	Situation F	Situation G (Scenario IV)	Situation H (Scenario V)
Do I *understand* this message?	Yes	No	No	Yes
Has it been *meant* by the system or by another user?	System	System	User	User
Has it been *written* by the system or by another user?	User	User	User	User
Is the system interfering with human communication?	{yes/no}	{yes/no}	{yes/no}	{yes/no}
Result (results are only illustrative, for the specific values of I, IIa,b and III, and for specific sets of Usability and Sociability principles and Sign Systems)	If III = Yes then check sociability. A user receives an understandable user message. But the system induced or forced the user send it, regardless of what the user really wanted to do. If it interferes with human communication then there may be a sociability problem	Check sociability regardless of III. A user receives an incomprehensible user message. But the system induced or forced the user send it, regardless of what the user really wanted to do. If it interferes with human communication then there may be a sociability problem. The sender may not see why she had to send the message to begin with	Check scaffolds regardless of III. If III = Yes then check usability. A user receives an incomprehensible message from another user. This is very probably a communicative problem of the user. But the evaluator senses that the system is interfering with communication. If it is, this is likely to be a usability problem	If III = Yes then check usability. A user receives an understandable message from another user. But the evaluator senses that the system is interfering with communication. If it is, this is likely to be a usability problem

1998), which are often reduced to 'consistent, predictable, controllable', and Nielsen's 10 heuristics (Nielsen and Mack, 1994) and the syntheses of these and other heuristics for various different purposes (e.g. Abras, 2003). Because of the more heuristic nature of its content, the usability and sociability constituent is neither presented as an entity–relation structure, nor as a function. Instead, it consists of a set of issues that designers must pay attention to.

Sociability is concerned with social interactions in the community. For example, should there be a registration policy, and if so who should be included and who should be excluded? What are the conditions that must be satisfied for membership? These questions need to be resolved before designers decide how to communicate the policy to community members and then design the software and usability. Should the policy be manifest at the human–computer interface? Should there be a form? If so, what should the interface be like? Perhaps a moderator should contact each person, though that would be very time-consuming. Perhaps each person should articulate her/his reasons for wanting to join. But how would this information be judged and what if some people challenged the fairness of this judging? Alternatively, perhaps a combination of a form and a description would be appropriate. These decisions bring sociability and usability together in design.

By addressing sociability first designers are encouraged to focus on the social needs of users before deciding on the software design. Many design decisions will need revisiting periodically. Policies and software that are appropriate for a community of 100 people, may be cumbersome and frustrating if the community increases to 1000 people, and unusable by 10,000 people. Some key issues that online community designers have to address include:

- keeping discussions on-topic (i.e. topicality—this involves determining whether the introduction of new topics is relevant or will distract the community from its agreed purposes);
- encouraging reciprocity (i.e. encouraging participants to give back to the community as well as to take from it so that social dilemmas (Smith and Kollock, 1999) are avoided);
- encouraging empathy and trust within the community (Preece, 1999)
- enabling people to develop their own online identities (Wenger, 1998)
- supporting shared understanding (i.e. common ground (Clark and Brennan, 1991; Clark, 1996)), and
- protecting their privacy.

As we already mentioned, these issues are relevant to both individuals and the community. The importance of each type of issue also varies from community to community according to:

- the community profile (e.g. the amount of lurking is different in technical support communities and medical support communities (Nonnecke and Preece, 2000));
- the community's stage of development (e.g. new, established, declining communities can differ considerably (Kim, 2000));
- the size of the community (i.e. communities of 100, 1000 and 10,000 typically have different needs (Preece, 2000)); and

- the culture that has developed in the community (e.g. some communities develop supportive cultures while others are argumentative, suspicious and hostile).

Usability is concerned mostly with what happens at the human–computer interface and it is a more established and better-understood concept than sociability. However, with the development of new technologies, including different kinds of online community software, traditional usability criteria and evaluation paradigms are frequently challenged (Preece et al., 2002). However, usability design can affect the community by impacting:

- conviviality (i.e. how community members communicate, react to each other and the kind of behavior that occurs. For example, people who mail to the whole group instead of targeted individuals can annoy others, even though it may not always be their fault. If the software had a better design, fewer people would inadvertently mail everyone);
- efficiency (i.e. how easily and quickly community members achieve operational tasks including communicating with others. Communities of practice are sensitive to efficiency. An online community must provide added value or it will not be used. If it is more efficient to get a job done off-line that's what people will do);
- effectiveness (i.e. how well community members perform activities. Does the software support people in working effectively?); and
- the satisfaction that individuals feel about belonging to a particular community (e.g. many factors contribute to satisfaction, which is related to how well a community fulfills its declared purpose).

An individual's experience of usability comes from interacting with the computer network infrastructure as well as with a particular online community's software. Infrastructure usability criteria include:

- media type (i.e. which media carry the message, the characteristics of the media, and whether alternative media can be used);
- network capacity (i.e. the slowest link in a network determines connection speed. Multimedia requires more network capacity than text, which may make receiving messages and information particularly difficult for some users depending on where they are situated on the network.); and
- capacity of an individual's computer and network connection to access the community.

Despite the advances in technology to broadband, providing versions for low bandwidth and older computers is essential for ensuring universal access (Preece, 2002), particularly in countries where people do not have access to high bandwidth computing facilities.

Finally, the usability of an online community's software is determined by the design of the community's:

- navigation structure (e.g. as for any website, being able to find what one wants and navigate through the community quickly and easily is essential);

- information presentation (e.g. members provide their own content but there are templates and advice on how to complete them. Care is also needed to ensure that terminology is used consistently and that information is positioned consistently); and
- required features (e.g. software with different features is needed to support different types of communities including search engines, the ability to send private messages, and communication support tools such as dictionaries and ways to signal emotion etc.).

4. Applying the framework

In this section, we demonstrate how our framework can be applied to analyze and evaluate an online education support community. The focus of our analysis is on communication, which extensively determines the sociability of an online community and, in view of the semiotic engineering perspective on HCI, where designers communicate to users, also determines usability in many respects. We start by providing a brief description of the community. Then we instantiate the parts of the framework into a model, in order to account for actual data from some example messages sent by the community we analyze. Each message gives rise to a part of the model that can be evaluated using CAT. Next, we refer to the list of sociability and usability principles discussed in Section 3.3 to suggest ways of improving communication within the community.

The Language, Literacy and Culture (LLC) education support community, is a new online community that has existed for approximately 1 year. It was developed so that students who come to study a new PhD program in LLC can share resources, keep in contact with each other and with the faculty who teach them and discuss issues of importance to them.

The two example messages that follow are from users and they point to different degrees of misunderstanding between the user and the system designer (in semiotic engineering terms—the system is 'the designer's deputy').

Message 1

A new user types: 'What's this site for?'

For this particular message the variable values for the *communication* entity in the instantiated OCF-based model are:

This is an entity of the type communication *with attributes*
Identifier = Message 1
Speaker = a new user
Listener = all the members of the community
Topic = the purpose of the community
Content = request for clarification about this community's purpose
Form = English sentence 'What's this site for?'
Speaker_intent = get a community member to respond to this request
Appropriateness = fully appropriate
Listener_understanding = ⟨*expected* to be understood by speakers of English⟩
Listener_response = ⟨*expected* to be a statement of the community's purposes or an indication of where this statement can be found⟩

Pre_conditions = user has listeners; user and listeners are part of the same context; users and listeners share the same language (or code of communication) in which the user's message is encoded; there is a channel of communication open between user and listeners; and the user's message is recognizably flowing through this channel.
Post_conditions = ⟨expected to be the fulfillment of user's intent⟩

The relationships in Fig. 4 (in Section 3) predict that all communication in an online community should *refer to* the community itself. Furthermore, the community has purposes constituted by goals and aspirations that should be *shared* by individuals. What the analyst realizes with the specifications of the attributes for the communication identified as *Message 1* is that this user may experience serious difficulties with respect to this community given that, since she does not know what the community's purposes are, she cannot be sure to share the goals and aspirations that constitute these purposes. Therefore, all the other relationships that emanate from the community and affect her (like, for instance, the adoption of norms and rules, or the performance of actions that ultimately foster the community's purposes) are jeopardized.

If we move on to the interpretive constituent, CAT reveals further aspects of the same problem. The user is stating (in Message 1) that she is not getting a message from/about this community (Community Message). By walking through CAT inspection we can see that Community Message was *not understood* by the user (value of I = 'No'). Proceeding to IIa and IIb we realize that Community Message is meant and phrased by users (so, we could check scaffolds for the language and/or style used to communicate the message content, and so on). However, when we come to III we see that the 'system' is interfering with this communication because the statement of purpose does not appear on the home page, and so it is easily missed. In order to find it, the user has to select the 'site map' from the home page and then must scroll down to the bottom of that page and click on statement of purpose. The statement of purpose is buried three levels deep. Also, new users often do not know that they must scroll down. So, even if they get to the site map they miss the statement of purpose. This has to do with the system's 'presentation' of this message (that is, with its 'phrasing'). Thus, if we follow the CAT inspection procedure, we conclude that there probably is a usability problem in this community. The solution to the problem, may come from usability guidelines available in the Usability and Sociability constituent.

A noteworthy aspect of this example is that the decision about where to put information in the community's website may have been made by members of the community, acting as designers of the system. We should not, however, be led into thinking that the non-member designers (i.e. the developers of the technological environment, or base system, on top of which the LLC community was built) could not or should not have made design decisions that could minimize or prevent the design problems introduced by the member designer. As we point out, the original non-member designers of the base system (like *YahooGroups*, for example), may anticipate this sort of problem and provide online design guidelines for member designers, so that some basic usability problems like this one can be avoided.

Message 2

A user says: 'Using pictures as logins is fun but I wouldn't want to send my gorilla picture to a professor because I'd be afraid that he wouldn't treat me seriously or that he'd

think I was being rude. The pictures are ok to send to friends but not to professors or to people you don't know.'

There are two interesting things here. Although the representation being mentioned is an image to both the user and the system, the meaning and contexts of use of the image are not the same for them. It is clear that the person would like an easier way to switch graphics off since they are not appropriate to send to professors. So, the lack of choice and flexibility is a design problem. The instantiated part of an OCF-based model of this communication is:

This is an entity of the type communication *with attributes*
Identifier = Message 2
Speaker = a user
Listener = all the members of the community
Topic = the alternative representations of participants in the system
Content = the use of pictures is not always appropriate as an identification of participants
Form = English sentences = 'Using pictures as logins is fun but I wouldn't want to send my gorilla picture to a professor because I'd be afraid that he wouldn't treat me seriously or that he'd think I was being rude. The pictures are ok to send to friends but not to professors or to people you don't know'
Speaker_intent = get opinions from other community members, and maybe make a point to the moderator or community manager
Appropriateness = fully appropriate
Listener_understanding = ⟨*expected* to be understood by speakers of English⟩
Listener_response = ⟨*expected* to be a statement of the views of others in the community, and maybe a change in the way the system works⟩
Pre_conditions = user has listeners (in particular a moderator or manager who can have some influence on changing the system); user and listeners are part of the same context; users and listeners share the same language (or code of communication) in which the user's message is encoded; there is a channel of communication open between user and listeners; and the user's message is recognizably flowing through this channel.
Post_conditions = ⟨expected to be the fulfillment of user's intent⟩

Because the system virtually does not allow for customization and contextual selection of user representations, the image associated with each user is the one used in all contexts of communication supported by the system. So when the user specifies a picture to accompany her login she is then stuck with that picture for everyone with whom she communicates. The designer did not take into account that the user needs different pictures; one when she communicates with her class peers; another when she communicates with her professor; and so on. And this flexibility should have been enabled by the base system (on top of which the community's system is built). Because it was not, one of the possible intentions of this user (that the system be changed by some intervention of the moderator or manager) may simply be impossible to be met by any

member of the community (including moderators and managers). So, the *action fosters goals and aspirations* of this community is a failing relationship in the model.

Although the CAT result for this user's communication per se fits in situation H (see Table 2) because the system does not allow for the community to respond fully to this user's implicit request, we can also use CAT to analyze the problem referred to by this user in her communication. The latter fits in situation A in Table 1. The system (not the user) is interfering with communication by using a sign that is neither *meant* nor *phrased* by the user in the context of her communication with professors. So, we have a usability problem causing a sociability problem.

5. Discussion and conclusions

This paper started by pointing out that there is a lack of consensus about the definition of an online community. It then proceeded to present a framework for modeling generic online communities. The framework is intended as a tool for analyzing both communication between the computer software and its users, and computer-mediated communication among users themselves, one with which potential breakdowns can be identified, explained and prevented or circumvented. It also provides a more formal definition of community, in terms of primitive entities, relations and attributes. Since the construction of the framework depended upon identifying, interpreting and generalizing these attributes using semiotic theory, it is befitting that the outcome should also contribute to an improved definition of our object of investigation.

Compared to our initial definition of an online community, an OCF-based model can help us discriminate and refine, in a very systematic way, all the definitional aspects predicted by the model. As an exercise, we have built a Prolog program to elicit the model of the online community constituent of the OCF. The program verifies all the entities, relations and attributes in the model and leads the analyst (modeler or designer) into thinking about some dimensions that might go unnoticed in an informal definition of an online community (be it for design or evaluation purposes). The program was extended to handle parts of the interpretive constituent, but not to handle the sociability and usability constituent. However, there are no predictable obstacles to this extension, which means that in principle we can have one of the most attractive by-products of formalizing models of online communities—the ability to build computational tools to support the construction or the evaluation of modeled objects. In other words, one of the major benefits of the OCF is to set the basis for the construction of computer-supported design, construction or evaluation of online community technology.

An important aspect of this framework is that it is a tool for identifying and analyzing usability problems as we saw in the analysis of the second message in Section 4, where the designer failed to recognize the full range of the user's needs (system to user, user to system, user to user). In addition it is valuable for identifying breakdowns between users communicating via the system. As with other HCI models, including GOMS (Card et al., 1983), it is not intended that OCF-based models should be applied to a whole online community. Rather, it is intended that it can be applied to particular aspects as in the examples discussed earlier. There may be times when it is more difficult to identify

the problems that a user encounters; for example, perhaps he misinterprets the meaning of an icon, or cannot find the icon or menu to take him to the next step of a process. Our framework, when combined with think aloud and semiotic engineering approaches, can be used by designers to identify misleading or missing design features.

Frameworks, such as the OCF can be useful for analyzing differences between systems and for gaining a better understanding about the constituents within a particular system. They are more closely related to an *ontology* of the domain to which they refer (i.e. to the general concepts, dimensions and relationships in it) than to a complete *model* of it that accounts not only for general concepts, dimensions and relationships of the domain, but also for the range of values that these may or may not take, as well as the behaviors or implications that result in such values. The OCF is intended primarily as a high-level analytical tool that can guide the inspection of existing or prospective (i.e. in the process of being developed) online communities. If needed, the outcome of this inspection can be refined into an actual model of the online community, in order to deal with specific lower-level design issues. The value of guiding inspection lies in calling the technology designers' attention to particular dimensions of online communities that are systemically related to each other and to the success of the community.

While interest in online communities continues to grow and practitioners want to know what distinguishes successful communities in which there is a constant flow of messages from those with little or no activity, little attention is given to evaluating online communities. Even successful 'how-to' books do not list evaluation or testing in their indexes (Kim, 2000; Powazek, 2002); other researchers have attempted to adapt the general evaluation approaches used in human–computer interaction (Preece, 2000). Nor is the issue debated much in the literature. While some researchers in CSCW acknowledge that evaluating collaborative, distributed systems requires different approaches from those developed to evaluate single user systems, online community researchers and developers have been more concerned with trying to understand the dynamics of online communities. A few attempts have been made to identify heuristics for evaluating usability and sociability (Preece, 2001) and these are starting to be empirically tested and validated (Abras, 2003). The OCF would complement heuristic evaluation well by providing a clearer description of how the relationships within the community affect each other and the community itself. The OFC will also be particularly useful for identifying and analyzing finer-grained problems such as those illustrated in the messages above. Moreover, unlike other evaluation techniques, OCF is a tool for analyzing user-to-user communication as well as interactions across the interface. Semiotic engineering underpinnings allow us to make an important distinction between usability and sociability, on the one hand, and, on the other, how usability and sociability factors are made explicit and available to users, by designers, via the system interface. In particular, as the examples in this paper have shown, the dependencies between communication, usability and sociability can be made much clearer when analyses are based on an OCF model. This approach can provide important guidance for online communities developers when they are trying to prevent or to solve social computing problems.

In addition to using the framework in the ways that we have just described there is scope for it to be further developed. The suggestions that follow could provide an agenda for researchers or they could be adapted by practitioners. For example, the OCF can be

instantiated to examine different theories about online communities. In the version described in this paper we used three main structuring components for sociability—people, purposes, and policies—and built the relationships within that framework. However, it could be used to model and test theories drawn from Sociology, Psychology, and more specifically, Decision Theory, Conversation Theory, or political theories. Depending on the theory chosen, a different set of definitional attributes, entities and relations could be modeled with the OCF. Another approach would be to use these theories to introduce alternative structures and/or entities and relations. Then the OCF would provide a comparison between the designs that would result from these different theoretical underpinnings. Yet another use would be to apply the OCF to different types of communities within a particular generic type, such as a few health support communities, to try to discern the characteristics of each that lead to its success or failure.

Other topics that might engage researchers include using the OCF: to analyze numerous existing online communities, successful and unsuccessful, and then evaluate its performance as an analytic tool; for *redesigning* a few cases of online community technology in order to compare their usability and sociability before and after the redesign; to explore tighter dependencies between CAT results and usability and sociability problems detected by other methods of analysis; to hypothesize semiotic engineering rules or specialized models that can inform the design online community technology; and to refine or develop alternative definitions of online communities.

There is considerable scope for elaborating upon the OCF and some researchers may wish to make the process easier by developing OCF-based computer tools that can support and speed up the exploration of some of these lines of research.

Acknowledgements

We wish to thank Ben Shneiderman for his comments on drafts of this paper. Chadia Abras and Diane Maloney-Krichmar have been excellent collaborators and have provided thoughtful comments on many of the ideas discussed in the paper. Clarisse Sieckenius de Souza would also like to thank CNPq for a research grant that allowed her to work at the Information Systems Department (IFSM) at UMBC during a short sabbatical leave from PUC-Rio. She also thanks IFSM for having provided her with a warm and intellectually stimulating environment to proceed with her research.

References

Abras, C., 2003. Determining success in online education and health support communities: deriving usability and sociability heuristics. PhD Thesis, University of Maryland Baltimore County.

Ackerman, M., 2000. The intellectual challenge of CSCW: the gap between social requirements and technical feasibility. Human–Computer Interaction 15, 179–203.

Austin, J.L., 1962. How to do Things with Words. Clarenton Press, Oxford.

Axelrod, R., 1984. The Evolution of Cooperation. Basic Books, New York, NY.

Bales, R.F., 1958. Task roles and social roles in problem-solving groups. In: Maccoby, E.E., Newcomb, T.M., Hartley, E.L. (Eds.), Readings in Social Psychology, Holt, Rinehart & Winston, New York, pp. 437–459.

Card, S., Moran, T., Newell, A., 1983. The Psychology of Human–Computer Interaction. Lawrence Earlbaum Associates, Hillsdale, NJ.

Clark, H.H., 1996. Using Language. Cambridge University Press, Cambridge.

Clark, H.H., Brennan, S.E., 1991. Grounding the communication. In: Resnick, L., Levine, J.M., Teasley, S.D. (Eds.), Perspectives on Socially Shared Cognition, APA, Washington, DC, pp. 127–149.

de Souza, C.S., 1993. The Semiotic Engineering of User Interface Languages. International Journal of Man–Machine Studies 39, 753–773.

de Souza, C.S., 2001. Editorial of the Special Issue on Semiotic Approaches to User Interface Design. Knowledge Based Systems 14 (8), 415–418.

de Souza, C.S., 2004. The Semiotic Engineering of Human–Computer Interaction. The MIT Press, Cambridge, MA, in press.

Dourish, P., 2001. Where the Action is. MIT Press, Cambridge, MA.

Eco, U., 1976. A Theory of Semiotics. Indiana University Press, Bloomington, IN.

Grice, H.P., 1975. Logic and conversation. In: Cole P., Morgan J.L. (Eds.), Syntax and Semantics. Vol. 3 Speech Acts, Academic Press, London, pp. 41–58.

Kim, A.J., 2000. Community Building on the Web. Peachpit Press, Berkeley, CA.

Leech, G., 1983. The Principles of Pragmatics. Longman, London.

Lewin, K., 1948. In: Lewin, G.W., (Ed.), Resolving Social Conflicts; Selected Papers on Group Dynamics, Harper & Row, New York.

Lewin, K., Lippett, R., 1938. An experimental approach to the study of autocracy and democracy. A preliminary note. Sociometry 1, 292–300.

Maloney-Krichmar, D., 2003. The Impact of an online community on its members: Group dynamics, usability and sociability, Unpublished Doctoral Dissertation, University of Maryland Baltimore County, Baltimore, MD.

McGrath, J.E., 1984. Groups: Interaction and Performance. Morgan Kaufmann Publishers, Inc., Englewood Cliffs, NJ.

Negretti, R., 1999. Web-based activities and SLA: a conversation analysis research approach. Language Learning and Technology 3 (1), 75–87.

Nielsen, J., Mack, R.L., 1994. Usability Inspection Methods. Wiley, New York.

Nonnecke, B., Preece, J., 2000. Lurker Demographics: Counting the Silent. Proceedings of ACM CHI'2000, Hague, The Netherlands, pp. 73–80.

Norman, D.A., Draper, S.W. (Eds.), 1986. User-Centered Systems Design, Lawrence Earlbaum Associates, Hillsdale, NJ.

Powazek, D.M., 2002. Design for Community. New Riders, Indianapolis, IN.

Preece, J., 1999. Empathic communities: balancing emotional and factual communication. Interacting with Computers 12, 63–67.

Preece, J., 2000. Online Communities: Designing Usability, Supporting Sociability. Wiley, Chichester.

Preece, J., 2001. Sociability and usability: twenty years of chatting online. Behavior and Information Technology Journal 20 (5), 347–356.

Preece, J., 2002. Supporting community and building social capital. Communications of the ACM 45 (4), 37–39.

Preece, J., Rogers, Y., Sharp, H., 2002. Interaction Design: Beyond Human–Computer Interaction. Wiley, New York, NY.

Searle, J.R., 1979. Expression and Meaning: Studies in the Theory of Speech Acts. Cambridge University Press, New York, NY.

Shneiderman, B., 1998. Designing the User Interface, third ed., Addison-Wesley Longman, Reading, MA.

Smith, M.A., Kollock, P., 1999. Communities in Cyberspace. Routledge, London.

Wenger, E., 1998. Communities of Practice. Cambridge University Press, Cambridge.