HUMAN-COMPUTER INTERACTION, 2000, Volume 15, pp. 75–106

Let's Stop Pushing the Envelope and Start Addressing It: A Reference Task Agenda for HCI

Steve Whittaker, Loren Terveen,
and **Bonnie A. Nardi**
ATT Labs–Research

ABSTRACT

We identify a problem with the *process of research* in the human–computer interaction (HCI) community–an overemphasis on "radical invention" at the price of achieving a common research focus. Without such a focus, it is difficult to build on previous work, to compare different interaction techniques objectively, and to make progress in developing theory. These problems at the research level have implications for practice, too; as researchers we often are unable to give principled design advice to builders of new systems. We propose that the HCI community try to achieve a common focus around the notion of *reference tasks*. We offer arguments for the advantages of this approach as well as consider potential difficulties. We explain

Steve Whittaker is a cognitive psychologist with interests in the theory, design, and evaluation of collaborative systems and speech access; he is a senior research scientist in the Human Computer Interaction Department of AT&T Labs–Research, Florham Park, NJ, USA. **Loren Terveen** is a computer scientist with interests in recommender systems and online communities; he is a research scientist in the Human Computer Interaction Department of AT&T Labs–Research, Florham Park, NJ, USA. **Bonnie Nardi** is an anthropologist with an interest in social networks and activity theory; she is a researcher in the Human Computer Interaction Department of AT&T Labs–Research in Menlo Park, CA, USA.

CONTENTS

how reference tasks have been highly effective in focusing research into information retrieval and speech recognition. We discuss what factors have to be considered in selecting HCI reference tasks and present an example reference task (for searching speech archives). This example illustrates the nature of reference tasks and points to the issues and problems involved in constructing and using them. We conclude with recommendations about what steps need to be taken to execute the reference task research agenda. This involves recommendations about both the technical research that needs to be done and changes in the way that the HCI research community operates. The technical research involves identification of important user tasks by systematic requirements gathering, definition and operationalization of reference tasks and evaluation metrics, and execution of task-based evaluation, along with judicious use of field trials. Perhaps more important, we have also suggested changes in community practice that HCI must adopt to make the reference tasks idea work. We must create forums for discussion of common tasks and methods by which people can compare systems and techniques. Only by doing this can the notion of reference tasks be integrated into the process of research and development, enabling the field to achieve the focus it desperately needs.

1. THE PROBLEMS WITH HUMAN–COMPUTER INTERACTION AS RADICAL INVENTION

Research in human–computer interaction (HCI), particularly as embodied in the CHI conference, focuses largely on novel problems and solutions that push the technology envelope. Most publications describe novel techniques or novel applications of existing techniques. A study by Newman (1994) provided quantitative evidence for this. He compared CHI with five other engineering research fields, such as thermodynamics and aerodynamics. He used content analysis to classify abstracts of published articles in terms of the type of contribution they made to the field. He found that in other engineering disciplines, over 90% of published research built on prior work. There were three major ways that research efforts could extend published work: (a) better modeling techniques (used for making predictions about designs), (b) better solutions (to address previously insoluble problems), and (c) better tools and methods (to apply models or build prototypes). The picture was completely different for HCI. Newman (1994) conducted a similar analysis of CHI abstracts for the 5 years from 1989 to 1993, attempting to classify abstracts as describing one of the three types of enhancements previously identified. However, only about 30% of articles fit into these categories of developing prior work. The majority of CHI articles either reported "radical" solutions (new paradigms, techniques, or applications) or described experience and heuristics relating to radical solutions.

1.1. Radical Invention Is Not Always Effective

This analysis strongly suggests that CHI is different from other engineering research disciplines. But, is this good or bad? Is it a problem that our field is dominated by attempts at radical invention, apparently crowding out the practice of "normal science" (Kuhn, 1996)? Or is it a virtue? We offer arguments that the current state of affairs is problematic based on two different criteria for success in our field.

One criterion for success that is consistent with the radical invention approach is *technology transfer*. A strong motivation for constant innovation is the example of whole new industries being created by user interfaces (UIs). People are aware that applications such as Visicalc and Lotus® 1-2-3 drove the early PC market, and Mosaic/Netscape® led to the Web explosion. In this view, HCI research is an engine room from which novel interaction techniques are snatched by waiting technology companies; or better yet, researchers start their own companies. There are undoubtedly successes originating from within the HCI community, including UI toolkits and general programming techniques (Rudisill, Lewis, Polson, & McKay, 1996), as well as the ideas

and technology underlying collaborative filtering (Goldberg, Nichols, Oki, & Terry, 1992; Hill, Stead, Rosenstein, & Furnas, 1995; Resnick, Iacovou, Suchak, Bergstrom, & Riedl, 1994; Resnick & Varian, 1997; Shardanand & Maes 1995). The graphical user interface (GUI) for the personal computer developed at PARC in the 1970s successfully combined together ideas that predated the HCI community such as overlapping windows and the mouse (Smith, Irby, Kimball, Verplank, & Harslem, 1982).

Nevertheless, the UIs that have had the most widespread impact have generally come from people outside the HCI community (Isaacs & Tang, 1996). Visicalc was invented by a business student and a programmer. CAD systems developed out of Sutherland's (1963) work on Sketchpad and also seem to have been independently invented by engineers at companies such as Boeing and General Motors (Foundyller, 1984). America Online® and Instant Messenger(sm) were invented by business people. Tim Berners-Lee, the inventor of HTML and the Web, is a computer scientist but was not a member of the HCI community.

The second criterion for success is a scientific one. The radical invention model has not aided the development of a "science" of HCI. This is a controversial area with acrimonious past debate concerning the scientific basis of HCI (Carroll & Campbell, 1986; Newell & Card, 1985), and extended arguments about the relation of HCI to psychology and cognitive science. It is true that there are isolated pockets of research that derive basic precepts from psychological theories (Card, Moran, & Newell, 1983; Gray, John, & Atwood, 1993; Olson & Olson, 1990). However, these articles are in the minority (as is evident from the Newman, 1994, analysis), and it is unclear that they have major effects on mainstream HCI practice (Landauer, 1995; Newman, 1994). The analysis so far should make it clear why this is so. The field cannot consolidate if everyone constantly is striking off in new directions. Although radical invention is vital to making progress, so too is research that builds on the work of others. When radical invention (whatever its source) opens up new conceptual territory, that territory must be settled. Concepts must be clarified, trade-offs determined, key user tasks and requirements described, metrics or critical parameters (Newman, 1997) identified, and modeling techniques constructed. We are simply not doing enough of this type of work.

1.2. What We Don't Know: Requirements, Metrics, and Uses of Everyday Technologies

The most significant problem caused by the lack of cumulative research is that the field is not gaining a clear understanding of core user tasks, interactive technologies, and techniques. When we consider some of the tasks that are es-

sential to people's everyday computing activities–including information browsing on the Web, retrieval and management of Web information, use of e-mail and voicemail, personal information management, and task management–we find little systematic information[1] about these tasks. Although there are many radical solution attempts in these areas, we do not have accepted bodies of knowledge about everyday computer activities. Examples include how and why people search for information, how they maintain information, how they organize their digital desktops and personal data, how they choose different communication technologies and organize communication tasks, and how they manage and schedule tasks with and without computers. In the majority of these cases, although initial studies have been conducted, there is no clear consensus about user tasks, no commonly held view of outstanding issues and problems, and no accepted success metrics. Thus, when addressing these problems, researchers often have to start from scratch in defining their version of a problem, requirements, and evaluation metrics. This difficulty is manifest in areas such as information retrieval interfaces (Amento, Hill, Terveen, Hix, & Ju, 1999; Whittaker et al., 1999), asynchronous communication interfaces (Whittaker, Hirschberg, & Nakatani, 1998a; Whittaker & Sidner, 1996), and desktop UIs (Barreau & Nardi, 1995). This makes it difficult to focus research on real shared problems, to compare research results, and to tell when a new solution is better rather than simply different (Newman, 1997).

A well-known problem with the radical invention approach is that, without empirical analysis to identify requirements, researchers can end up proposing radical solutions to things that users do not consider to be major problems and can neglect major problems that users do experience. For example, Barreau and Nardi (1995) studied how users organized information on the computer desktop. In open-ended interviews, they found that most people felt that their files were adequately organized so that archiving tasks were not perceived as requiring major support. Despite the absence of perceived user problems with archiving, much recent work has addressed the issue of support for this task (Fertig, Freeman, & Gelernter, 1996; Gifford, Jouvelot, Sheldon, & O'Toole, 1991; Rao, Card, Johnson, Klotz, & Trigg, 1994). On the other hand, many people experienced problems in moving data around between applications. Here, basic empirical investigation uncovered an important task that was not being addressed by the research community. This insight led to work on Apple Data Detectors (Nardi, Miller, & Wright, 1998), now a part of the

1. By systematic bodies of knowledge, we employ the very weak criterion that at least two studies have been conducted in a given area. Note that we are not even insisting that the studies agree on their core findings. Our informal observations are that there are often one or two pioneering studies in a given domain, after which no further research gets done.

Macintosh operating system. The research also identified a second requirement that desktop organizers should support, namely *reminding*. By simply looking at their folders and files, users were reminded of outstanding tasks. This too has general implications for desktop UIs. UIs that present alternatives to the folders and files metaphor need to address the reminding function. This research thus discovered two novel user problems (and hence criteria for evaluating new versions of desktop organizers), as well as finding that a commonly addressed technical problem–archiving–requires less support.

In addition to a lack of shared task descriptions and sets of requirements, we also have little systematic data about how people use popular technologies. We lack information about how people actually use e-mail systems, voicemail systems, cellular phones, the Windows interface, digital personal organizers, and instant messaging.[2] There may be one or two studies in each area, but there is hardly a body of robust knowledge. Given the popularity of these technologies and the frequency with which they are used, it would be useful to know how people use them, what they use them for, how successful they are, and where their problems lie.

Furthermore, we do not have a good understanding of why certain core UI techniques are successful. For example, GUIs are central to the enterprise of HCI, and although we have successful guidelines for building GUIs (Shneiderman, 1982), we still do not understand why they are successful (Baecker, 1987; Brennan, 1990).

Of course, as radical solutions continue, forays into new areas such as immersive virtual realities, augmented realities, affective computing, and tangible computing simply make the problem worse. Not only do we not understand these new technologies and their basic operation, we do not have a clear sense of how much innovation is tolerable or desirable. There may be limits on individual and social capacity to accept radical innovation. Many people have invested considerable time in learning to use specific hardware and software interfaces and may be resistant, for good reasons, to novel technologies. How quickly can school systems, for example, absorb radical technical change? Is radical innovation the way forward, or will incremental changes or extremely easy to use applications have more impact? Do continued radical inventions distance us from one another as the global society emerges? Without understanding basic computing tasks, we cannot address any of these questions. In sum, although we lack basic understandings of current users, tasks, and technologies, the field is encouraged to try out even more radical solutions without

2. One complicating factor here is that some studies of these technologies have been conducted in industrial contexts, and these results have usually ended up being proprietary rather than public. Nevertheless, we still need publicly available data about technologies that are used by millions of people multiple times a day.

pausing to do the analysis and investigation required to gain systematic understanding.

1.3. How We Don't Know It: The Dissemination Problem

Furthermore, even when a useful body of knowledge does exist for a core task, the HCI community does not have institutions and procedures for exploiting this knowledge. One major change in practice that we advocate is the institution of workshops for articulating knowledge of core tasks and practices for disseminating such knowledge. We also suggest that changes in community standards–for example, reviewing guidelines for the CHI conference and in HCI instruction (both at universities and in various professional tutorials)–will be necessary for a new way of doing things to take hold. These are the methods by which our suggestions can be institutionalized.

2. THE REFERENCE TASK SOLUTION

To address the overemphasis on radical invention and lack of knowledge about important tasks, we propose a modified methodology for HCI research and practice centered on the notion of *reference tasks*. Our proposal has both technical and social practice aspects. We discuss (a) how reference tasks may be represented and used by individual researchers or practitioners, and (b) new practices that the HCI community must adopt to develop and utilize reference tasks.

The goal of reference tasks is to capture and share knowledge and focus attention on common problems. More specifically, by working on a common set of tasks central to HCI, the community will enjoy a number of benefits:

- We will be able to agree on a set of tasks that are central to the field and worthy of sustained investigation; by focusing on a common set of tasks and problems, and developing a shared body of knowledge, the field will be able to assess progress and achieve more coherence in our collective efforts.
- More specifically, the community can share problem definitions, datasets, experimental tasks, user requirements, and rich contextual information about usage situations.
- We can agree on metrics (e.g., critical parameters; Newman, 1997) for measuring how well an artifact serves its purpose; this will enable researchers and designers to compare different UI techniques objectively and to determine when progress is being made and where more work is required.

- Researchers will have a sounder basis for giving advice to designers; they should be able to identify core tasks within a domain, the importance of the tasks, metrics for measuring how well an artifact supports the task, and the best-known techniques for supporting the task.
- Researchers will have a basis for developing theory; when we know the relation between critical tasks and their subtasks, interface techniques and critical parameters, we have the basis for a predictive model.

Our proposal partly overlaps with those of Roberts and Moran (1983) and Newman (1997). Roberts and Moran proposed standard tasks be used to evaluate word-processing applications. Our proposal differs from theirs in being independent of a specific application. Newman (1997) suggested the use of critical parameters as a method of focusing design on factors that made critical differences to UI performance. We are motivated by Newman's (1994) original findings and applaud the simplicity of focusing on a single factor–namely, critical parameters. However, we offer a broader approach that emphasizes the relation between requirements, reference tasks, and metrics. Newman's (1994) account is unclear about the methods by which the tasks relevant to critical parameters are chosen. Furthermore, one of our concerns is that metrics may be task specific rather than general as his approach would seem to imply. Finally, we are concerned with the social and institutional processes required to make this approach work–in particular, how researchers can jointly identify reference tasks, collect data, analyze the tasks, and disseminate and make use of the results.

2.1. Reference Tasks in Other Disciplines

To motivate our approach, we discuss several case studies from other disciplines. We trace the role of related concepts in speech recognition and information retrieval in some detail as well as briefly mention digital libraries and machine learning

Speech Recognition (The DARPA Workshops)

Until the late 1980s, speech recognition research suffered from many of the same problems we have pointed out in HCI research. Researchers focused on different tasks and different datasets, making it difficult to compare techniques and measure progress. Then, 10 years ago, the Defense Department's Advanced Research Projects Agency (DARPA) organized an annual workshop series that brings researchers together for a "bake-off" to compare system performance on a shared dataset (Marcus, 1992; Price, 1991; Stern, 1990;

Wayne, 1989). A dataset consists of a corpus of spoken sentences defined and made available to the researchers in advance of the bake-off. The data contain both "training data"—sentences that can be used to train the system (i.e., to tune its performance)—and "test data"—sentences on which the systems' performance is measured. The initial task simply was to recognize the sentences in the corpus. The systems did not engage in dialogue, and there were no real-time constraints, with the metric being the number of correctly recognized words in the corpus.

At each bake-off, each participating group presents and analyzes the results of how their system performed. The utility of different techniques can thus be quantified, making it possible to show that some techniques are better for certain types of data, utterances, or recognition tasks. All interested researchers get an annual snapshot of what is working, what is not working, and the overall amount of progress the field is making.

Progress has indeed been made. Initial systems recognized small vocabularies (1,000 words), had (sometimes extremely) slow response times, and had high error rates (10%). Current systems recognize much larger vocabularies (100,000 words) and operate in real time while maintaining the same error rate and recognizing increasingly complex spoken sentences. Furthermore, as system performance has improved, more difficult tasks have been added to the yearly bake-offs. Early systems were tested on monologues recorded in high-quality audio, whereas more recent tasks include dialogues recorded in telephone-quality speech. More recent developments include a series of workshops attempting to extend these methods into more interactive settings using the approach advocated in Walker, Litman, Kamm, and Abella (1998).

There are also benefits derivable from the existence of shared speech datasets, independent of the use of those datasets in the annual bake-offs. The speech community now has a common easily accessible shared dataset, which has led to standard ways to report results of research taking place outside bake-offs. These independent studies now report their word error rates and performance in terms of shared datasets, allowing direct comparison to be made with other known systems and techniques.

Information Retrieval (The TREC Conferences)

Information retrieval is another discipline in which a core set of tasks and shared data have been used to successfully drive research. The Text REtrieval Conference (TREC; Voorhees & Harman, 1997, 1998), sponsored by the United States National Institute of Standards and Technology (NIST), plays a role analogous to the DARPA speech recognition workshops.

As with the DARPA workshops, a major goal of TREC was to facilitate cross-system comparisons. The conference began in 1991, again organized as a bake-off, with about 40 systems tackling two common tasks. These were *rout-*

ing (standing queries are put to a changing database, similar to a news-clipping service) and *ad hoc queries* (similar to how a researcher might use a library, or a user might query a search engine). Systems were judged according to their performance on several metrics. The information retrieval field has used several accepted evaluation metrics for quite some time: *precision*–the proportion of all documents a system retrieves that actually are relevant (i.e., those judged by humans as relevant) and *recall*–the proportion of relevant documents that are retrieved. More refined metrics, such as average precision (over a number of queries at a standard level of recall), also are used.

The field has made major progress during the seven TRECs held to date; average precision has doubled from 20% to 40%. Figure 1 shows the details for a typical Information Retrieval (IR) research group at Cornell. The figure shows seven different systems (labeled System '92 to System '98), representing seven different versions of the base Cornell system for those 7 years. The chart shows mean average precision for those seven systems for seven different datasets (TREC–1 to TREC–7). For each dataset we can see that, in general, later systems performed better than earlier ones, as evidenced by the fact that all dataset curves have a performance trend upward over time.

Furthermore, the set of TREC tasks is being refined and expanded beyond routing and ad hoc queries. Over the years new tasks have been added, such as interactive retrieval, filtering, Chinese, Spanish, cross-lingual, high precision, very large collections, speech, and database merging. In each case, participants address a common task with a shared dataset. Common tasks and metrics have made it possible not only to compare the techniques used by different systems but also to compare the evolution of the same system over time (Sparck Jones, 1998b).

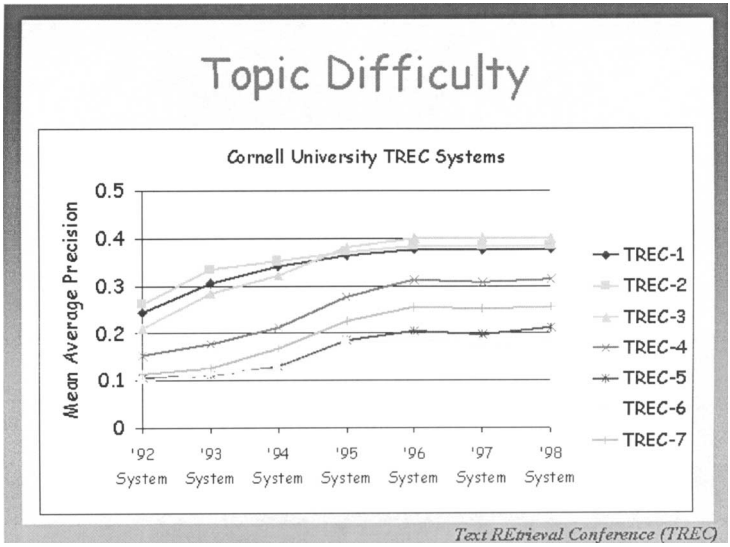
Digital Libraries and Machine Learning

Similar experiments are being carried out in other disciplines. For example, progress has been made in digital libraries by focusing on the core tasks of searching, copyrighting, and cataloging. In machine learning, there are a number of accepted tasks (such as learning classification rules). The University of California Irvine repository provides a common set of learning data that many researchers use, enabling them to compare the performance of their algorithms (Blake, Keogh, & Merz, 1998).

2.2. Lessons From DARPA and TREC

The experience of the information retrieval and speech recognition fields with shared tasks, metrics, and datasets reveals a number of lessons of conse-

Figure 1. **Mean average precision of different Cornell systems for lifetime of TREC.**



quence to the HCI reference task proposal. First, there are a number of positive outcomes:

- They show the essential role of the research community. Researchers defined tasks, produced and shared datasets, and agreed on suitable evaluation metrics. Furthermore, practices within the community were changed to reflect the new situation. Groups applied their systems to common tasks and data, then met to present and analyze their results The yearly bake-off became a key event for the community.
- It is possible to work within a framework of accepted tasks while refining and extending the set of tasks over time. Both TREC and the DARPA workshops have added more tasks over the years as well as increased their difficulty and realism. This is important because it suggests that discovering ideal reference tasks is likely to be an iterative collective process.
- One unexpected outcome of the workshops is that system architectures and algorithms have tended to become more similar. In consequence, it has become possible to carry out independent "black-box" evaluations of different modules. In the case of IR, this common architecture has also become a de facto decomposition of the overall retrieval task.

- A common architecture and shared datasets make it possible for more people to participate. Small research groups do not need to collect large datasets (which can be expensive and time consuming). In addition, they can evaluate their techniques on a subpart of the overall task, which means that they do not need to construct entire large systems to experiment with their ideas.

There are also several more problematic issues arising from the TREC and DARPA workshops:

- These workshops rely heavily on a bake-off model. The bake-off model we have discussed so far is premised on the assumption that research results are embodied in a working system. Furthermore, we have seen that these systems are evaluated according to objective metrics (number of words recognized correctly, average precision for given recall, etc.). When we consider the case of HCI, however, we must ask how well the system bake-off model will work.
- Are there key HCI research results that cannot be implemented, and thus cannot be evaluated, as part of a system? Are there alternatives to the bake-off model? Might we extend the bake-off model to areas of HCI that are not focused on systems (e.g., design, methods, or requirements analysis)? For example, with methods we might ask whether an ethnomethodological analysis yields better data for design than an experiment, and under which conditions are different methods most useful (Gray & Salzman, 1998)? In addition, the bake-off itself is not strictly necessary, although it serves an important social function. We can distinguish different elements of the DARPA–NIST process; for example, one could provide and utilize shared datasets without having annual bake-off meetings to compare performance on them. Obviously, this would decrease the social interaction surrounding the annual meetings, but it would still provide the data to allow direct comparison of systems.
- There are also complex issues concerning interactivity. TREC and DARPA have focused on simple noninteractive tasks. Going from simple tasks (where objective metrics can be easily defined) to more difficult and realistic tasks cannot always be done easily. Making this step may require fundamentally different algorithms and techniques. Both the TREC and DARPA workshops have found the process of moving toward interactive tasks with subjective evaluation criteria difficult and painful, albeit necessary.

- Previous evaluations allowed researchers to test their systems on existing datasets, with no role for people; this allowed the calculation of objective success measures such as word error rate, precision, and recall. Bringing the human element into the evaluation (as users, participants, judges, etc.) produces a more complicated, costly, and subjective process. However, HCI tasks must include people. Thus, to the extent that HCI researchers want to experiment with the bake-off model, they must begin precisely at the point in which researchers in other fields have experienced major problems–where noninteractive tasks with wholly objective criteria were abandoned. Rather than metrics that measure objective system performance, evaluation experiments will be required. This will necessitate the definition of common tasks and metrics so that we can compare the effects of people using different UI techniques to carry out the same task and allowing direct task-based evaluations to be made.
- We previously presented system convergence as a positive feature, but it may also have a negative side. Experience from both speech and IR has shown that groups sometimes take the strategy of imitating the best system from the previous year's bake-off, with additional engineering to improve performance. If this strategy is consistently applied throughout a community, the net effect is to reduce the diversity of approaches being explored. A community eventually might find itself trapped in a "local minimum," finding that its repertoire of techniques does not generalize well when new, more complicated tasks and problems are to be faced. For this reason it is critical that the reference task set is continually modified and made more complex to prevent "overlearning" of specific datasets and tasks.

We do not yet have solutions for these potential issues. Instead, we view these as cautions that must be kept in mind as we experiment with the reference task model.

Criteria for Selecting Reference Tasks

How then do we choose appropriate reference tasks for HCI? Candidate reference tasks are those that are important in everyday practice. A task may be important for different reasons, however; most notably, it may be

- Frequent–A given task may be central to multiple user activities so that addressing it will have general benefits. An example may be processing of asynchronous messages. Given the centrality of communi-

cation for many user activities, improved ways to manage messages will have general benefits.

- Critical–Other tasks may be executed less frequently, but there may be large incentives to execute them correctly. Examples include safety-critical applications such as air traffic control.
- Real–Tasks must also be real, avoiding the danger of being abstracted from actual user practice. These criteria of reality and importance cannot be determined by researchers' intuitions: Significant empirical investigation of user activity must be undertaken to establish which tasks fit these criteria. We have in mind a number of areas that we think are worthy of intense study and are likely to yield reference tasks, including
    - Information browsing, retrieval, and management.
    - Task management.
    - Information sharing.
    - Computer-mediated communication.
    - Document processing.
    - Image processing and management.
    - Financial computation.

In selecting reference tasks, we also must aim for tasks that are unlikely to become obsolete. Although radical inventions are impossible to anticipate, we must keep an eye on the ongoing curve of faster and cheaper computer chips and memory. Tasks that are likely to become unimportant, or be radically transformed, simply through predictable technological progress are not candidate reference tasks.

Our goals in defining a reference task include generating shared requirements, accepted task definitions, descriptive vocabulary, task decomposition, and metrics. Common definitions are critical for researchers to determine how other research is related to their effort. The intended purpose of an interactive artifact needs to be defined with respect to a given task and requirements, with precise metrics for measuring whether its stated purpose is achieved, so that designers and researchers can evaluate the quality of their solutions.

We discuss how we propose to go about defining a reference task, discuss what the definition might look like, and give an example to illustrate this approach. First, however, we think it is worthwhile to discuss potential drawbacks of our approach.

Potential Objections to Our Proposal

What are the limitations of the reference task approach? One potential drawback is that HCI becomes a "clean-up" operation, with its sole aim to un-

derstand and improve existing tasks, techniques, and applications. However, the areas of information retrieval and speech recognition provide an interesting counterargument. Speech recognition technology has become faster and more robust through experimentation on the original set of DARPA-defined tasks. One consequence of these developments is that the technology has begun to be applied successfully to novel problems such as the search of speech and video archives–and TREC has begun to add tasks in these areas (Voorhees & Harman, 1997, 1998). Thus, improvements to a well-known and focused technique have enabled it to be generalized to novel and important problems in a completely different research area.

Another potential objection is that a focus on reference tasks might stifle innovation. However, the history of science and technology indicates that most major inventions required a critical mass of innovators producing multiple versions of a given technology before its successful uptake (Marvin, 1988). By working in a radical invention mode, we precisely fail to achieve the necessary critical mass along with the repeated solution attempts that are necessary to make such breakthroughs. Again, we are not calling for an end to radical invention, just arguing that the scales are tilted too heavily in this direction and that more "normal science" is needed (Kuhn, 1996).

Finally, there is the danger of adopting a faulty paradigm. If our field were to be based on commonly accepted assumptions that are flawed, potential progress would be severely limited. Within cognitive science and artificial intelligence (AI), there has been lively and sometimes bitter debate over foundational assumptions (Dreyfus, 1992; Ford & Pylyshyn, 1995; Harnad, 1990; Searle, 1981). The notion of representation that was taken for granted in symbolic AI has been attacked (Bickhard & Terveen, 1995). More specifically, the notion of "planning," as formalized in the restricted Stanford Research Institute Problem Solver formalism and applied in the artificial "Blocks World," has been criticized (Agre, 1988). Similar arguments have been offered in the speech community. The emphasis on noninteractive tasks with performance measured using the single metric of word error rate has produced predominantly hidden Markov-based techniques that do not generalize well to nonstandard situations or phenomena such as hyperarticulation (Oviatt, Levow, MacEachern, & Kuhn, 1996) or speech in noisy environments (Junqua, 1999).

We do not believe the reference task approach runs this risk. We are not proposing new assumptions, or a new theory–instead, we are simply proposing a somewhat altered methodology in which much more attention is paid to existing tasks. Note that completely radical solutions are consistent with the approach we are proposing; it is just that they need to be made relevant to a reference task and be followed up by systematic analysis. The field needs to devote substantially more effort to producing a rigorous understanding of the core conceptual territory of HCI, even as new radical solutions expand that territory.

A variant of the last argument is that the reference task approach will lead to a focus on the quantifiable, with an accompanying blindness to more subtle issues and considerations. Much important recent HCI work has shown how factors that are not easily quantifiable, such as ethical issues (Nardi, Kuchinsky, Whittaker, Leichner, & Schwarz, 1996) and social relationships among various stakeholders (Grudin, 1988; Orlikowski, 1992), can affect the success of interactive technologies dramatically. It is also clear that from a design perspective that aesthetic issues can have a substantial impact on the success of applications (Laurel, 1990). The reference task approach is, at the very least, neutral with respect to factors such as ethics and aesthetics. Although we have not focused on such issues thus far, to the extent that they are crucial to user performance and satisfaction in a task domain, successful reference task definitions naturally must incorporate them. Many of these issues seem to relate to subjective judgments by users. In our discussion of appropriate metrics, we talk about the need for subjective measures such as user satisfaction. Our hopes are that there are systematic ways that users and groups make decisions about interfaces and that, by defining appropriate methods to elicit this information, we can address this problem.

3. HOW TO DEFINE A REFERENCE TASK

The first question is, What is a task? We adopt the activity theory view that a task is a conscious action subordinate to an object (Kaptelinin, 1996). Each action, or task, is in support of some specific object such as completing a research paper, making a sale, building an airplane, or curing a patient. The object in the most fundamental sense in these cases is the paper, the sale, the airplane, the patient. The tasks are performed to transform the object to a desired state (complete paper, closed sale, functioning airplane, healthy patient).

The same tasks can occur across different objects, so the task of outlining, for example, would be useful for writing a book, preparing legal boilerplate, or specifying a product. In studying reference tasks, it is often useful to know what the object of tasks is so that, for example, appropriate customizations can be offered to users. Although there may be a generic "outlining engine," outlining a product specification could entail special needs to be supported through customizing the basic engine. Keeping the object in mind will bring designs closer to what users really need.

We also need empirical work to determine good domains to investigate candidate reference tasks. Of the broad range of tasks involving computers, we need to identify tasks satisfying our earlier criteria of frequency and criticality. Defining a reference task may begin with a very thorough analysis of existing work in a given area. In the past, it seems as though each individual research effort tended to define its own somewhat different problem, set of re-

quirements, and (post hoc) evaluation metrics. However, by analyzing a broad set of articles that seem to be in the same area, one can attempt to abstract out the common elements, such as

- What are the user requirements in this area? Are they based on solid empirical investigation? Often the answer is no; this means that empirical studies of user activity in this area are necessary.
- Is there a common user task (or set of tasks) that is being addressed?
- What are the components of the tasks? Is a task decomposition given, or can one be abstracted from various articles?
- What range of potential solution techniques are offered? What problems do they solve, and what problems do they leave unsolved? Are there any problems in applying these techniques (e.g., Do they require significant user input, scaling, privacy, or security concerns)?
- How are solution techniques evaluated? Are any general metrics proposed that are useful beyond the scope of the single study in which they were introduced? This last issue is crucial–it is the search for Newman's (1997) "critical parameters" that help to define the purpose of an artifact and measure how well it serves that purpose.

If researchers engage in this process of abstracting from related work in a given area, they may be personally satisfied with the result. However, satisfying others no doubt will be harder–as well as essential. Different researchers may have different perspectives on every aspect of the task. For this reason there are important social processes that need to be introduced. It is important that a representative set of researchers and practitioners who are concerned with a particular area get together to discuss, modify, and approve the reference task definition. We see this process as being something like a standards committee meeting, although much faster and more lightweight. Perhaps it would be a good idea for some number of such groups to meet at CHI each year, for example, as part of the workshops program. Alternatively, the enterprise might be run through a government sponsored agency such as NIST or DARPA, as has been the precedent for speech and IR technologies. After such a group has approved a reference task, its definition needs to be published. Notices could be posted in the *SIGCHI Bulletin* and *Interactions,* with the complete definition appearing on the Web. Even after a reference task definition has been decided, there has to be a means for it to be modified, as researchers and practitioners experiment with it. Again one might use a model similar to the NIST-TREC model in which tasks are discussed and defined at the yearly meeting, with modifications being made at the next meeting, in the light of participant feedback.

Finally, the community must reinforce the important role of the shared knowledge embodied in reference tasks. Educational courses must emphasize the problems the reference task approach confronts, show how tasks are defined, and show the benefits from using this knowledge. The CHI review process could be modified so that reviewers explicitly rate articles with reference to our model.

4. AN EXAMPLE REFERENCE TASK: BROWSING AND RETRIEVAL IN SPEECH ARCHIVES

We now discuss an example reference task: browsing and retrieval in speech archives. The example is intended to illustrate the process by which we might identify reference tasks, how they can be used to evaluate and improve UIs, and the set of issues arising in this endeavor. In doing so, we summarize work reported in a number of our recent research articles (Choi et al., 1998; Nakatani, Whittaker, & Hirschberg, 1998; Whittaker, Choi, Hirschberg, & Nakatani, 1998; Whittaker et al., 1999; Whittaker et al., 1998a; Whittaker, Hirschberg, & Nakatani, 1998b). Obviously, other areas would have served just as well in producing an example reference task; we selected this area simply because of our personal expertise in this domain.

4.1. Selecting and Specifying Reference Tasks in the Domain of Speech Archives

Two criteria we proposed earlier for selecting a reference task were that the task is either frequent or critical. Therefore, what is the evidence that accessing speech data is an important user task? First, conversational speech has been shown to be both frequent and central to the execution of many everyday workplace tasks (Chapanis, 1975; Kraut, Fish, Root, & Chalfonte, 1993; Whittaker, Frohlich, & Daly-Jones, 1994). Second, voice messaging is a pervasive technology in the workplace and at home, with both voicemail and answering machines requiring access to stored speech data. In the United States alone, there are over 63 million domestic and workplace voicemail users. Third, new areas of speech archiving are emerging: Television and radio programs are becoming available online, news and sports sites are including audio interviews, and public records such as Congressional debates are being made available. Together these observations indicate that searching and browsing speech data meet the criteria of being frequent, general, and real. Furthermore, we argue that the tasks we identify in speech retrieval may generalize to retrieval of textual data, making it possible to use them more widely.

However, identifying the area of speech retrieval does not provide us with information about the specific tasks that users carry out when they are access-

ing speech archives. To gather more detailed information about this, we collected several different types of data concerning people's processing of voicemail data. We chose to examine voicemail access rather than news data given that voicemail is currently the most pervasive application requiring access and retrieval of speech data. We collected qualitative and quantitative data to identify users' key tasks for processing voicemail for a typical voicemail system, Audix™, including (a) server logs from 782 active users, (b) surveys from 133 high-volume users (receiving more than 10 messages per day), and (c) interviews with 15 high-volume users. We also carried out laboratory tests to confirm our findings on 14 other users.

We found evidence for three core tasks in accessing voicemail archives: (a) *search,* (b) *information extraction,* and (c) *message summarization.* Search is used for *prioritizing* incoming new messages and for locating valuable saved messages. Prioritization is critical for users who must identify urgent incoming messages while accessing the mailbox under time constraints (e.g. during a meeting break). These users have to rapidly determine which new messages require immediate attention. Search also occurs when users access old archived messages to locate those containing valuable information. Our working definition of search was as follows: Given a set of messages, identify a (small) subset of those messages having various attributes with certain values (e.g., being from a particular person or being about a particular topic). Information extraction involves accessing information from within messages. When a relevant message is identified, users have to extract critical information from it. This is often a laborious process involving repeatedly listening to the same message for verbatim facts such as caller's name and phone number. Our definition of information extraction is as follows: Given a message or set of messages, identify particular classes of information from within the message. In terms of attribute value representations, this means the following: Given a set of messages and a set of attributes, identify the values associated with those attributes. A final task at the message level is summarization: To avoid repeatedly replaying messages, most users attempt to summarize their contents, usually by taking handwritten notes consisting of a sentence or two describing the main point of the message. Our definition of summarization is that it involves selection of a subset of information from within the document that best captures the meaning of the entire document. For more formal definitions of summarization, we refer the reader to Sparck Jones (1998a).

It is important to note that these three tasks were generated by analysis of voicemail user data. Despite the fact that they were derived from speech data, each task has an analogue in the independently generated TREC set of tasks for retrieval of textual data. The fact that these three tasks may be common to searching both speech and text is encouraging for the reference task approach.

It argues that there may be general tasks for search that are independent of data type.

4.2. Defining Metrics

In addition to identifying tasks, our data suggested several possible metrics that might be used to gauge task success. In the interviews it seemed that people oriented to three different aspects of system usage when trying to execute their tasks. First, it was important to users whether they completed their tasks correctly and accurately. People would repeatedly access the system until they felt that they had correctly extracted critical information such as a caller name or phone number from a message, or until they had found the message they were searching for. We call this criterion *task success.* However, people were also focused on issues of efficiency: A major complaint by almost all users was that executing the three core tasks took far too long, requiring far too many button presses and menu choices. This led us to conclude that another useful evaluation criterion involved the time to complete a given task (for a discussion of the utility of time as a critical parameter, see Burkhart, Hemphill, & Jones, 1994; Newman, 1997). Finally, users made comments about the subjective or experiential quality of the interaction, leading us to a criterion of subjective evaluation.

4.3. Task-Oriented Evaluation of a Speech Browsing System

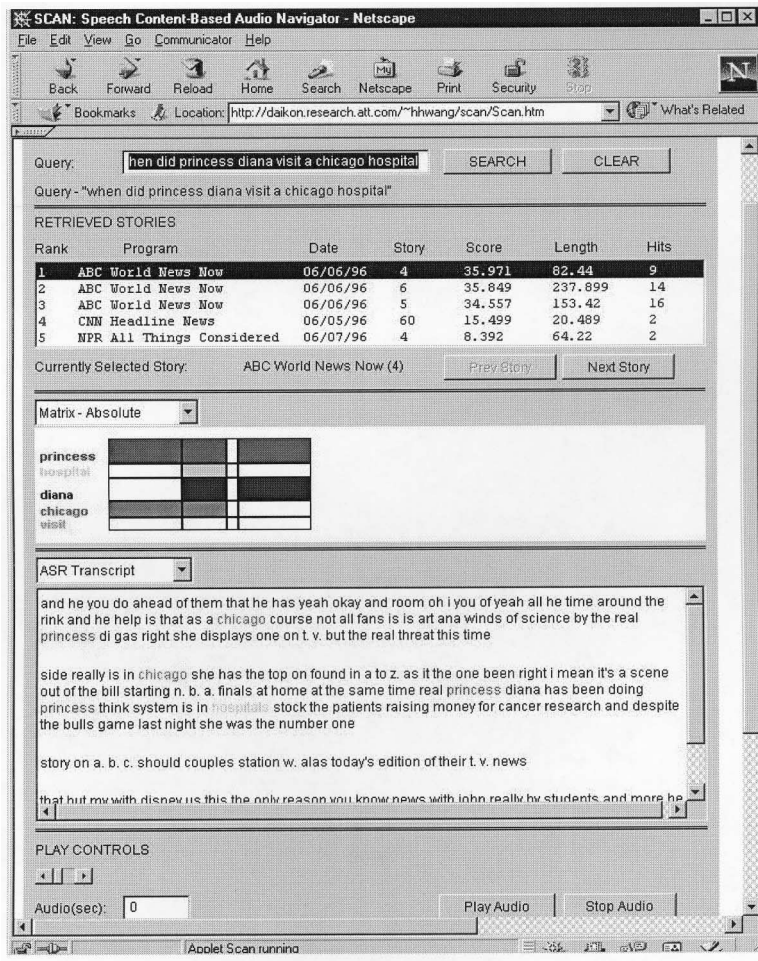
Having identified core tasks and success metrics, we attempted to apply these to a real system that allows users to search and browse recorded news broadcasts.[3] The system works in the following way: It applies an automatic speech recognition system to digitized recorded broadcasts, indexes the resulting errorful[4] transcriptions of the speech for information retrieval, and provides a UI to support search and browsing (for a full architectural description, see Choi et al., 1998). Figure 2 shows the UI. The details of the UI are described elsewhere (Whittaker et al., 1999; Whittaker et al., 1998b), and the elements of the UI support a new paradigm for speech retrieval interfaces: "What you see is (almost) what you hear" (WYSIAWYH).

To evaluate two different versions of the UI (and hence two different UI techniques), we conducted laboratory experiments in which users were given three tasks: search, summarization, and information extraction, correspond-

3. We are also currently carrying out similar experiments with voicemail data (Whittaker, Davis, Hirschberg, & Muller, 2000).

4. The errors arise because Automatic Speech Recognition (ASR) performance for this type of data is imperfect: with about 70% of words being correctly recognized.

Figure 2. **"What you see is (almost) what you hear" browser providing overview and transcript for browsing. Netscape Communicator browser window © 1999 Netscape Communications Corporation. Used with permission. Netscape Communications has not authorized, sponsored, endorsed, or approved this publication and is not responsible for its content.**



ing to the three reference tasks we had identified. For the search task, users were asked to find the most relevant speech document addressing a given issue. In the summarisation task, we asked users to produce a six- to eight-sentence summary of a single speech document (where documents were about 5 min in length). Finally, for information extraction, we asked people to find a fact in a given speech document (e.g., What were the names of the actors who

starred in the Broadway musical *Maggie Flynn*?). We used three evaluation metrics (viz., *task success, time to solution,* and *perceived utility*) of the UI. To determine task success, we had independent judges rank documents for relevance, rate summaries, and determine the correctness of factual answers.

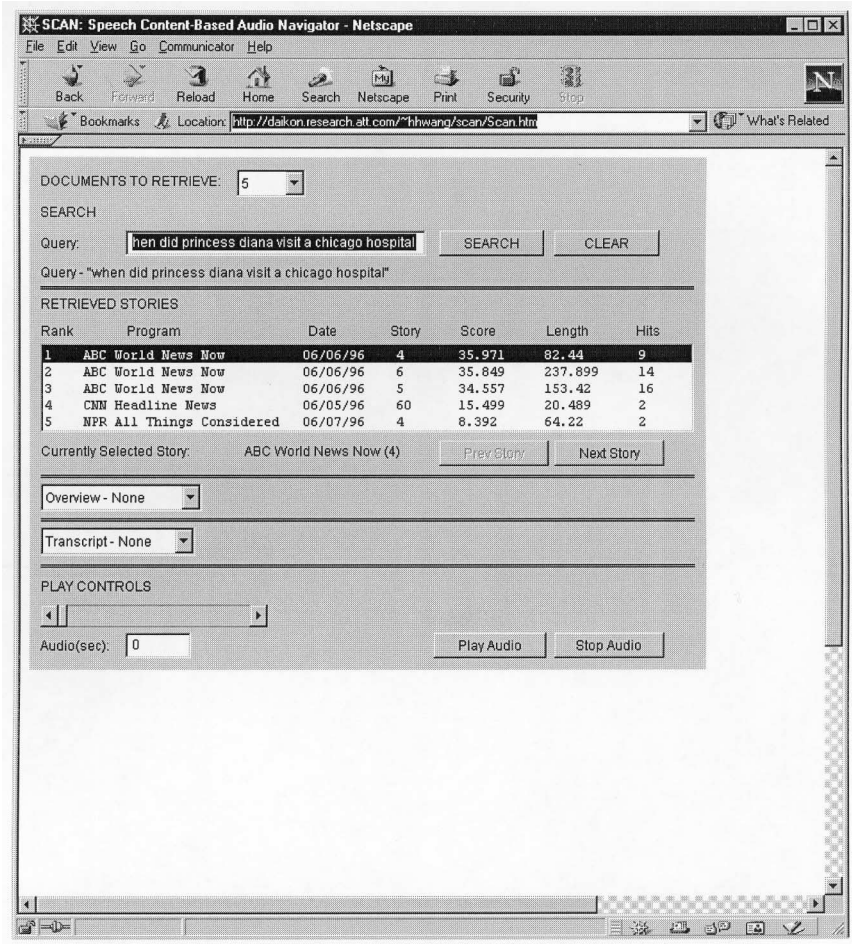
We initially used the method to compare two different versions of the UI. The main problem with browsing speech is that of random access to relevant materials. When browsing text, people are able to visually scan exploiting structural cues (formatting, paragraphs, headers) and look for key words, enabling them to focus on relevant document regions. One version of the UI attempted to emulate this by providing a visual analogue to the underlying speech allowing people to visually scan as they would with text (see Figure 2). This WYSIAWYH UI provided users with graphical information about how the terms in their query were distributed in a given document, allowing them to "zoom in" on regions containing large numbers of query terms and ignore the parts of the document that were not relevant to their query. It also provided information about the content of each speech document by presenting the errorful transcript of each story (including highlighted query terms) allowing users to visually scan through stories to identify relevant regions for playing. We compared this with a simpler version of the UI without these browsing features. It allowed users to search for speech documents but provided no browsing support: Users selected audio to play using tape-recorder-type controls (see Figure 3). We found that for all metrics the more complex UI was better for search and information extraction tasks, but we observed no differences between UI versions for the summary task. More details are supplied in Whittaker et al. (1999).

We have since conducted further studies using identical metrics and tasks to evaluate different versions of the UI, and also the effects of systematically varying the quality of automatic speech recognition on browsing and search.

4.4. General Issues Arising From Reference Task-Based Evaluation

Although our task-based approach has generally been successful, a number of issues have arisen in applying the method. One major issue concerns our choice of metrics and the importance we associate with each. We have chosen to employ multiple evaluation metrics, an approach that differs from methods that advocate the use of a single evaluation metric, such as the PARADISE (Walker et al., 1998) method for evaluating interactive spoken language systems. Our decision was influenced by several factors. The issue of appropriate evaluation metrics has generated much previous debate, and it is clear that the selection of metrics is a highly complex process (Gray et al., 1993; Gray & Salzman, 1998; Roberts & Moran, 1983; Walker et al., 1998). Prior evaluation

Figure 3. **Basic browser providing play controls for browsing. Netscape Communicator browser window © 1999 Netscape Communications Corporation. Used with permission. Netscape Communications has not authorized, sponsored, endorsed, or approved this publication and is not responsible for its content.**



work, for example, has shown inconsistencies between *objective measures* (such as time to solution and task success) and *subjective measures* (such as user satisfaction) for people doing the same task using the same system (Sellen, 1995; Whittaker, Geelhoed, & Robinson, 1993). This inconsistency means that it may not be possible to have one metric "stand in" for another metric, which can be possible if they are highly correlated. Other evaluation work has made

strong claims for the use of the single metric of *user satisfaction* in evaluating system success (Walker et al., 1998), based on the argument that persistent long-term system use is motivated by user's perception of the system's value, rather than externally calculated measures.[5] Even acknowledging the persuasiveness of this argument, there are still outstanding questions as to how exactly we define and measure user satisfaction. Our (conservative) view is that multiple objective and subjective metrics should be used to measure system success. We regard it as a research question as to the exact relation between these measures and whether one metric turns out to be more useful and predictive than others. We also need more work addressing how user satisfaction might be defined and measured.

A second issue concerns reference task selection. One of our chosen tasks, summarization, was relatively insensitive to different UI techniques. Although it was clear from our user data that summarization was a critical task for users, it has not proved to be a useful way to distinguish between different UIs for any of our metrics. Does this mean that summarization is a poor candidate for a reference task? Closer examination of our data suggests possible reasons for our failure to find effects. Overall performance on the summarization task was low. It may therefore be the case that none of our current UI techniques helped with summarization but that better UI techniques would improve performance and produce observed differences on this task. Another possibility is that our definition of the summary task is underspecified and the task was not well defined for users (Sparck Jones, 1998a). Our experience with summarization has an important implication for the reference task approach. It is not enough to select important tasks by careful analysis of user data; these tasks must be well operationalized for evaluation purposes. Operationalization itself may be a complex undertaking to achieve plausible instantiations of tasks in experimental settings.

Another problem concerns the relation between requirements gathering and reference task selection. Most requirements gathering takes place in the context of specific applications. In our case, we gathered information about speech retrieval by investigating voicemail users because voicemail is a pervasive speech archive. However, the primary function of voicemail is as an asynchronous communications application rather than a speech archive. One decision we had to make when selecting reference tasks was whether the ob-

---

5. This is an oversimplification of the Walker, Litman, Kamm, and Abella (1998) position. They argued that multiple factors contribute to system success (e.g., task completion, time to solution, speech recognizer accuracy, use of help prompts), but in modeling the contribution of these factors, their regression analyses treat user satisfaction as the dependent variable. In other words, they view user satisfaction as the critical metric, and they address the question of how these other factors affect it.

served tasks were relevant to speech retrieval or whether they arose from the fact that voicemail is an asynchronous communications application. In our requirements gathering we actually identified two further tasks–status tracking and archive management–that we excluded from the speech retrieval reference task set because they did not directly concern retrieval. Of course, if we were trying to identify reference tasks for managing asynchronous communications (e.g., for e-mail and voicemail applications), then such tasks would be highly relevant.

We also experienced the problem of task granularity. In processing voicemail, users carry out activities that are analyzable at multiple levels of granularity. At the highest level we might describe "processing voicemail" as an activity that users engage in. At the opposite end of the spectrum are low-level acts such as "press Button 3" (e.g., to delete a message). Neither characterization would have been useful as a reference task. The process voicemail characterization is too general and includes tasks that are not directly relevant to speech retrieval (namely status tracking and archive management). In contrast, the "press Button 3" characterization is too specific to the details of a particular implementation. In identifying our three reference tasks we were forced to make a decision about the level of abstraction of the target tasks, and the criteria we used to do this were intuitive. A critical technical issue for the research program concerns specification of the ideal granularity of reference tasks.

We should also be concerned about task specificity. Our results showed that performance was not identical for search, summarization, and information extraction tasks. It may be that we discover that different UI techniques are successful for different reference tasks. Such a conclusion would indeed be consistent with observations about task-specific interfaces (Nardi, 1993) as well as with current theories of situated cognition (Lave, 1988; Suchman, 1987). Our findings may be highly task specific, which again highlights the importance of careful task selection. Our reference tasks must be chosen so they are critical to our users' everyday computing activities. Careful task selection ensures that we still generate important and useful data to help improve system design for important user problems, even if that design does not generalize to all user tasks.

Of course, our hope is that our approach leads to the discovery of general techniques and principles for UI design, but if not, then at least we have data about tasks that are relevant and important to our users. In the worst case, it might mean that the field of HCI splinters into multiple task-based areas of research, but at least those areas would be informed by well-researched user needs about critical user problems, with well-defined evaluation metrics. Furthermore, a number of factors would still unite such task-based communities, including methodologies such as user-centered and participatory design,

modeling techniques such as GOMS, broad frameworks such as activity theory, and computational tools such as rapid prototyping environments and interface builders. As far as application design and development is concerned, having task-specific information may correspond well with common practice, as most application development takes place in a highly task specific context.

Another issue concerns user population. Although we have made every attempt to ensure the representativeness of the people participating in our experiments, it may turn out that particular sets of users (e.g., elderly people or children) act very differently with the technology. User population is another factor that needs to be included in the reference task analysis. Our reference task statements should therefore be of the following form: For user Population X and Task Y, Technique A improves performance on Metric Z.

Another issue concerns inherent limitations of task-based evaluation. People participating in experimental studies are asked to perform prespecified tasks over a short period of time. As such, the approach does not allow us to detect ad hoc or creative usage of the UI or how usage strategies evolve over extended periods of time. These phenomena can only be observed in field trials. Of course, field trials also have their drawbacks. Field trial users select their own tasks, making it impossible to draw direct comparisons between different techniques or systems because different users are executing different tasks. We therefore advocate that extended usage in field trials should be used as a method to complement our task-based evaluation. It is also important that the entire evaluation process is iterative and combine the results of experimental and field-based methods. Field trials may show that critical user tasks have been neglected or that technologies may be developed and used in novel ways. The results of the field trials should therefore be used to modify the next set of task-based evaluations and the technology that is used in those evaluations.

Finally, we revisit the issue of what is new about the reference task approach in the light of the speech browsing and retrieval example. After all, is the process we just described good, but standard, HCI practice? To a large extent, the answer is yes. It is standard best practice in HCI to interview users to understand their needs, develop a system to meet these needs, and evaluate the system with users to see if it does in fact meet their needs.[6] Recall though, that the reference task agenda involves both technical and social aspects. We make a major divergence from standard practice on the technical front in our recommendation (following Newman, 1997) that we use general evaluation metrics, along with the need to derive these metrics for important tasks. However, the more important implications of our worked example are social. We found

6. There may be major differences between *ideal* and *actual* descriptions of the process of HCI. Although the ideal is to follow the three steps we describe, few actual studies seem to execute all three.

there was no accepted body of work we could draw on for task definitions or user requirements. There were no accepted metrics. And, in moving toward developing this knowledge, there are no accepted community mechanisms for refining and disseminating the knowledge iteratively. Developing such social mechanisms is the major activity we must undertake to put the reference task approach into practice.

5. CONCLUSIONS

We identify a problem with the process of research in the HCI community, namely that the emphasis on radical innovation is preventing the building of a common research focus. Without such a focus, people cannot build on the work of others, and it is not possible to compare UI techniques to improve them. The lack of common focus also makes it difficult to generate the necessary critical mass required for theory development. In consequence, we cannot give informed design advice to builders of new systems. We have proposed that the HCI community try to achieve such a focus around the notion of reference tasks. We have offered general arguments for the advantages and disadvantages of this approach and described an example reference task for searching and browsing speech archives. We point to a number of outstanding issues that arose from our experience of reference task-based evaluation—choice of metrics, selection, and operationalization of tasks, task-specificity of results, user variability, and the need for complementary field trials. We also point to the absence of methods for distributing and sharing data and results within the field.

We have also outlined what steps need to be taken to execute the reference task research agenda. We make recommendations at two levels: technical and social. The technical research that needs to be carried out to successfully implement the reference task proposal involves identification of important user tasks by systematic requirements gathering, definition and operationalization of reference tasks and evaluation metrics, and execution of task-based evaluation along with judicious use of field trials. The major technical hurdles are likely to be (a) reaching agreement on task definitions; (b) developing general templates for describing reference tasks, setting out the criteria they must satisfy, and including their level of granularity; (c) defining appropriate metrics; and (d) designing appropriate task-based evaluation techniques. Perhaps more important, we have also suggested changes in community practice that HCI must adopt to make the reference tasks idea work. We must create influential forums for discussion of common tasks and methods by which people can compare systems and techniques. The major obstacle here promises to be to define a process that will allow researchers to reach agreement on task definitions and provide methods to disseminate these definitions in a way that

they come to be broadly used by the HCI community. Only by doing this can the notion of reference tasks be included into the process of research and development and the field achieve the focus it desperately needs.

NOTES

Acknowledgments. Thanks to Julia Hirschberg, Candy Kamm, Fernando Pereira, and Marilyn Walker, along with the attendees at *HCIC 1999* who gave us useful suggestions, feedback, and comments about these ideas.

Authors' Present Addresses. Steve Whittaker, AT&T Labs-Research, 180 Park Avenue, Florham Park, NJ 07932. E-mail: stevew@research.att.com. Loren Terveen, AT&T Labs-Research, 180 Park Avenuem, Florham Park, NJ 07932. E-mail: terveen@research.att.com. Bonnie Nardi, AT&T Labs West, 75 Willow Road, Menlo Park, CA 94025. E-mail: nardi@research.att.com.

HCI Editorial Record. First manuscript received March 17, 1999. Revision received November 11, 1999. Accepted by Clayton Lewis, Wendy Kellogg, and Peter Polson. Final manuscript received May 2000. – *Editor*

REFERENCES

Agre, P. (1988) *The dynamic structure of everyday life.* Unpublished doctoral dissertation, MIT AI Laboratory, AI Department, Cambridge, MA.

Amento, B., Hill, W., Terveen, L., Hix, D., & Ju, P. (1999). An empirical evaluation of user interfaces for topic management of Web sites. *Proceedings of the CHI'99 Conference on Computer–Human Interaction,* 552–559. New York: ACM.

Baecker, R. (1987). Towards an effective characterization of graphical interaction. In R. Baecker & W. Buxton (Eds.), *Readings in human computer interaction* (pp. 471–481). San Francisco, CA: Kaufmann.

Barreau, D., & Nardi, B. (1995). Finding and reminding: Organization of information from the desktop. *SIGCHI Bulletin, 27,* 39–45.

Bickhard, M. H., & Terveen, L. G. (1995). *Foundational issues in artificial intelligence and cognitive science: Impasse and solution.* New York: Elsevier.

Blake, C., Keogh, E., & Merz, C. J. (1998). *UCI repository of machine learning databases.* Irvine: University of California Press, Department of Information and Computer Science. Retrieved October 31, 2000 from the World Wide Web: http://www.ics.uci.edu/~mlearn/MLRepository.html

Brennan, S. (1990). Conversation as direct manipulation: An iconoclastic view. In B. Laurel (Ed.), *The art of human computer interface design* (pp. 393–404). Reading. MA: Addison-Wesley.

Burkhart, B., Hemphill, D., & Jones, S. (1994). The value of a baseline in determining design success. *Proceedings of the CHI'94 Conference on Computer–Human Interaction,* 386–391. New York: ACM.

Card, S., Moran, T., & Newell, A. (1983). *The psychology of human–computer interaction.* Hillsdale, NJ: Lawrence Erlbaum Associates, Inc.

Carroll, J., & Campbell, R. (1986). Softening up hard science. *Human–Computer Interaction, 2,* 227–249.

Chapanis, A. (1975). Interactive human communication. *Scientific American, 232,* 36–42.

Choi, J., Hindle, D., Hirschberg, J., Magrin-Chagnolleau, I., Nakatani, C. H., Pereira, F., Singhal, A., & Whittaker, S. (1998). SCAN–speech content audio navigator: A systems overview. *Proceedings of the International Conference on Spoken Language Processing,* 604–608. Piscataway, NJ: IEEE.

Dreyfus, H. L. (1992). *What computers still can't do*. Cambridge, MA: MIT Press.

Fertig, S., Freeman, E., & Gelertner, D. (1996). "Finding and reminding" reconsidered. *SIGCHI Bulletin, 28*, 66–69.

Ford, K. M., & Pylyshyn, Z. (1995). *The robot's dilemma revisited: The frame problem in artificial intelligence*. Norwood, NJ: Ablex.

Foundyller, C. (1984). *Cad/CAM, CAE: The contemporary technology*. Cambridge, MA: Daratech Associations.

Gifford, D., Jouvelot, P., Sheldon, M., & O'Toole, J. (1991). Semantic file systems. *Proceedings of 13th ACM Symposium on Operating System Principles,* 341–349. New York: ACM.

Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. *Communications of the ACM, 35,* 51–60.

Gray, W. D., John, B. E., & Atwood, M. E. (1993). Project Ernestine: Validating a GOMS analysis for predicting and explaining real-world task performance. *Human–Computer Interaction, 8,* 237–309.

Gray, W., & Salzman, M. (1998). Damaged merchandise? A review of experiments that compare usability evaluation methods. *Human–Computer Interaction, 13,* 203–262.

Grudin, J. (1988). Why CSCW applications fail: Problems in the design and evaluation of organizational interfaces. *Proceedings of the CSCW'88 Conference on Computer Supported Cooperative Work,* 85–93. New York: ACM.

Harnad, S. (1990). The symbol grounding problem. *Physica D, 42,* 335–346.

Hill, W. C., Stead, L., Rosenstein, M., & Furnas, G. (1995). Recommending and evaluating choices in a virtual community of use. *Proceedings of the CHI'95 Conference on Computer–Human Interaction,* 194–201. New York: ACM.

Isaacs, E., & Tang, J. (1996). Technology transfer: So much research so few good products. *Communications of the ACM, 39,* 22–25.

Junqua, J.-C. (1999). The Lombard effect: A reflex to better communicate with others in noise. *Proceedings of the International Conference on Acoustics Speech and Signal Processing,* 2083–2086. Piscataway, NJ: IEEE.

Kaptelinin, V. (1996). Activity theory: Implications for human–computer interaction. In B. Nardi (Ed.), *Context and consciousness: Activity theory and human–computer interaction* (pp. 103–116). Cambridge, MA: MIT Press.

Kraut, R., Fish, R., Root, B., & Chalfonte, B. (1993). Informal communication in organizations. In R. Baecker (Ed.), *Groupware and computer supported co-operative work* (pp. 287–314). San Francisco, CA: Kaufmann.

Kuhn, T. S. (1996). *The structure of scientific revolutions*. Chicago: University of Chicago Press.

Landauer, T. (1995). Let's get real. In R. Baecker, J. Grudin, W. Buxton, & S. Greenberg (Eds.), *Human computer interaction: Towards the year 2000* (pp. 659–666). San Francisco, CA: Kaufmann.

Laurel, B. (1990). *The art of human computer interface design.* Reading, MA: Addison-Wesley.

Lave, J. (1988). *Cognition in practice.* New York: Cambridge University Press.

Marcus, M. (1992). *Proceedings of speech and natural language workshop.* San Francisco, CA: Kaufmann.

Marvin, C. (1988). *When old technologies were new.* New York: Oxford University Press.

Nakatani, C. H., Whittaker, S., & Hirschberg, J. (1998). Now you hear it now you don't: Empirical studies of audio browsing behavior. *Proceedings of the International Conference on Spoken Language Processing,* 1003–1007. Piscataway, NJ: IEEE.

Nardi, B. (1993). *A small matter of programming.* Cambridge, MA: MIT Press.

Nardi, B., Kuchinsky, A., Whittaker, S., Leichner, R., & Schwarz, H. (1996). Video-as-data: Technical and social aspects of a collaborative multimedia application. *Computer Supported Cooperative Work, 4,* 73–100.

Nardi, B., Miller, J., & Wright, D. (1998). Collaborative, programmable intelligent agents. *Communications of the ACM, 41,* 96–104.

Newell, A., & Card, S. (1985). The prospects for psychological science in human computer interaction. *Human–Computer Interaction, 1,* 209–242.

Newman, W. (1994). A preliminary analysis of the products of HCI research using pro forma abstracts. *Proceedings of the CHI'94 Conference on Computer–Human Interaction,* 278–284. New York: ACM.

Newman, W. (1997). Better or just different? On the benefits of designing interactive systems in terms of critical parameters. *Proceedings of DIS'97 Designing Interactive Systems,* 239–246. New York: ACM.

Olson, J., & Olson, G. (1990). The growth of cognitive modeling in human computer interaction since GOMS. *Human–Computer Interaction, 5,* 221–265.

Orlikowski, W. (1992). Learning from notes: Organizational issues in groupware implementation. *Proceedings of the CSCW'92 Conference on Computer Supported Cooperative Work,* 362–369. New York: ACM.

Oviatt, S. L., Levow, G., MacEachern, M., & Kuhn, K. (1996). Modeling hyperarticulate speech during human–computer error resolution. *Proceedings of the International Conference on Spoken Language Processing,* 801–804. Piscataway, NJ: IEEE.

Price, P. (1991). *Proceedings of speech and natural language workshop.* San Francisco, CA: Kaufmann.

Rao, R., Card, S., Johnson, W., Klotz, L., & Trigg, R. (1994). Protofoil: Storing and finding the information worker's documents in an electronic filing cabinet. *Proceedings of the CHI'94 Conference on Computer–Human Interaction,* 180–185. New York: ACM.

Resnick, P., Iacovou, N., Suchak, M., Bergstrom, P., & Riedl, J. (1994). GroupLens: An open architecture for collaborative filtering of netnews. *Proceedings of CSCW' 94 Conference on Computer Supported Cooperative Work,* 175–186. New York: ACM.

Resnick, P., & Varian, H. R. (Eds.). (1997). Special issue on recommender systems [Special issue]. *Communications of the ACM, 40*(3), 56–58.

Roberts, T. L., & Moran, T. P. (1983). The evaluation of text editors: Methodology and empirical results. *Communications of the ACM, 26,* 265–283.

Rudisill, M., Lewis, C. L., Polson, P. G., & McKay, T. D. (1996). *Human–computer interface design: Success stories, emerging methods and real-world context.* San Francisco, CA: Kaufmann.

Searle, J. R. (1981). Minds, brains, and programs. In J. Haugeland (Ed.), *Mind design* (pp. 282–306). Cambridge, MA: MIT Press.

Sellen, A. (1995). Remote conversations: The effects of mediating talk with technology. *Human–Computer Interaction, 10,* 401–444.

Shardanand, U., & Maes, P. (1995). Social information filtering: Algorithms for automating "word of mouth." *Proceedings of the CHI'95 Conference on Computer–Human Interaction,* 210–217. New York: ACM.

Shneiderman, B. (1982). The future of interactive systems and the emergence of direct manipulation. *Behavior and Information Technology, 1,* 237–256.

Smith, D., Irby, C., Kimball, R., Verplank, W., & Harslem, E. (1982). Designing the star user interface. *Byte, 7,* 242–282.

Sparck Jones, K. (1998a). Automatically summarising: Factors and directions. In I. Mani & M. Maybury (Eds.), *Advances in autmomatic text summarization* (pp. 341–376). Cambridge, MA: MIT Press.

Sparck Jones, K. (1998b). Summary performance comparisons TREC2, TREC3, TREC4, TREC5, TREC6. *Proceedings of the Sixth Text Retrieval Conference (TREC-7),* B1–B8. Washington, DC: NIST Special Publications.

Stern, R. (1990). *Proceedings of speech and natural language workshop.* San Francisco, CA: Kaufmann.

Suchman, L. (1987). *Plans and situated actions.* Cambridge, England: Cambridge University Press.

Sutherland, I. (1963, January). *Sketchpad: A man–machine graphical communication system* (MIT Lincoln Library Technical Report #296).

Voorhees, E. M., & Harman, D. K. (1997). Overview of the sixth text retrieval conference (TREC-6). *Proceedings of the Sixth Text Retrieval Conference (TREC-6),* 1–24. Washington, DC: NIST Special Publications.

Voorhees, E. M., & Harman, D. K. (1998). Overview of the seventh text retrieval conference (TREC-7). *Proceedings of the Seventh Text Retrieval Conference (TREC-7),* 1–24. Washington, DC: NIST Special Publications.

Walker, M., Litman, D., Kamm, C., & Abella, A. (1998). Evaluating spoken dialogue agents with PARADISE: Two case studies. *Computer Speech and Language, 12,* 3.

Wayne, C. (1989). *Proceedings of speech and natural language workshop.* San Francisco, CA: Kaufmann.

Whittaker, S., Choi, J., Hirschberg, J., & Nakatani, C. (1998). What you see is almost what you get: Design principles for user interfaces for speech archives. *Proceedings of the International Conference on Speech and Language Processing,* 1009–1013. Piscataway, NJ: IEEE.

Whittaker, S., Davis, R., Hirschberg, J., & Muller, U. (2000). Jotmail: A voicemail interface that enables you to see what was said. *Proceedings of the CHI'2000 Human Factors in Computing Systems,* 89–96. New York: ACM.

Whittaker, S., Frohlich, D. M., & Daly-Jones, O. (1994). Informal workplace communication: What is it like and how might we support it? *Proceedings of the CHI'94 Human Factors in Computing Systems,* 130–137. New York: ACM.

Whittaker, S., Geelhoed, E., & Robinson, E. (1993). Shared workspaces: How do they work and when are they useful? *International Journal of Man–Machine Studies, 39,* 813–842.

Whittaker, S., Hirschberg, J., Choi, J., Hindle, D., Pereira, F., & Singhal, A. (1999). SCAN: Designing and evaluating user interfaces to support retrieval from speech archives. *Proceedings of the SIGIR'99 Conference on Research and Development in Information Retrieval,* 26–33. New York: ACM.

Whittaker, S., Hirschberg, J., & Nakatani, C.H. (1998a). All talk and all action: Strategies for managing voicemail messages. *Proceedings of the CHI'98 Conference on Computer–Human Interaction,* 249–250. New York: ACM.

Whittaker, S., Hirschberg, J., & Nakatani, C. H. (1998b). What you see is almost what you hear: Design principles for user interfaces for accessing speech archives. *Proceedings of the International Conference on Spoken Language Processing,* 2031–2036. Piscataway, NJ: IEEE.

Whittaker, S., & Sidner, C. (1986). Email overload: Exploring personal information management of email. *Proceedings of CHI'96 Conference on Computer–Human Interaction,* 276–283. New York: ACM.