Conveying the Extent of Referential Intent

Christopher S. Komuves
Bell Labs Research, Lucent Technologies
600 Mountain Ave.
Murray Hill, NJ, US
+1 908 582 7856
ck@bell-labs.com

Paul C. Lustgarten
Avaya Labs Research
233 Mt. Airy Rd.
Basking Ridge, NJ, US
+1 908 696 5153
plus@avaya.com

ABSTRACT
In order to manifest one's intentions in the world, they must be translated into corresponding extentions of actions and objects. In a social system, this process might take the form of directives from a manager to her subordinates. Such directives could remain mostly in the realm of intention, or could be translated into more concrete terms.

In human-computer interactions, the human has traditionally borne most of the responsibility for this translation. When more of this responsibility is shifted to the computer, it becomes possible for the computer to participate more fully in the fulfillment of the human's intentions. Furthermore, the power of this intentional stance becomes available to the computer for use throughout its subsequent processing.

While discussion of intentions readily brings to mind goals, plans and other actions, every use of a referential term to identify some object is also an intentional act. In this paper, we seek to convey the extent to which such referential intent can be used to promote more effective human-computer interaction.

Categories and Subject Descriptors
*H.5.2 [**Information Interfaces And Presentation (e.g., HCI)**]: User Interfaces – interaction styles (e.g., commands, menus, forms, direct manipulation), natural language, theory and methods; H.1.2 [**Models and Principles**]: User/Machine Systems*; *D.2.2 [**Software Engineering**]: Design Tools and Techniques – user interfaces*; *H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – query formulation, retrieval models.*

General Terms
Design, Human Factors, Theory.

Keywords
reference, intention, extention, referential intent, implicit relational database, IRDB.

1. INTRODUCTION
Children begin interacting with the world around them by direct manipulation; they select an object by touching or grabbing it. They quickly expand their abilities to also refer to objects by pointing or other gestures. As they develop the ability to communicate through language, their referential abilities again expand to include a variety of linguistic forms—using proper names, descriptive phrases, pronouns and more conceptually complex or context-dependent expressions. Across this developmental progression, then, one might see the infant reaching for her bunny, the toddler pointing at her bunny, and the preschooler asking for or describing her bunny. While reference is frequently thought of as a property of language alone, the referential function of these non-linguistic modes is well-recognized and reflected in their characterization as ostensive reference [1]. These actions, linguistic or non-linguistic, serve a common communicative function of reference; they are referential acts.

Obviously, one of the salient characteristics of this developmental progression is the dramatic expansion of expressive power that it brings. While those earliest, non-linguistic forms of communication, including ostensive reference, continue to be part of the communicative repertoire of an adult, the use of language (or other symbol systems) brings with it the ability to refer far more expansively, to things that are not present in the immediate environment, to actions as well as objects, and to abstract concepts such as tomorrow or a department in a company. This in turn is crucial for the broader universe of abstract reasoning about concepts and categories of objects, which would otherwise be limited to reasoning about specific instances and individual objects.

We may think of any referential act as being comprised of two aspects. One aspect is a referential expression or term itself—which is to say, the proclamation of the intention to refer. The other aspect is that which is referred to by that expression. This later aspect, the target of the referential act, if you will, is the extention[1] of that referring expression. Note, of course, that not every referential act is successful; that is, there may in fact be nothing that fulfills the expressed intent—an intention may have no corresponding extention.

So, with regard to the child's developing communicative abilities, we can now observe that part of the power and sophistication children acquire lies in the differentiation of their intentional statements and references from their corresponding extentions. For instance, two referential expressions, embodying two different intentions, can both refer to the very same object, but with very different implications. For example, Suzy referring to Billy as "my best friend" or as "the kid who sits in front of me in science class" tells us very different things about Suzy's relationship to Billy.

[1] While the term extention is more often rendered "extension", we here use the former spelling to emphasize its complementarity with "intention", as well as to better distinguish this sense of the term from other, unrelated uses of the term.

WWW 2003, May 20-24, 2003, Budapest, Hungary.
ACM xxx.

Conversely, one referential expression can have a consistent intentional character (its "meaning"), while its extention varies. For example, the meaning of the word "tomorrow" cannot be grasped without understanding that the same (intentional) concept has different extentions, depending in this case on the day on which it is uttered. And yet, we don't regard the meaning of the word "tomorrow" to change from day to day, while its extention does.

So, what does this have to do with human-computer interaction? Those interactions, we contend, and the interfaces through which they are conducted, are replete with referential acts. Necessarily then, these referential intentions must be translated into corresponding extentions. Traditionally, however, the human has borne most of the responsibility for these translations. When more of this responsibility is shifted to the computer, it becomes possible for the computer to participate more fully in the fulfillment of the human's intentions. Furthermore, the power of this intentional stance [3] becomes available to the computer for use throughout its subsequent processing.

Our goal is to encourage others to join us in finding ways to preserve more of that underlying intention when committing referential acts towards computers. Towards that end, we will provide a conceptual perspective on what we mean by intentional HCI, followed by several examples illustrating these concepts. We then conclude with insight into how to architect such systems that preserve intention, and how this can lead to more effective human-computer communication.

2. INTENTIONAL HCI

We hold communication to be a collaborative process between intentional, cognitive systems, in the spirit of Grice [4] and Dennett [3]. We further hold that this perspective applies not only to communication amongst humans, but between human and computer, and in fact we would center the analysis of human-computer interaction as a communication process.

The issue we are examining falls firmly within the larger framework of the role of intentionality in effective communications, which we narrow to the role of referential intention in human-computer communication. This communication, as a form of collaboration, is bi-directional, and thus can also be appropriately regarded as a dialogue, even if not confined to a purely linguistic interaction of the sort more traditionally implied by that term.

What does it mean to convey intention through a user interface? To answer such a question, we find it helpful to first contrast two extremes in the possible character of human-computer communication. At one extreme, all matters of intention are solely in the purview of the human; the human's task in communication with the computer is to translate or decompose their intentions into constituent actions on the tangible, non-intentional plane that can be performed through that user interface. Correspondingly, the user interface designer's goal at this extreme is to provide an adequate set of concrete operations for the human to engage in this process, accompanied by a correspondingly concrete range of feedback that (it is hoped) the human can map back into the realm of their governing intentions.

At the opposite extreme, the human-computer interaction occurs through the exchange of intentional statements, and responses or other feedback to those intentions. In this model, the human's task is to articulate their intentions as clearly, concisely and accurately as possible. The interface designer's goal is to provide a means of expressing and differentiating a useful range of possible intentions, with sufficient feedback and participation to both affirm the accurate comprehension of the human's intentions, and adequately convey the results.[2]

While it is likely that neither of these extremes is possible (or desirable) in practice, we believe there is a qualitative difference between the opposing ends of the spectrum. Classical HCI seems inclined to reserve intentionality to the human[3]. We propose an alternate model of HCI, centered around intentionality, which we refer to as Intentional Human-Computer Interaction, or IHCI. This model encompasses both a reformulation of certain existing aspects of HCI, as well as illuminating the possibility of new modes (more intentional) for those interactions. These two models are depicted in Figure 1.

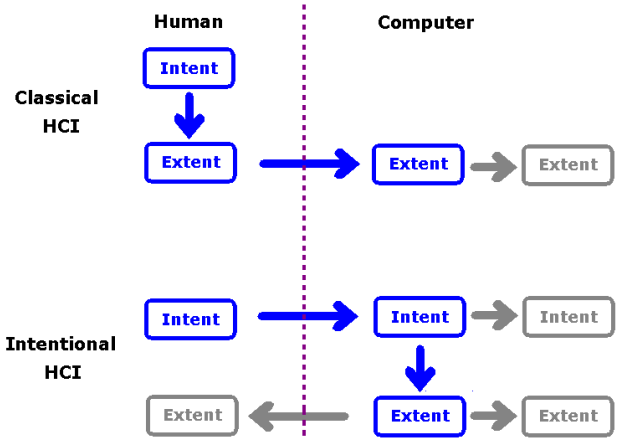

Figure 1: Classical And Intentional Models of HCI

The translation of intent to extent is an information losing transform; it is typically not possible to recover, even inferentially, the intentional dimension of an expression given only the corresponding extention. However, if a computer system can preserve that intentional dimension throughout whatever stages of processing are performed on that expression, while mapping or resolving its extention as needed during processing, then that intentional dimension can be used in subsequent processing as needed, in order to better interact at the intentional level.

The logically most direct way to enable such processing and subsequent interaction at the intentional level, almost to the point of being a precondition, is to structure the UI so as to maximize the amount of the human's intent that is conveyed across the barrier of that interface. This is in contrast to the seemingly very common and definitely strongly opposing design strategy of guiding (or forcing) the human to be the one to resolve their referential

[2] In practice, the central role of intentionality in human-computer dialogue would be counterbalanced by judicious and selective anchoring of those intentions into more concrete terms.

[3] Consider for example, Don Norman's 7 stages of action and his accompanying concepts of the gulf of execution and the gulf of evaluation [5]. While some of the best current thinking in the field, we find the discussion of these concepts permeated by a seeming presumption of an inability to directly communicate one's intentions to the computer.

intentions into some corresponding extention and then convey that extention across the interface. Such an extention-oriented design suffers on a number of counts, including increasing the interface's complexity, imposing more work on the human, increasing their response latency, reducing their efficiency, inducing more errors, and denying the application the ability to exploit the intentional dimension in its subsequent processing and interactions.

In contrasting classical HCI with IHCI, it's revealing to highlight an intermediate case in the handling of intention. That is, one way in which computer systems have become more sophisticated, acknowledging the significance of intentionality, has been for them to attempt to infer a user's intentions from the patterns of their concrete operations (the widely reviled Microsoft Office animated paper clip may be one of the best known examples of this trend). While such inferences may be made with varying degrees of success (more on which later), the point here is simply to note that to infer intent is conceptually and operationally different from conveying intent.

There are many layers of intentionality, since all use of symbolic communication (or some might say simply all communication) involves the underlying intentionality of meaning [1]. Most of our focus in this paper is on a more semantic level of intent, e.g., what mailbox did the human intend to identify by the address plus@avaya.com, or what person should be conferenced into an impending call corresponding to the string ck@bell-labs.com. There is another level that might be characterized by the question "what symbols did the human intend to emit?"—what is the intended surface form of this utterance, what words were spoken or typed, vs. the interpretation of those tokens[4]. This aspect of intentionality becomes quite prominent in the realm of speech recognition interfaces, where, (mostly) separate from any question of semantics, there is a non-trivial perceptual question of discerning which words the human intended the recipient to hear (i.e., what words did they say?). Even in the textual realm of a web form this step may be significant: witness the very useful practice of search engines and other query processing sites, such as google.com or dictionary.com, to respond to certain queries with a counter-query (e.g., google.com's response to "extention" of "did you mean **extension**").

Communication among humans and computers at the intentional level can occur in four basic relationships. These relationships can be classified by the nature of the agent possessing the intentions and the nature of the agent to whom those intentions are being conveyed:

Human → Human

Human → Computer

Computer → Human

Computer → Computer

We are here concerned with the second of these; we have barely begun to assess the philosophical (and sociological!) implications of the fourth.

3. EXAMPLES OF IHCI

We turn now to exploring several examples of human-computer interfaces, specifically with regard to the ability they provide their users to express intent. In particular, we look at the communication of referential intent, since in our view the essence of every referential act is the underlying **intention** to refer.

As we begin to assess the world of user interfaces with particular regard to the dynamics of intentionality, we see that essentially all of them allow intentions to be expressed to some degree. The question, then, becomes how, and to **what** degree, they allow such expression. And, going forward, the opportunity exists to design future user interfaces with a conscious regard for these dynamics—to provide intentionality intentionally.

In this section we present four examples. The first two are existing applications that (happen to) particularly exploit the direct expression of referential intent by their users; it's tempting to speculate how much the success of these applications derives from these intentional qualities. The second two examples are drawn from our current research projects and illustrate some further possibilities arising from consciously utilizing the expression of referential intent.

In reviewing these examples, pay particular attention to the benefits that come from preserving the intentional character of the referential act, rather than simply communicating the result of the referential act—that is, of conveying intention more than extention.

3.1 Specifying Email Recipients

For our first example, consider the specification of recipients for Internet email. There are two layers of recipient identifiers present in such email. At the layer closer to the human, the intended recipients of a message are designated by addresses, each of which is classified as a primary recipient, a secondary recipient, or a blind-copy recipient through its association with a `To:`, `Cc:`, or `Bcc:` header, respectively [2]. At the lower layer, the recipients of a message are designated through an otherwise undifferentiated list of addresses associated with the SMTP protocol element `RCPT TO:` [6].

The upper layer is a preservation of the user's original intent, while the lower layer is the extention of that referential intent. These layers are not redundant, but parallel and both quite useful. Independent of whether an address is listed in a `To:` or `Cc:` header, it results in exactly the same actions and processing—the message gets delivered to the mailbox(es) designated by that address. But the distinction is important at the level of the intentions of the originator and her expectations about the recipients' obligations in processing or replying to the message. The `RCPT TO:` list gets rewritten with forwarding and gets subdivided as copies of the message are split off from one another to be delivered via separate paths through cyberspace to their respective destinations.

Additionally, if a message has a `To:` header containing an address that specifies "members of dept X" and a `Cc:` header that specifies an individual member of Dept Y, then the recipient gets more information by knowing that intentional characterization

[4] This is a distinction between errors in the realm in automatic versus conscious processing akin to that identified and labeled by Norman as slips versus mistakes [5]; it may be necessary to expand this concept to include distinctions between errors in the **computer**'s automatic versus conscious processing.

than if she were simply to see the list of individuals encompassed by those designations. And yet, somewhat paradoxically, the recipient gets that "more" information through a smaller (possibly much smaller) collection of address terms.[5] Furthermore, this intentional level of recipient specification can be re-used at a later time, preserving that original intent. For example, a recipient of the original message might reply to that message a week later, during which time the extention of the original address has changed (someone has joined or left the department). This is an illustration of how an intention may be durable even in the presence of a changing extention.

Those familiar with a variety of email systems have likely observed alternative practices in the handling of addresses of mailing lists—in some cases the list's address is expanded, in other cases it's preserved. The tension between expanding the address of a list into the addresses of its constituents or not is the tension between using an intentional characterization or its corresponding extention.

3.2 Search Engines

The advent of the Internet has fostered the spread of an interesting class of applications that tend to preserve referential intent in their operation. These are full-text search engines, such as Google. When a word or phrase is entered, since the search is performed on written language itself, the result is to provide a response that is also within the realm of written language, and thus stays in the intentional realm. These results are useful because an intentional characterization, particularly a referential one, is likely to have been used by others. So, when successful, a search returns those documents that share the user's referential intent.

For instance, searching on Google for "the first president of the United States" returns roughly 7,000 matches. Most of the top matches are about George Washington, while some are about John Hanson (president of the United States under the Articles of Confederation). The effect for the user is one of bridging from her referential intent to the extention of the person described in the resulting documents. It's easy to miss the intermediate link in this bridge, formed by the presence in those same documents of both the referential characterizations and the identification of their extention.

Notice that the fact that a query can be ambiguous illustrates the interplay of intention and extention. If the human were forced to perform the resolution to a particular extention, the ambiguity would be lost, and with it, the opportunity to discover those alternate extentions (e.g., the other first president of the United States).

In a search engine then, a query forms a kind of referential expression, although not one organized by the syntactic structure of a well-formed descriptive noun phrase, nor necessarily the referential focus of a proper name, but a referential expression nonetheless. The search engine itself becomes a vehicle by which the human can explore the resolution of various intentional characterizations (i.e., her queries).

That the results of each query are accompanied by the query itself, allows the human to engage in a search dialogue with the option of iteratively refining the query, modifying it in some other way, or searching within the results. A useful characteristic of this dialogue is that it can remain at the intentional level as long as the human desires—the computer does not force it down to an extentional level.

In the case of Google, it also tries to infer intent. For instance, if one specifies a complete street address, such as "600 Mountain Ave., Murray Hill, NJ 07974", it will preface the regular search results with links to display the address entered on a map. Similarly, a query in the form of a phone number will preface the regular search results with any public directory entry matching that phone number, along with corresponding mapping links. The utility of these inferences illustrates the complementarity of responding to intentions in multiple ways—sometimes through direct expression, and sometimes through inference, just as humans do with each other.

We conjecture that the (partial) ability provided by search engines for expression of intent has fueled their utility and corresponding popularity, even if that ability was not consciously imbued by their creators.

3.3 Click2Dial Conference Manager

Our next example is a research prototype of a Web-based service for managing conference calls, written by Chris Komuves (see Figure 1). It leverages shared (communal) directory information to more easily establish, track, and manage conference calls. It incorporates strong authentication, use of registered contact numbers for participants, and real-time display of call participants to provide a much higher level of security than is possible with traditional conference call bridges.

One of the central functions of the user interface of this application is to perform the referential act of identifying people, and to do so in ways that best connect with the user's intent. We will highlight two user interface techniques that facilitate the expression of this referential intent.

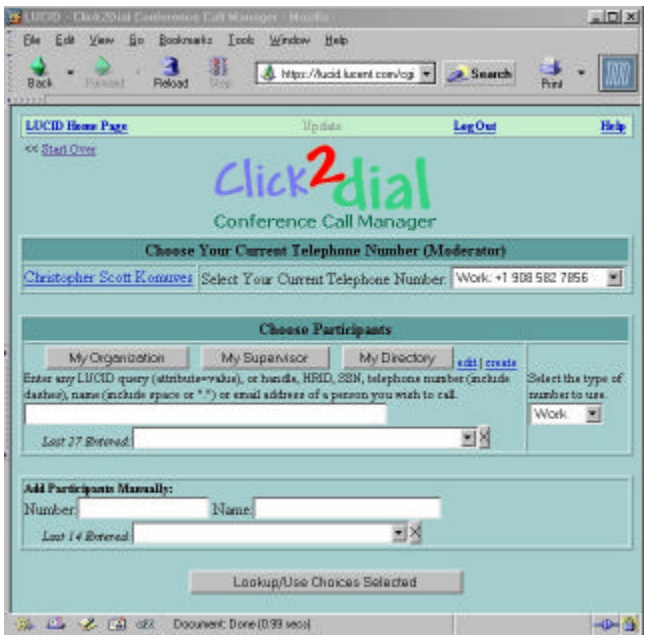

Figure 2: Click2Dial Conference Manager Prototype

[5] And yes, it's also important that the recipient be able to (independently) resolve those terms, to determine what their extention had been in the processing of this message.

Users can refer to people by any method, without being required to distinguish what particular method is being used. For example, rather than having six separate input boxes for each of first name, last name, employee ID, social security number (SSN), telephone number and email address, a single unified input box is provided to specify participants. The user can enter any reasonable identifier for a person or set of people. If what was entered happens to match a single person within the relevant domain, the application will report and use that match. If, however, there are multiple matches, all of the possible choices will be displayed with appropriate information to easily distinguish the person being sought. The effect for the user becomes one of sharing their referential intent with the application and then jointly, interactively, resolving the extension of that reference when there are multiple extentional candidates.

This unified reference box avoids requiring the human to classify the type of referential term they are generating. This makes data entry faster, since the user does not have to look for the appropriate box in which to enter the particular reference to a person she wishes to use. It also makes the UI less cluttered, and should reduce the likelihood of mode errors from entering the wrong identifier in the wrong box (e.g., entering an employee ID in an SSN input box). This eliminates many of the errors in the class of what Norman calls slips [5].

Consider the cognitive process of generating and conveying this referential term. Starting with an internal communicative intent of who is to be specified,[6] it's necessary to generate some referential term or expression identifying that person. If the UI requires separate (moded) entry of the possible referential terms, then the user must further self-reflect to assess the linguistic form (name, SSN, etc.) of the identifier they have chosen, and use that information to enter the identifier in the corresponding input box on the screen. Requiring self-awareness of the chosen linguistic form imposes an additional—and unnecessary—processing burden on the user.

Another dimension of referential intent expressible by the user is the role to be sought in contacting the identified party—"home", "work", or "mobile". The application uses this role as an "advisory" constraint, separately listing those matches to the primary identifier for whom no contact information is available under that role.

Another UI technique is the inclusion of shortcut buttons for people and groups that are in particular relationships to the user of the application. These buttons are "My Organization", "My Supervisor", and "My Directory". The meaning (intent) of each of these buttons is consistent for all users, but the expansion (extention) of each varies by user. Such context-dependent reference (what linguists formally call relational deixis) facilitates identification of the most common referents for each person—the members of her organization, her supervisor, and her personal directory of people she commonly interacts with. Another advantage of both the directory linkage and deixis here is that as the extentions of the references change, the user does not have to attend to those changes. If the user gets a new supervisor, or a person leaves or joins that person's organization, etc., there is no change required in the referential act that must be committed by the user.

3.4 Daisy Bell

Our final example steps out of the visual domain of text and graphical interfaces, into a speech interface. By observing the dynamics of how referential intent can be conveyed in the speech domain as well, it becomes clearer that these dynamics apply across all modes of human-computer interaction.

Daisy Bell, a research project led by Paul Lustgarten, is a virtual personal assistant accessed through spoken dialogue over a telephone. In contrast to speech systems intended for more casual/incidental use, such as a vendor's customer inquiry system, a personal assistant is intended for repeated, daily use by a relatively closed set of users. This shapes and heightens the requirements for linguistic precision and habitability, or naturalness, of the dialogue; we are currently investigating the role of "referential richness" in fostering such naturalness.

In the context of Daisy's current topical domain of managing a personal calendar, consider the following short dialogue segment that might occur between Daisy (D) and a human (H) on Thursday, May 22nd, 2003:

> H: Schedule a two-hour meeting tomorrow at noon.
>
> D: I've scheduled a meeting for tomorrow at noon, lasting two hours.
>
> H: Reschedule it for the fourth Friday in May.
>
> D: Excuse me—the fourth Friday in May is the same day as tomorrow, when it was already scheduled.

In this exchange, the human has used two very different referential expressions—"tomorrow" and "the fourth Friday in May"—that happen to have the same extention, or referent; a date that we might otherwise identify as May 23rd, 2003.[7]

Consider what's going on in this example. First, at the subjective, experiential level, Daisy's corrective response in this exchange always elicits startled laughter from onlookers during demonstrations, with subsequent reports of feeling (and we use that term quite specifically) the presence of a deeper understanding than expected. Why? At the extentional level, she is merely reflecting the vacuity of changing the date of an appointment to its current setting; a mildly sophisticated bit of feedback, perhaps, compared with merely saying (truthfully but less helpfully) "OK", but hardly shocking for a contemporary computer system.

However, this exchange is not limited to a purely extentional level. On hearing her corrective feedback, it is suddenly revealed that she must have been tracking, in parallel, **both** the intentional and extentional levels of the human's temporal references: accepting these two date expressions from the human, doing the obvious step in cooperative communication of discerning (or perhaps presuming) that they are **intended** to refer (and not just refer to anything, but refer to a particular type of thing, a date), and mapping those references to an extentional level on

[6] An intent that might be expressed in internal "mentalese", if such a thing **were** to be expressed, as "I want to convey to this application my choice of what person(s) I want added to this conference call."

[7] That these two expressions co-refer is dependent, of course, on the date on which the first is uttered, given that the referent of "tomorrow", as an instance of what the linguists call *temporal deixis*, is dependent on the temporal context of its utterance.

which their temporal equivalence is subsequently discerned. And then, to generate the feedback statement itself, this process must be reversed, in a sense, since the informational feedback of the equivalence is most helpful (and linguistically felicitous) only back at the intentional level, by explicit appeal to the intentional form of these two very different referential phrases. So Daisy must not only have correctly interpreted these phrases as she heard them, but must also have retained them in correspondence to their extention and the overall task/dialogue, in order to appropriately invoke them in the subsequent dialogue.

To see this last point more clearly, consider, in contrast, a possible alternative response, centered on the extentional level:

> H: Reschedule it for the fourth Friday in May.
>
> D: The 23rd of May is the same day as the 23rd of May, when it was already scheduled.

While completely accurate, there is clearly something missing in this hypothetical response. While this deficiency could be glossed over by some judicious rephrasing of this extentional-level response, that it's there to **be** revealed or hidden is the telling point. That is, even having correctly interpreted the two temporal phrases and noted their equivalence, the exchange would be deficient without completing the loop by also **responding** in intentional terms corresponding to those of the human—thereby meeting the human on our own terms, matching us as intentional systems.

4. Prolegomena of a Practice of IHCI

If we have been at all successful in the previous sections suggesting that there is something to our proposed concept of Intentional HCI and that various extant and emerging applications can usefully be seen to embody some of the principles thereof, the question arises of how IHCI might generally guide us in the creation of new applications. While we have no comprehensive answer to this question, we invite the field at large to take up the pursuit. In this section, we offer some observations and suggestions in support of that pursuit.

4.1 Principles of Intention-Friendly Interfaces

Intentionality is one of the most distinguishing characteristics of consciousness and intelligence, and as such is one of the least amenable to simple codification and technological reduction. We thus regard the design of "intention-friendly" user interfaces as an art, still lacking a rich and informative set of precedents and exemplars. This art is focused not on the esthetics of the layout of the screen, but on the character of the interaction and what's required of the computer to create that character. What follows is a list of principles and guidelines that we have identified to date for the design of computing applications capable of engaging in an intentional dialogue with their human users.

Engage at the intentional level. At the most basic level, in order for a UI to deal with intentionality, it must first be able to receive intentions from the user. This may take many forms, including typed or spoken natural language interfaces, flexible input boxes that allow any form of identification to be entered, or UI elements that allow a richer set of references to be performed—such as referring to dates as "tomorrow" or "last Wednesday", or people as "my supervisor" or "my secretary".

Bridging from intention to extention requires (topical) knowledge. In order to transform an intentional reference into a set of corresponding extentions, an application must possess some additional knowledge about the domain of relationships relevant to the task at hand, more than it would if it did not have to deal with intentions. This knowledge may include work flows, database structure, task structure, grammatical or semantic structure of a human language, or any other topical knowledge. For instance, an application like Click2Dial (see section 3.3), must know something about the formats of various identifiers for people in order to determine what type of identifier was used by the human—that an SSN is a sequence of 9 numeric characters that might include some dashes, that email addresses begin with a username followed by an @ sign, etc. Search engines are also a non-obvious case of this, since they leverage the topical knowledge in the documents they index. The more knowledgeable the computer is, the better able it is to interpret the human's intention.

As this conversion of epistemological state requires knowledge to proceed, and by the nature of the transformation looses information, it can be thought of as akin to how the conversion of water from a crystalline solid to liquid simultaneously requires the input of energy (the knowledge required to go from intent to extent) while losing the information of the crystalline structure (of the intentional character) as when going from tomorrow to "Friday, May 23rd".

Be stateful. Since the process of resolving an intention may be iterative, there is a greater need for preservation of state within the system, such as from one page to the next in a Web application, or from one statement to the next in a voice application. Focus is often critical for the expression of intention. Preserving focus preserves the human's intention. To the degree to which the interaction between human and computer is more of a dialogue, there is more of a need to track context.

One common use of state information to facilitate fulfillment of intention is the retention of focus in file tree locations by most applications. If you have previously saved a file into a particular directory, most applications will remember that focus the next time you try to save a file, and place you by default in that directory you last saved a file to. This may also take the form of cookies for individual sessions, or centralized profile information for globally manipulated data. Click2Dial, for instance, uses cookies to store the focus of whether the person using each particular computer is currently using her home, work, or mobile phone number, and centralized storage for directory information and history of prior actions. This allows state information that is more closely tied to location to be coupled with each individual computer used, whereas state information that is more global in nature is, appropriately, stored globally.

Preserve the user's intention. Applications need to retain the intentional characterization received from the user for use in subsequent processing. This is related to the need for statefulness, and is central to creating UIs that interface with humans at the level of intention. For example, an email system that preserves the original form of a recipient specification in a `To:` header serves to honor the user's intention across time (e.g., when replying to that message two weeks later). Similarly, having bookmarked a URL listing events happening "tomorrow", each future invocation of that URL should always display the event listing for the day following that invocation.

Offer to advance the dialogue on both the intentional and extentional levels in tandem. It's important for the interaction to be a dialogue in which there is an iterative process of the human and computer working together to hone the intentional expression & view/browse the possible extensions of that expression. The conveyance of intention is an ambiguity-laden process, and one of the key ways for a human to know that she's been successful in conveying her intention is to see the extentional consequences laid out in front of her . So the UI design process becomes one of providing access to these two facets in parallel, with the human free to shift attention dynamically from further refinement of conveying their intention to the consequences of applying that intention. The computer should participate in the resolution/interpretation/application of that intention, progressively refining its "understanding" of the human's intention. This might be as simple as the familiar form of interaction with a search engine, in which the (intentional level) query is presented on each page of results, with the user free to refine the query, search within it, or shift her attention to the (extentional) list of matches.

Dare to be linguistic. Language is one of the most effective and most widely used methods we have for expressing intentionality. It can therefore be a potent element in intentional UIs [7], even when interwoven in discrete pockets in an otherwise graphical display. Particularly in communicating our referential intentions, there is much to be gained by exploiting the referential richness of language. Two dimensions of this richness we want to highlight are the size of the *referential field* and the variety of *referential forms* available to the user. Increasing either of these expands the expressive power of the interaction. For referential fields, this means doing something like resolving a reference to a person against an enterprise-wide directory instead of a much smaller personal directory (or, even better, against a composite of the two). For referential forms, this means something like allowing SSNs, email addresses, telephone numbers and names in a single input box as Click2Dial does. Or, more linguistically (as we saw with Daisy Bell in section 4.4), it means accepting proper names, descriptive noun phrases, deixis, and anaphora, instead of the more limited choice of selecting from a selection box or menu.[8] The more referentially rich a UI, the more easily and efficiently its users can convey their referential intent.

Intentionality awareness is architecturally significant. The parts of the application that are aware of intentionality should be architecturally differentiated from those parts that are not. We actually find it useful to classify the architectural elements of a system into three categories: those that are responsible for processing intentionality (and intentionality only), those that bridge between the intentional and extentional realms (knowing about both), and those that have nothing to do with intentionality. Having so classified them, we recommend separating these classes from each other, and coalescing the members of the first class—those that are intentionality-aware—with one another.

4.2 Architectural Example: IRDB

In our experience of recent years, building systems with more and more regard for intentionality, we've made use of a particular tool that has been instrumental in our results in this domain. The tool to which we refer was invented at Bell Labs in 1988 by Michael Baldwin and is called an Implicit Relational Database, or IRDB.[9] This tool has proven to be profoundly useful as a bridging element in the style of the architectural distinction introduced in the prior section, allowing us to write more focused intentionality-oriented code by relying on the IRDB for the translation of referential intent to extent.

First a word about IRDB itself. The distinctive aspect of IRDB relative to other relational database technologies is that the query language is essentially schema-less, requiring only the specification of a set of attributes whose values are known and a set of attributes whose values are desired. All of the table joins to obtain those desired attribute values are dynamically determined by the IRDB itself, rather than through the explicit direction and specification of the governing application. This technology has been used to create a materialized meta-directory of our corporate personnel data for the last 14 years for AT&T and its descendents, integrating data from dozens of sources and now serving over 40 million queries a month in Lucent alone. The most advanced application of IRDB has been the Lucent Universal Corporate Internal Directory (LUCID), a research project led by Chris Komuves for the last 6 years, and used by more than 90,000 people. LUCID contains more than two million records spanning roughly 100 tables and over 500 attributes. Click2Dial is built upon the LUCID platform.

So, how does IRDB help with the development of effective IHCI? The most important feature is the freedom from specifying table selects or joins in queries—that is, the application developer needn't specify which table an attribute is in, nor which tables to join in order to resolve a query. This creates a "use what you know" style of query capability, which has proven to be easy to map to user-level intentions. For example, a query for the names of experts in Perl who work in Murray Hill ("expertise=Perl|city=Murray Hill") is resolved through linking the attributes *expertise* to *handle* to *location code* to *city*, and then to *handle* to *name* to return the requested results. In conjunction with some much earlier work on a conference-call manager, Paul Lustgarten wrote a small, yacc-based front-end that determined the referents of typed, natural-language descriptive noun phrases by translating them into IRDB queries against LUCID.

The following thought experiment illuminates the architectural significance of interplay of the preceding architectural principles with a tool such as IRDB. Imagine that an enterprise had an application (say, for managing conference calls) based on its enterprise directory and then consider the impact of adding access to national or other public data as well. The field of possible extentions to the referential expressions in that application (the referential field of those expressions) would be greatly expanded.

[8] It is often thought that pointing, or selecting from a list, is a qualitatively different operation than these linguistic forms of reference. However, as noted in the opening of this paper, pointing is but another (pre-linguistic) form of committing a referential act—thus, its designation as ostensive reference. The most comprehensive interface will allow both these linguistic and non-linguistic referential forms.

[9] The IRDB technology is freely available from Lucent as a supplemental package (called PQ) to the Plan 9 open source distribution, available at http://plan9.bell-labs.com/.

But that expansion of the referential field would occur (mostly) independently of the portion of the application that works with the human's intentions. This degree of independence illustrates the utility of separating the intentionality-processing portion of the application from the portion that resolves those intentions to specific referents (the IRDB). However, this expansion of the referential field would likely require some modicum of changes to the intentionality-oriented portion of the application. The nature of these changes further illustrates our point: the primary changes to the intentional level of the application would (or should) be linked specifically to the ways in which the expansion of the referential field itself interacts with the **intentions** of the human.

For example, the application (designer) might in effect now need to ask the user, is it your intent to limit the participants of the conference call you are about to initiate to fellow associates within your enterprise (which, prior to the expansion of the underlying directory data, was implicit in using the application at all), or not? While adding this distinction to the user interface of the application may seem like an obvious consequence of the postulated expansion of its referential field (through adding extra-enterprise directory data), imagine trying to convey this new distinction across that interface at the **extentional** level of which telephone numbers are "in" and which are "out": "Did you mean to include +1 908 476 1100 in your conference call?" How would the human even know how to answer? And yet, back at the intentional level of enterprise communication policies and intellectual property rights, this distinction between fellow associates (who are safely presumed to be bound by appropriate agreements restricting disclosure of proprietary information) and others not so bound is an important and salient distinction—if, that is, the call being initiated is for the purpose of communicating proprietary material, and not for some non-proprietary purpose, such as contacting a vendor, or participating in an industry-wide standards meeting, etc.

This example further illuminates the potential interaction of intentional-level constructs with one another—in this case, membership in a legally-defined set of people, on the one hand, and the planned (might we say intended?) agenda of a call. One of these constructs is known and significant to the application (the field of potential participants in the call), while the other is not (the agenda of the call). It is clearly the human who must assess the interaction of these two elements. Our point is that the application's role—we would even say responsibility—is to participate in that assessment, to both present information **to** the human and receive instructions **from** them, at the level in which this assessment must occur…which is to say, the intentional level. And, having achieved clarity at that level, to bear the brunt of the effort in resolving those intentions to the extentional level (with due transparency and opportunity for clarifications or corrections, of course…themselves fittings subjects for some other paper).

5. CLOSING REMARKS

We've discussed intentionality of referential acts, but we believe the points we've made apply more broadly to all aspects of intentionality in communication.

Classical HCIs place responsibility for the transformation from a user's referential intent into an explicit extent upon the human user. When this responsibility is shifted to the computer, the full information content of the original intent can be conserved throughout the entire system.

Communication only through extent is crude, clumsy, and inefficient. Luckily, however, this extreme is not actually possible, and even the most ardent direct manipulation interface has been forced to accommodate referring expressions for filenames, URLs, etc. Communication through intent can make user interfaces more robust, efficient, precise, and adaptive (cf. learning, personalization and use of context). The conveyance of intent is an essential aspect of human to human communication, and is not yet realized in most human to computer communication.

The natural evolution of user interfaces is to become more oriented towards expressions of intent from users. Being aware of this shift and why it is important and desirable will allow developers of user interfaces to create superior applications and drive this natural evolution forward more quickly and consciously. We must expand the intentional envelope to include the computer, so that it keeps track of that intentional dimension. The challenge is to explicitly design our computers to adopt the intentional stance.

Intentionality is central to the effectiveness and power of communication, whether amongst humans or between humans and computers. Bringing intentionality into our communications with computers is like bringing language skills to our growing children, enabling them to ask for what they want, rather than merely pointing. Looking at their ability to process intentionality from a developmental perspective, our computer systems and their interfaces are still in their infancy. It is time for them to grow up.

6. ACKNOWLEDGMENTS

Michael Baldwin, for countless hours of insightful exhortations on the intricacy, power, and capabilities of IRDB, system architecture and the "proper" role of computers in relation to humans.

7. REFERENCES

[1] Bruner, J. Acts of meaning. Harvard University Press, Cambridge, MA, 1990.

[2] Crocker, David H. (revised by). RFC 822 "Standard for ARPA Internet Text Messages". August 1982

[3] Dennett, Daniel C. The Intentional Stance. The MIT Press, Cambridge, MA, 1987.

[4] Grice, H. P. Logic and Conversation. In Cole, P., Morgan, J.L., eds.: Syntax and Semantics: Vol. 3: Speech Acts. Academic Press, San Diego, CA, 1975.

[5] Norman, D. A. The design of everyday things [previously published as The psychology of everyday things]. Doubleday, New York, 1990.

[6] Postel, Jonathan B. RFC 821 "Simple Mail Transfer Protocol". August 1982.

[7] Scoville, Thomas. "The Elements Of Style: UNIX As Literature". In Performance Computing. September 1998.