# Discourse analysis techniques for modeling computer-mediated collaboration

**Alexander Feinman and Richard Alterman**
*Brandeis University*

May 11, 2005

**Alexander Feinman** is a cognitive scientist with interests in collaboration and groupware analysis; he is a Ph.D. candidate at Brandeis University. **Richard Alterman** is a cognitive scientist with interests in cognitive modeling, planning and activity, and text and discourse; he is a Professor at Brandeis University.

**Corresponding author:**

Alexander Feinman
Computer Science Department
Brandeis University MS018
415 South Street
Waltham, MA 02454 USA
*afeinman@cs.brandeis.edu*

DRAFT COPY

(To be submitted to Journal of HCI)

## Abstract

Online collaboration has become very common. Remotely-taught educational courses, collaboration with work divisions in remote locations, and coordinating military personnel distributed across an information-driven battlefield all require design work to construct systems that enable participants to effectively coordinate remotely. Understanding the complex interaction of online participants in a joint activity is crucial to designing effective software tools to support their task. By examining the structure of participant interaction, we can reveal the work participants must do to exchange information, and provide data for redesigning the system to support this work.

We have created two methods for systematically analyzing online interaction. One examines coordination issues at a social level by identifying recurring talk about coordination and secondary structure participants create in the discourse to handle these complexities. The other method tracks the references participants make in the discourse to reveal patterns of information use; these insights can inform design of appropriate representations for information. In this paper, we use these methods to examine data from our groupware test bed, VesselWorld, and a variety of other groupware systems, and with them explain the successes and shortcomings of introducing new representations. We also present data from a semester-long experiment in which a class of students applied the methods to successfully justify redesigns of the groupware systems they designed and built.

# Contents

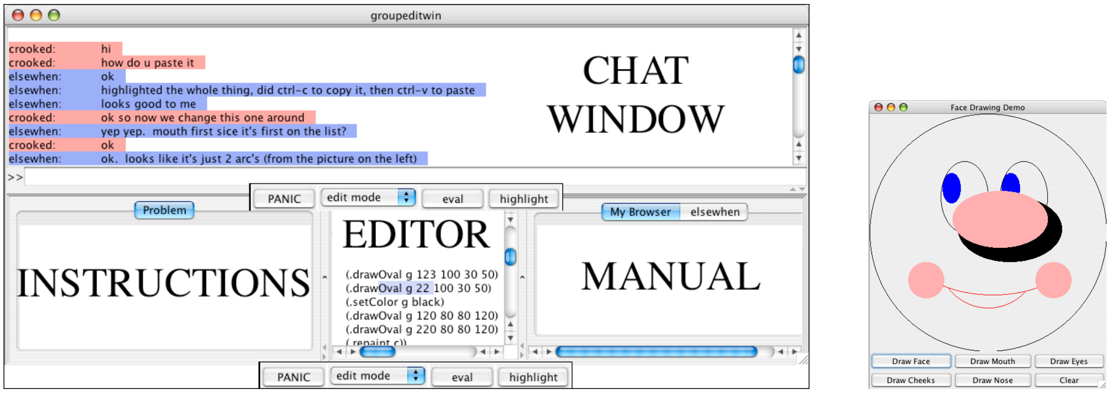
# 1 Introduction

Computer-mediated collaboration has become ubiquitous. Remotely-taught educational courses, collaboration with work divisions in remote locations, and coordinating military personnel distributed across an information-driven battlefield all require extensive design work in order to construct computer-mediated environments that enable participants to coordinate remote activity in an effective fashion. Staying coordinated in such same-time / different-place (Ellis, Gibbs, and Rein 1991) interactions can be difficult. Because participants can be in different locations and have access to different physical environments, the methods they use for interacting differ from those participants would use in a face-to-face interaction. Procedures for referring to, pointing at, modifying, and reviewing objects, as well as gauging the focus, intent, and emotional state of other participants must necessarily be altered when participants interact via a mediating computer system.

However, despite the best efforts of designers, groupware applications often end up interfering with or fundamentally altering the very work they are designed to support (Foster and Stefik 1986; Olson et al. 1992). It is crucial, but difficult, to provide a system which match the needs of the participants. As participants perform a joint activity, a work practice emerges which can be hard to foresee. Ideally, a software system would be built to match the emergent practice of a community of users. One method is to take the same ethnographic techniques used to examine situated activity of participants in face-to-face activity and apply them to online collaboration to produce a representation system which matches the observed work practice.

The major contribution of this paper is the application of discourse analysis techniques to online discourse as tools for guiding redesign. We adapted concepts from discourse analysis to create two methods that investigate the interaction at a conversational level. The first method looks at the recurring features in the dialogue and uses them to highlight problem areas in the interaction. The second method uses referential structure to discover how participants handle information; this in turn informs design of new representations. These methods organize ethnographic observations and generate concrete conclusions about the emergent work practice of a system: what sorts of information are exchanged, what areas of the interaction are problematic, and where the system needs to be modified to reduce coordination effort. These conclusions can lead to redesign directly, or can provide evidence for hypotheses generated by more theoretical approaches. Because of the common-sense approach to analysis, and the accessibility of the concepts, the methods can be taught to and applied by novice analysts.

The specific discourse analysis techniques used are not novel. In the first method, we provide a new set of indicators of interaction difficulty, based on previous work regarding analysis of the coordination work of users. For the second method, techniques detailed in Lockman and Klappholz (1978) and other literature on coherence and reference resolution (Hirst 1981; Grosz and Sidner 1986; Grosz, Joshi and Weinstein 1995), as well as examining formulation of and grounding of references (Clark and Brennan 1991; Clark 1996) are all relevant. The method involves tagging references in the discourse and combining them into coreference chains. By assigning types to the referents pointed at by



```
crooked:     hi
crooked:     how do u paste it
elsewhen:    ok
elsewhen:    highlighted the whole thing, did ctrl-c to copy it, then ctrl-v to paste
elsewhen:    looks good to me
crooked:     ok so now we change this one around
elsewhen:    yep yep.  mouth first sice it's first on the list?
crooked:     ok
elsewhen:    ok.  looks like it's just 2 arc's (from the picture on the left)
>>
```

```
(.drawOval g 123 100 30 50)
(.drawOval g 22 100 30 50)
(.setColor g black)
(.drawOval g 120 80 80 120)
(.drawOval g 220 80 80 120)
(.repaint c))
```

**Figure 1: Interface for the Group Homework Tool experiment; to the right, the face produced by subjects in the experiment session explained below**

these coreference chains, and computing statistics about these referents, we make visible the ways participants handle different types of information. Although grounding anaphoric references and forming coreference chains can be problematic (van Deemter and Kibble 1999), we have achieved good results by restricting the set of referents under consideration and aiming for investigative, rather than comprehensive, data.

Many of the systems presented in this paper were developed using THYME, a component-oriented groupware toolkit built by our lab (Landsman, forthcoming; Landsman and Alterman 2003). THYME enables a developer to quickly and easily generate groupware systems that will automatically record interaction data. Because we were able to reuse pre-built groupware components, like chat windows and shared tables, development time was significantly reduced. Additionally, THYME interaction data can be replayed using SAGE, a data playback program (Landsman and Alterman 2002), which provides a variety of tools for performing both qualitative and quantitative analyses of the data.

We will begin by presenting an initial discussion the methods in the context of an experimental domain, the Group Homework Tool. This will be followed with a discussion of related work. After this, a case study will serve to further explain application of the methods, and demonstrate the results that can be achieved using them. Finally, we present experimental evidence of the general applicability of the methods.

## 1.1 Analysis methods

To illustrate the analytic methods we have developed, we present an example taken from one of our experimental domains, the Group Homework Tool (Langton, Hickey, and Alterman 2004). This project studied the educational impact of having students program in pairs versus solo programming. Two students are situated at two remote computers, and asked to complete a coding assignment together using a chat tool, a shared editor, and a pair of shared web browsers. Before and after the assignment, the students are individually tested on their programming skills; the results are then compared to examine knowledge transfer during pairs programming.

| Line | | User dialogue | Comments |
|---:|---|---|---|
| 7 | B: | yep yep. mouth first sice it's first on the list? | *The instructions provide sample code, plus a list of required face components.* |
| 8 | A: | ok | |
| 9 | B: | ok. looks like it's just 2 arc's (from the picture on the left) | *The mouth is shown in the instructions as a semi-circle constructed from two arcs.* |
| 10 | B: | although I'm not sure what the parameters for .drawArc are..... | *B initially guesses parameters by looking at the sample code, rather than the manual.* |
| 11 | A: | how will we be abe to see if its correct | |
| 12 | B: | well, [it] doesn't have to be 100% correct i'm guessing..[it] just has to look similar | *The instructions allow for some leeway in the actual appearance of the face.* |
| 13 | A: | ok | |
| 14 | B: | so we just use the eval. button and pray it looks ok :) | *The students can use the* `eval` *button to execute and debug the code* |

**Figure 2: Discourse from the example Group Homework Tool session.**

A screenshot of the Group Homework Tool is shown in Figure 1. In the middle is the shared editor; above it is the chat window. On either side are shared browsers to view the instructions and the reference manual for the programming language. All communication and actions are recorded for later analysis. The assignment in this case was to draw a cartoon face, with various features such as a nose, mouth, and so forth.

The completed face that one pair of students constructed is shown on the right side of Figure 1; a portion of the chat transcript from near the beginning of this session is shown in Figure 2. Note that the dialogue is copied exactly as it was typed, with elisions, misspellings, and the like retained intact. Necessary interpolations of the dialogue are indicated with square brackets.

In this example a pair of students have completed their pre-test and are beginning their problem-solving session. The two students, who have not met before, are of disparate skill levels, and as a result user B ends up tutoring user A for most of the session. At the start of this excerpt, user B has copied and pasted some sample code from the instructions into the shared editor. The two users begin by discussing what part of the multi-part assignment to do first, and then move on to discuss implementation details. We will use this analysis to demonstrate application of our methods; a more thorough discussion of analysis results will be presented in a later section.

## 1.2 Recurrence analysis

Our first method, *recurrence analysis*, highlights specific portions of an interaction where participants are having difficulty. Recurrence analysis is a lens that can be applied during observation of a domain. It builds on previous work which examines ethnographic data for recurring indications of difficulty (e.g., Suchman and Trigg 1991). In the process of observing interaction, whether live or replayed, the analyst notes interactions of three particular kinds:

- **Recurrent communication about coordination** — Situations where participants must repeatedly discuss their coordination to perform a joint activity. This indicates that the available procedures for coordination may be insufficient.
- **Recurrent errors of coordination** — Situations where participants repeatedly commit errors. This is a good indication that the process is too difficult to perform correctly with the tools available.
- **Creation of secondary structure** — Conversational or procedural mechanisms devised and employed by the participants to help them coordinate their actions. It is a clear indication that the existing structure is insufficient.

Identification of problem areas using these indicators allows an analyst to focus redesign efforts. However, these are not meant to be *de facto* indication of the need for redesign — rather, they provide evidence for a need for further investigation. For example, recurring communication occurs naturally as a part of any repeating task, and if the analyst determines it to be part of smooth coordination, this does not necessarily indicate the need for a redesign. However, if the participants must repeatedly discuss their coordination, rather than being able to establish conventions and procedures which solve their problems, it indicates that the tools available to them may be insufficient to accommodate these coordination needs. Hence, such situations are a strong indication that the system should be redesigned.

In the example shown in Figure 2, participants repeatedly refer to the manual and instructions, but have no good way to point at particular portions of them. Instead, participants must construct long references like those on lines 9 ("the picture on the left") and 10 ("the parameters for .drawArc"). This is a source of increased work and causes errors later in the transcript. From this indication, an analyst might decide that a potential redesign should include a way to highlight a particular portion of the manual, reducing the overall team work required.

## 1.3 Referential structure analysis

The second method is *referential structure analysis.* The aim of referential structure analysis is to discover what sorts of things participants in a joint activity spend their time talking about, how much they talk about them, and for how long. For example, an analyst might investigate how frequently participants refer to some domain object, or how long they spend discussing a plan for action. To do this, the analyst tags the references made by participants and consolidates them into chains, classifies the resultant referents into types, and computes various parameters about each referent identified in the dialogue.

| Line | | Discourse | References | *mouth* | *plan* |
|---|---|---|---|---|---|
| 7 | B: | yep yep. mouth first sice it's first on the list? | $\underline{\text{mouth}}_a$, $\underline{\text{it's}}_b$, $\underline{\text{mouth first}}_c$ | a, b | c |
| 8 | A: | ok | $\underline{\text{ok}}_a$ | | a |
| 9 | B: | ok. looks like it's just 2 arc's (from the picture on the left) | $\underline{\text{ok}}_a$, $\underline{\text{it's}}_b$ | b | a |
| 10 | B: | although I'm not sure what the parameters for .drawArc are..... | | | |
| 11 | A: | how will we be abe to see if its correct | $\underline{\text{its}}_a$ | a | |
| 12 | B: | well, [it] doesn't have to be 100% correct i'm guessing..[it] just has to look similar | $\underline{\text{[it]}}_a$, $\underline{\text{[it]}}_b$ | a, b | |
| 13 | A: | ok | | | |
| 14 | B: | so we just use the eval. button and pray it looks ok :) | $\underline{\text{it}}_a$ | a | |

**Figure 3: Analyzing GHT data with referential structure analysis**

In Figure 3, we show the results of tracking two referents through sample dialogue from the GHT domain. There are many possibilities for reference: code constructs, elements of the instructions, plans for action, the shape of the desired output, division of labor, and so forth. For clarity, we have pulled out only two referents from the dialogue, and marked each reference with a unique subscript; references are sorted into separate columns depending on which referent they point at.

The first of the referents is the "mouth" referent, referring to a portion of the face which the code is meant to produce. References to it are collected in the second column from the right. On line 7, B refers to it by name ("mouth"), followed by an anaphoric reference ("it's"). On line 9, B refers to the mouth — in this case, the picture of the mouth provided in the instructions. Discussion on lines 11 and 12 refers to the mouth a number of times, including by means of elided pronouns; and at the end of the dialog on line 14, B refers to the mouth once more. From this, we can see that the mouth referent remains relevant for some time — in this segment, the first reference to it is on line 7, and the final reference is on line 14. (Conversation about it continues past this brief segment of dialogue, but we are restricting analysis to this segment for didactic purposes.) For this segment of discourse, it has a lifetime of relevance to the participant of seven utterances (inclusive), and is referred to seven times in five separate utterances during that lifetime.

Participants discuss a plan for drawing the mouth at the start of the extract. This referent is referred to differently than the "mouth" referent. References to this plan referent are noted in the right-most column of Figure 3. On line 7, B proposes the plan ("mouth first [. . . ] ?"); on line 8, A accepts ("ok"), and on line 9, B acknowledges acceptance ("ok"). This constitutes the entire conversation about this plan. In contrast to the mouth referent, the plan for ordering tasks is relevant for three utterances (lines 7–9), but after this is never discussed again. Although the plans continues to be a topic for discussion — lines 10–14 are primarily concerned with how to carry out the plan — the participants have

| Method | Source data | Focus | Goal |
|---|---|---|---|
| Recurrence Analysis | Ethnographic | Representation work | Discovery of design problems |
| Referential Structure Analysis | Ethnographic | Referential structure | Identification of information use |
| Discourse Analysis | Ethnographic | Team work | Understanding of social interactions |
| (Hierarchical) Task analysis | Task design | Task work | Task efficiency |
| GOMS | UI design | Interface work | Interface efficiency |
| Collaboration Usability Analysis | Ethnographic, UI prototypes | Team work / workflow | Process design recommendations |
| Distributed Cognition | Ethnographic | Representation system; artifacts | Understanding of cognitive system |
| Other workplace analysis methods | Ethnographic | Work practice; Social structure | Understand interaction of design and practice |

**Figure 4: Comparison of analytic methods**

committed to it and do not refer to the plan itself again. This particular plan referent has a short lifetime of relevance (three utterances), and is mentioned three times over that span.

Observing and quantifying these sorts of characteristics of information access, and quantifying the ways that information is handled, are the goals of referential structure analysis. Later in the paper, we will use experimental data from a case study to demonstrate the results of applying these methods to real data, and explore the utility of these methods in redesigning groupware systems.

# 2 Related work

Figure 4 gives an overview of some of the existing methods available for analyzing interaction. These methods use a variety of sources of data, ranging from ethnographic observation of participants engaged in their everyday work activity, to investigation of online interactions, to *a priori* designs of future interactions, to prototype design. We will focus on those methods that have been specifically designed to analyze ongoing online interaction.

**Conversation analysis** (Sacks, Schegloff and Jefferson 1974; Schegloff 1991; Sacks 1992) and related techniques have been used to examine the minutiae of interaction. Conversation analysis identifies the specific devices, such as conversational openings or adjacency pairs, that participants use to organize their talk. Our methods grow out of conversation analysis, but are focus on following the ebb and flow of interaction using conversational indicators as markers to reveal areas of potential redesign.

Turn-taking, speaker choice and speech act type can be used to identify breakdowns in coordination by highlighting departures from a standard model of interaction (Goodman et al. 2005). However, their work is primarily concerned with identifying and solving breakdowns as they occur (via intervention of an intelligent agent), our methods are concerned with how breakdowns are handled by the participants, and how to redesign the system to reduce their incidence.

Examination of the duration and type of conversational utterances has also been used as a way of determining the impact of alternate representations on conversation (Kraut, Fussell, and Siegel 2003). This work provides general conclusions about the impact of providing additional representations to participants, in this case a video stream of one user's activity; it is based around formal experimentation and analysis of the conversation among participants. In contrast, we look at the conversation that emerges from an ongoing practice and seek to redesign the interaction based on this analysis.

**Hierarchical task analysis** (HTA; see, e.g., Kirwan and Ainsworth 1992) breaks up behavior into a set of goal-oriented tasks. By mapping out the task structure, an analyst can make salient such difficulties as missing steps, mistimed steps, bottlenecks, and redundant work. The model of tasks is generated by the analyst from an idealized expression of the domain, possibly informed by observation of participants in the domain. Critical work has been done that argues that these sorts of hierarchical plans are more a resource for action or an *ex post facto* description of what happened, rather than providing a plan for future action by the participants (Garfinkel 1967; Schank and Abelson 1977; Suchman 1987; Grosz and Kraus 1996). Given this critique, modern usage has been to include a task-analytic perspective as a part of a larger approach; this is the method in both Collaborative Usability Analysis and Cognitive Work Analysis.

**GOMS** (Card, Moran, and Newell 1983; John and Kieras 1996) extends HTA to analyze the interface work required to perform a task by adding experimentally-determined timing information and an explicit model of mental operations. The resulting method helps an analyst predict interface work necessary to complete a task. GOMS works well to inform design choices early in the product development cycle, allowing a designer to examine alternate designs and determine the suitability of each for a specific task. Combining GOMS modeling with recurrence analysis benefits both approaches. Recurrence analysis serves to highlight recurring problems in an interaction; GOMS allows the analyst to measure the interface-work efficiency of a proposed solution to such a recurring problem. Integrating results from referential structure analysis produces an interactional design which a GOMS analysis can be used to refine.

**Collaboration Usability Analysis** (CUA; Pinelle, Gutwin, and Greenberg 2003) is a discount evaluation technique aimed at reducing the cost of evaluating groupware design. CUA constructs a task model of the desired interaction which includes the team work of participants and interaction between participant actions in a structured fashion. It derives design recommendations based on the expected interactions of participants, seeking to head off difficulties by mapping out the interacting paths of activity that participants must take to achieve their goals. In contrast, our methods analyze the work practice of participants in an existing, ongoing online interaction to aid redesign.

**Distributed cognition** views the work of participants as a sequence of representational states and the transformations between those states (Hutchins, 1995 (twice); Hollan, Hutchins, and Kirsh 2000). From this perspective, much of the work of the users is either mediated by representations or is concerned with the articulation, transformation, and alignment of information between representations. By enumerating this work, an analyst can see how participants use the tools in their environment to complete tasks and to interact with others. Some of the representations users work with in order to stay coordinated have an unstructured format, such as speech or textual chat. The discourse methods we have developed provide a structured way to analyze these free-form communication channels.

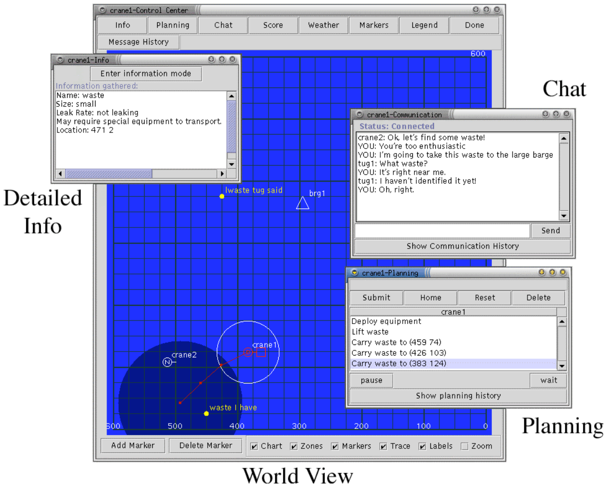
**Other workplace analysis methods** investigate the impact of introducing new technology into an ongoing interaction. Activity Theory (Leont'ev 1978; Bødker 1990; Cole and Engeström 1993; Kuutti 1996) seeks to identify the interactions between the design of the work environment — including the people, tools, and rules that comprise it —and the emergent work practice by examining conflicts between different levels of work. Cognitive Work Analysis (Rasmussen 1986; Vicente 1999) investigates interactions between the actions available to workers and the expectations put on them by enumerating the constraints on an activity and comparing the residual set of possible behaviors for the system to the set of actions required to achieve the desired goals. Workflow analysis (Basu and Blanning 2000; Sierhuis and Clancey 2002) constructs functional models of the work process, and examines interactions between participant tasks, informational elements, and the resources needed for work. All these methods are concerned with the social and cultural impact of introducing new technology into a workplace. Our methods have been developed primarily to address redesign of the interaction for an existing online collaboration, but they could also be applied to provide data for these higher-level methods.

# 3 Case Study

We turn now to a case study to explain our methods more thoroughly. We applied recurrence analysis to data from VesselWorld, a groupware testbed, and used the results of this analysis to redesign the system. We then performed an experiment comparing the redesign to the original. Finally, we analyzed the resulting data using referential structure analysis and used the results to explain successes and failures of this redesign.

## 3.1 VesselWorld

VesselWorld is one of the groupware systems our lab created to study issues in same-time/different-place coordination. It is a turn-based multi-user simulation where three users situated at separate computers conduct a clean up of a harbor via a graphical interface. Though the users cannot see or hear each other, they are able to chat via the VesselWorld interface. As the users interact, the system logs all actions and communication for later analysis. The ability to generate and play back transcripts of interaction makes



Information gathered:
Name: waste
Size: small
Leak Rate: not leaking
May require special equipment to transport.
Location: 471 2

Status: Connected
crane2: Ok, let's find some waste!
YOU: You're too enthusiastic
YOU: I'm going to take this waste to the large barge
tug1: What waste?
YOU: It's right near me.
tug1: I haven't identified it yet!
YOU: Oh, right.

Deploy equipment
Lift waste
Carry waste to (459 74)
Carry waste to (426 103)
Carry waste to (383 124)

**Figure 5: The VesselWorld interface**

VesselWorld ideal for exploring issues of group interaction. At this time, we have collected over 300 hours of data from more than 20 groups using various versions of VesselWorld.

In the VesselWorld system, each user acts as the captain of a ship navigating a simulated harbor. The harbor contains barrels of toxic waste that must be safely retrieved and loaded onto a large barge. Two users pilot ships with waste-retrieval cranes attached (referred to as crane1 and crane2), allowing them to lift and load barrels of toxic waste; the other user pilots a tugboat (referred to as tug1), and is able to move small barges around the harbor, identify barrels, and seal the leaks caused by mishandling of barrels. Each user can see a small region of the harbor close to them. The users clear the harbor in a turn-based fashion, with the users explicitly planning their action for a turn before jointly submitting them to the system for evaluation. The basic interface is shown in Figure 5. The large central window shows the harbor, with the small portion of the harbor currently visible to the user shown as a darker circle in the lower left. Clockwise from the left, the smaller windows are: the Info window, displaying detailed information about objects in the harbor; the Chat window, allowing textual communication with other users; and the Planning window, with controls for creating a plan for domain action.

VesselWorld participants have only a few channels of communication available to them for coordinating their activity. Primary among these is textual chat, which provides a very flexible means of expression. Participants also made extensive use of private markers, which provided a way to attach a private textual annotation (visible only to the user) to a section of the harbor, to keep track of information about the barrels of toxic waste. In addition, there is evidence of participants coordinating their activity by taking advantage of the visibility of actions of nearby vessels in the harbor: when two vessels are close enough to actually see each other's current state, participants make use of this information in their planning.

| | | |
|---|---|---|
| crane2: | what eq is needed for the small on top of the attached barge | *Crane2 inquires what equipment is required to lift a waste.* |
| crane1: | none | *Crane1 has incorrect info.* |
| tug1: | Dredge | *Tug1 provides conflicting info.* |
| crane1: | huh? i thought that was the sm none? | *The group must now spend time reconciling the conflict; as Tug1 can get the info directly, his version is treated as authoritative.* |
| tug1: | It apparently isn't. | |
| crane1: | k | |

**Figure 6: A mistake in transcribing information leads to confusion.**

## 3.2 Recurrence analysis

We applied recurrence analysis to data collected with VesselWorld. In the data we found evidence of each of the three indicators outlined above, and used this evidence to motivate a redesign of the system. Examples of these, pulled from our experimental data, are discussed below; results of an experimental comparison of the redesign to the original follow after.

**Recurring communication about coordination** — Analysis of the VesselWorld data revealed a number of areas where participants repeatedly discussed their coordination. One of the most frequent patterns involved the reporting of toxic waste information. A large portion of the communication participants generate during the early part of the session consists of participants reporting the discovery of new barrels of waste. Due to the nature of the task, barrels could be discovered by any user, but required a particular set of actions involving one, two, or all three users to handle successfully. For this reason, succesful clearing of the harbor depended on participants sharing information about newly-discovered barrels. Because of the frequent reporting, each group eventually settled on their own stylized vocabulary and interaction pattern for reporting barrels. Despite the conventions that groups created for reporting this information, the task of not only reporting but also understanding, transcribing, and remembering the information was cumbersome and error-prone. As will be discussed below, this difficulty was addressed in the redesign.

**Recurring errors in coordination** — In the VesselWorld data, we found recurring errors in (among other situations) recall of toxic waste information, planning of future actions, and planning and execution of joint actions. Discrepancies frequently intruded into the flow of information surrounding toxic waste reporting, creeping into each step involved in discovering new barrel: reporting the information, understanding that report, properly transcribing it to a local representation (whether internal or external), and recalling it from that representation when the time came to act.

A typical situation is shown in Figure 6. Here, the group has difficulty because private representations have become misaligned — Crane1 created a local marker indicating (incorrectly) that a particular barrel of toxic waste requires no equipment to lift, whereas Tug1 (the authoritative source of equipment information) has a private marker indicating it requires the Dredge. While the problem is quickly resolved, this sort of mismatch of

| | |
|---|---|
| crane1: sub Lift | *Crane1 proposes to submit ('sub') a joint lift plan step, now, of a predetermined barrel.* |
| crane2: k | *Crane2 explicitly commits to action, completing the adjacency pair.* |
| crane1: sub Load | *Crane1 continues the formula.* |
| crane2: k | *Crane2 again explicitly commits.* |

**Figure 7: Adjacency pairs in VesselWorld dialog**

private representations (whether internal or external) is endemic in almost any situation where participants have their own private views of shared information. This difficulty was addressed in the redesign by introducing shared representations which reduced or eliminated the extra work required to align private representations of toxic waste data and plans.

**Creation of secondary structure** — An example of secondary structure is shown in Figure 7. In joint lifts, where the two cranes needed to align their domain actions to lift a large or extra-large barrel of toxic waste, timing of the joint actions was very error-prone. Users were not able to see what plans other users had submitted to the system, leading to timing mistakes and general frustration. Participants eventually established structural conventions in their discourse to organize their actions, such as the adjacency pairs (Schegloff and Sacks 1973) shown in the sample dialogue.

However, producing and using this structure proved quite time-consuming, and the procedure was itself error-prone. Because of the limited tools available to the participants to structure their work they were not always able to successfully construct solutions. Indeed, here is no guarantee that the organizational structure that the users add will improve the situation at all; it is possible that some problems of coordination are best dealt with using a context-free form of communication like textual chatting. In general, however, introducing properly structured representations which match the interaction will improve performance in such situations; experimental evidence for this particular case is presented in the next section.

## 3.3 The VW3 Experiment

We redesigned VesselWorld based on this analysis to include three new *coordinating representations* (CRs). Coordinating representations (Alterman et al. 2001) are ubiquitous cognitive artifacts (Norman 1991; Schmidt and Wagner 2002) that give participants a way to organize their behavior in a joint activity by creating shared expectations of roles and actions and by partially structuring actions. For example, a stop sign creates expectations in the participants of a joint traffic activity but does not determine activity completely. An agenda for a meeting serves both to organize activity by partially ordering topics for discussion and to create expectations about the structure of the meeting. CRs serve to simplify a task both by offloading some of the cognitive load of the task, much the way a notebook serves to ease the burden of remembering information (Perkins 1993). CRs also serve to make problem-solving easier; for example, “complex sheets” for handling airport luggage help baggage handlers align multiple sources of information (Suchman and Trigg 1993).

We conducted a single-variable experiment to assess the impact of introducing new representations. There were two sets of subjects: the control set (which we will call the non-CR groups) used the version of VesselWorld shown above. The other set (the CR groups) used a version of VesselWorld with three new coordinating representations added: the Shared Planning representation, which made other users' planned actions visible; the Object List representation, which provided a persistent repository for toxic waste information in tabular format; and the Strategy representation, which provided a shared agenda which users could employ to plan out future actions.

Each set consisted of three groups of three people. The subjects were a mix of area professionals and undergraduate students; all were paid a flat fee for participating. Each group was trained together for two hours in use of the system and then spent approximately ten hours solving VesselWorld problems. To alleviate fatigue concerns, the experiment was split into three four-hour sessions. While no time constraint was imposed on the users' harbor-clearing task, the proctor emphasized minimizing both the number of turns required for a problem and the number of errors committed. A group score also provided feedback to the users as to their progress. These proved adequate to encourage users; exit interviews indicated strong user involvement.

Subjects were asked to fill out entrance surveys to obtain population data and exit surveys to get feedback about their experience with the system and their group. A set of random problems was presented to the subjects. Groups did not necessarily see the same problems, nor in the same order, and because of differences in performance, did not complete the same number of problems. To account for this, a general measure of the complexity of a particular problem was devised, taking into account the quantity and type of the barrels of toxic waste in the harbor, their distance from the large barge, and the number of small barges available to the subjects. This metric was used to normalize results. To account for a long learning curve, the quantitative results presented are a comparison of the final five hours of play for each group; by this point the performance of the groups had stabilized.

### 3.3.1 Quantitative results

The experiment produced a number of results; major results are summarized in Figure 8. The performance of the CR groups was significantly better than non-CR groups according to many measures: clock time necessary to solve a problem, interface work (measured as the number of system events generated per minute), and number of errors committed that resulted in leaking barrels. Performance in some of the trouble areas we had previously identified — close coordination, domain object reference — was notably improved, and errors due to miscommunication of object information was reduced.

The most significant effect, though not the one of greatest magnitude, is the 58% reduction in the lines of chat generated per minute. This is a very interesting but not

| Measure | Non-CR groups | CR groups | Improvement |
|---|---|---|---|
| Chat communication (lines per minute) | 5.53 | 2.35 | 58% ($p < 0.01$) |
| Solution time (minutes per session) | 96.7 | 56.6 | 52% ($p < 0.01$) |
| Speed of play (rounds per minute) | 1.29 | 1.73 | 34% ($p < 0.05$) |
| Mistakes (errors per minute) | 0.121 | 0.047 | 62% ($p < 0.2$) |

**Figure 8: Comparison of CR and non-CR groups in VesselWorld**

unexpected result, indicating the migration of a great deal of routine task-related communication to other representations. It is reminiscent of the work of Kraut et al. (op cit.), who posited a view of visual information as an alternative resource for communication. In a similar fashion, the communication previously appearing in the chat window vanished, in favor of communication via other representations: the Object List representation for toxic waste information, and the Shared Planning representation for close coordination of actions.

Also highly significant is the 42% reduction in clock time per session. Coupled with the result for the increase in rounds of activity per minute — up 34% — we see that the CR groups worked faster with less interface effort. This provided indication that the new system is more efficient. This was due to two effects: first, the new representations were faster to use — preliminary GOMS analysis showed that entering information in the Object List is about 20% faster than using chat. Second, the persistence of information meant participants were able to eliminate a number of steps in the lifecycle of toxic waste information, such as creating private markers (which went almost entirely unused in the CR groups), searching for the information in chat history, or redundantly communicating old information to other users.

Mistakes resulting in leaking barrels of waste were reduced by a dramatic margin (62%); however, variability in the data (such errors were infrequent) made this reduction statistically insignificant ($p < 0.2$). Nevertheless, we feel confident in the veracity of the finding. Anecdotally, errors due to mismatched expectations about the equipment type needed to handle a barrel of toxic waste — the most common type of error — were reduced by the availability of the Object List shared representation. Having an authoritative source of this information reduced communication overhead, which in turn reduced opportunity for errors in transcribing the information. This was as expected, given the strong indications seen in the preliminary data.

### 3.3.2 Unexpected results

However, there were a number of unexpected results. One troublesome result was that certain columns of the Object List representation went unused — specifically, the ‘Action’

| Type | Frequency | References | Lifetime | Density |
|---|---:|---:|---:|---:|
| Plan | 57% | 3.4 | 12.0 | 28.5% |
| Waste | 17% | 6.6 | 168.7 | 3.9% |
| Location | 8% | 2.6 | 62.6 | 4.2% |
| Repair | 8% | 3.0 | 4.8 | 62.5% |
| Barge | 4% | 11.9 | 294.0 | 5.6% |
| Vessel | 4% | 3.1 | 183.6 | 1.7% |

**Figure 9: Referential structure data from the VW3 experiment.**

column, meant to aid users in tracking the next action to be performed on a barrel of waste, and the 'Leaking' column, used to indicate that a barrel was leaking and needed immediate attention. This was notable because these two pieces of information were the subject of nearly as much communication as equipment, which as noted above was used extensively in the Object List.

Most disconcertingly, the Strategy representation was not used at all once training in it was complete. Subjects in the CR groups gave some insight into why: "We never used the Strategy window because we could see what we were doing in the planning window." One user's assessment of the fundamental problems with the High-level Planning CR was especially interesting: "...since all plans must de facto be agreed upon by all (relevant) players, negotiation via the Chat window is required. Since the plans are discussed in detail there, putting those plans in the Strategy window would be redundant."

Clearly, the users felt the Strategy representation did not match the way they handled the planning information it was supposed to be storing. These results could not be accounted for by our recurrence analysis. However, we did note that the discrepancies seemed based on the way that participants exchanged and recalled information. This led directly to the development of referential structure analysis.

## 3.4 Referential structure analysis

Using concepts from distributed cognition, we followed information as it flowed from representation to representation within the VesselWorld system. Information usually had a simple life cycle. For example, information about a barrel of waste is first discovered via exploration; optionally, the discoverer uses the Info Window to retrieve ancillary details; the information is then reported, either in the chat window, or (in the CR groups) via the Object List or Shared Planning representations; it may then be noted by other users and potentially stored in private representations; at some future time, it again becomes relevant and must be retrieved from either public or private representations; and finally the barrel is dealt with and the information becomes irrelevant.

After some exploratory analysis, we settled on a small set of referent types for our referential structure analysis of the VW3 data. A summary of the data (for the non-CR groups), organized by referent type, appears in Figure 9. The most notable result was the

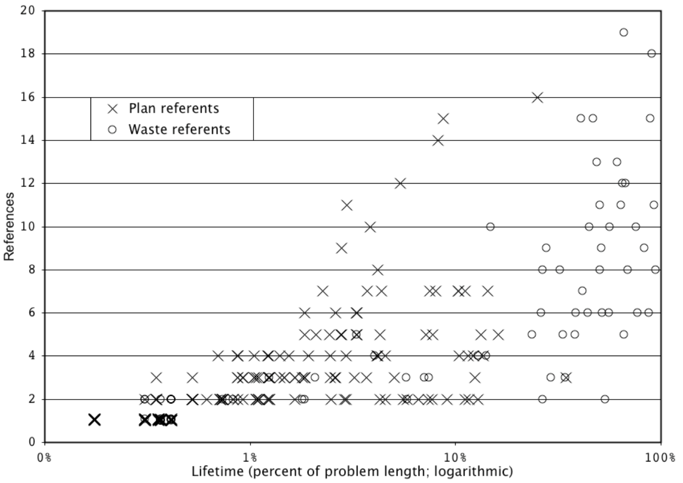


**Figure 10: Scatter plot reveals differences between plan and waste referents**

obvious difference between plan and waste referents. Plan referents, by far the most numerous type (representing 57% of the tagged referents), tended to have a short lifetime (averaging 12 lines of chat). In comparison, toxic waste referents were relevant for a much longer period (averaging about 169 lines of chat). This meant that information about a barrel of toxic waste had to be retained in some representation for that rather long period — whether in a participant's memory or in one of the available external representations.

Another result was the difference in density between plan, waste, and repair referents. Density — the ratio of references to lifetime — is a rough measure of how dominant a referent is within the conversation; a referent with a high density dominates conversation, occurring in almost every utterance. Repair referents (with an average density of 63%) did just this; almost two out of every three utterances referred to the repair during its lifetime. As a result, it seems unlikely that participants would require a new external representation to mediate repairs; because of their tendency to short lifetimes and high density, repairs can be adequately handled in the chat window. In comparison, the low density of waste referents reinforces the conclusion above, that is, that a persistent external representation will reduce the effort needed to share information about barrels of toxic waste.

### 3.4.1 Visualizing the data

Visualizing the analysis data is useful for revealing general differences between referent types. One example is shown in Figure 10. Here, plan and waste referents from non-CR groups are compared using a scatter plot. The horizontal axis is the percent lifetime of relevance — derived by dividing the lifetime of the referent by the total number of utterances in the session. The vertical axis is the number of references. To reveal where multiple data points overlap, the source data has been altered by adding small fractions to the discrete source values. Also, due to the wide variance in the data, the horizontal axis is logarithmic.

This sort of graph is useful for examining the data for outliers and to check the distinctiveness of referent types. The plot clearly shows the separation between plan referents, which primarily fall to the left of a dividing line at 20% lifetime, and waste referents, which primarily fall to the right of this imaginary line. However, there are a number of waste referents which have a shorter lifetime, overlapping the characteristic pattern of plan referents. By going back to the source data, we were able to determine that these corresponded to fictitious referents.

Fictitious referents occur when participants refer to an object that does not exist. A referent may be entirely fictitious, existing only in the mind of some or all of the participants. Additionally, multiple conversational referents may exist for a single "real" object, because the understanding shared by participants may not match objective truth. For example, participants may (individually or collectively) forget about an object, only to later discover it again and treat it as an entirely separate referent. We are interested in what the users are discussing, rather than establishing some authoritative set of objects, and so track conversational referents as the participants create them.

Hence, in the above scatter plot, some of the waste referents do not correspond to actual barrels of toxic waste in the harbor. Re-examination of the data showed that, in fact, all waste referents with four or fewer references were fictitious. While it is unlikely that this result is true in all cases, an intelligent system could use a tendency such as this to aid participants. For example, if waste referents are encoded in an external representation like the Object List, and participants fail to access that information, the system could make it more salient on the assumption that it is spurious and needs to be dealt with.

## 3.5 Explaining unexpected VW3 results

These observations gave us insight into why participants in the VW3 experiment did not make use of all available representations. The Strategy representation was intended to store and communicate planning information. However, the way in which the representation presented information was at odds with the way that users shared planning information. Our analysis showed that users discuss plans only briefly: the lifetime for a plan referent averaged twelve lines of chat, and was frequently much shorter. Because of this, it was barely worth the effort for a participant to encode the plan into the Strategy window, a task that was noticeably more difficult than simply describing the plan in chat.

Also in conflict with the emergent work practice was the fact that the Strategy representation required a user to fully describe a plan from the start. Some users, as noted previously, felt this made the act of creating a plan in Strategy a form of authoritative planning, robbing others the opportunity to participate in negotiations about that plan. Finally, because of the relatively short lifetime of plans, participants were usually only discussing one or perhaps two plans at a time. This meant that the burden of remembering the current plan information was minimal: participants could rely on short-term memory to store this information, as it was lightweight and did not need to be referred to later. Hence, there was little need for a persistent representation.

The case of the unused columns in the Object List required careful reinvestigation of waste referents. As one user wrote in the exit survey, "...the Object List had too many options. Many weren't used because we were in constant chat contact." Indeed, we found that despite the fact that waste referents had long lifetimes and low density — implying the need for a persistent representation — some aspects were handled differently. Specifically, the status information that the 'Action' column was designed to store changed very frequently; additionally, transitions between states were either broadcast by users in the chat window as a side effect of planning, or were uninteresting to other users and hence went unshared. As a result, the effort to update the column appeared unnecessary to users; the representation was superseded by other procedures.

The 'Leaking' column provided a persistent, shared representation for a simple but important fact: whether or not a particular barrel was leaking. However, in practice, a leaking barrel dominated the activity. Barrels leaked infrequently; there was a high cost (in terms of score) of leaving a leaking barrel unattended; and in almost all cases only one was leaking at a time. Due to these factors, a leaking barrel became the focus of the participants. Because of its importance, and the simplicity of the information, participants were willing and able to store the fact that a particular barrel was leaking in their short-term memory. They did not need a persistent, shared representation to remind them of the leaking status.

It is interesting to note that this behavior would not be predicted by, e.g., a GOMS analysis of the interface; the work required to fill in the 'Leaking' column of the Object List was about the same as that required to fill in other columns. Likewise, task analysis of the toxic waste reporting procedure would indicate that both sorts of information should be recorded. But the emergent practice of the participants showed that, because they had no need for the leak status to be stored long-term, they skipped transcribing it entirely. In contrast, referential structure analysis of the discourse clearly indicated that this behavior had emerged and why.

# 4 General applicability of the method

To establish general utility of our methods, we demonstrate a few important qualities of the method. This section will summarize experiments that were performed to verify these claims.

1. Referential structure depends on referent type.
2. Representations affect referential structure.
3. The methods can be taught to other analysts.
4. Different analysts draw similar conclusions from the same data.
5. Students produced superior redesigns when taught these techniques.

We performed a statistical analysis of the VesselWorld data presented above to investigate whether the first two hypotheses were valid. Validation of these two hypotheses provides important guarantees that the conclusions we are drawing from referential structure analysis are meaningful. To test the three remaining hypotheses, we ran two experiments with the help of a class of students to demonstrate that the methodology could be taught and applied to novel domains. In all cases we found that the data

| T-Test | % Lifetime | References |
|---|---|---|
| Plans | $t(210) = 0.38, \quad p > 0.5$ | $t(210) = 1.96, \quad p < 0.1$ |
| Wastes | $t(72) = 3.61, \quad p < 0.01$ | $t(72) = 2.96, \quad p < 0.01$ |
| Locations | $t(43) = 1.48, \quad p < 0.15$ | $t(43) = 1.41, \quad p < 0.2$ |

**Figure 11: Representation system has an effects on information access patterns for certain referent types.**

indicated the hypotheses were true.

## 4.1 Referential structure depends on referent type

We analyzed the VesselWorld data to investigate whether differences in the referential structure were correlated with referent type. Our hypothesis was that groups using the same system for the same task exhibit similar statistical measures for each type of referent. Formally, we measured whether differences between referent type statistics (lifetime and references) are larger than the variation due to having different people in a group.

The data supported the this hypothesis. We found that different non-CR groups exhibit slight variation in how they handle a particular type of information, but that these differences are minor in comparison to the differences between information types. To show this statistically we examined the differences between participant groups in our experiment. We chose to focus on the three most commonly occurring referent types (plans, wastes, and locations). An F-test on these referent types showed that effect due to group membership was not at all significant. The miniscule values for the F-test (e.g. for plans, $F(1, 210) < 0.5$, $p < 0.01$) indicate that variability between groups is much less than variability within groups. The conclusion is that it is highly unlikely that access patterns depend on which particular group of participants generates them; they must instead depend on other variables, such as the type of the referents.

## 4.2 Representations affect referential structure

Our second hypothesis was that groups using the different systems for the same task would generate significantly different referential structure. That is, by providing alternate representations for certain types of information, we would alter how participants talked about that information. Formally, we compared the distributions of referents for CR groups and non-CR groups to determine whether they could have come from the same source population. We compared referent statistics (lifetime and references) from the CR groups to comparable data from the non-CR groups. A T-test performed on the data generated for the two experimental conditions showed that it was unlikely that the differences between the data for the non-CR and CR conditions were due to chance; therefore, we tentatively attribute the change in information access patterns to the change in representations. A summary of the relevant figures appears in Figure 11.

As expected, the effect that introducing new representations had on referents was dependent on referent type. Most noticeable is the strong effect on waste referents. As explained previously, the new representations altered the way that users talked about waste referents much more heavily than they way they talked about plans and location referents. This is reflected in the data: plan referents (which were generally unaffected due to rejection of the Strategy window) show little effect, whereas the statistics for waste referents (strongly affected by the introduction of the Object List) show a very significant change. Location referents, whose characteristics were changed somewhat by the availability of the Object List "location" column, show a moderate degree of separation.

Overall, this test showed that there were significant alterations in the referential structure generated by participants after new representations were introduced. This is in line with the observations of other researchers, who have shown using other features of conversation that availability of alternate representations significantly alters how participants talk (Clark and Wilkes-Gibbs 1986; Gergle, Kraut, and Fussel 2004).

## 4.3 Teaching the methods

In the Fall of 2003, we performed an experiment involving teaching our analysis methods to a class composed of twenty-one Master's students and upper-level undergraduates. For the class project, the students worked into groups of two to four; each group created problems for pairs of subjects to solve cooperatively using the GrewpTool, a groupware framework similar to GHT. Students were asked to submit an initial design based on a survey of their available user population; topics ranged from "plan a 5-night vacation to Boston" to "the wedding dinner planner" to "create a web page describing the culture of a nation." After constructing a prototype of the system, students recruited three or four pairs of subjects, trained them in use of the system, and generated about 10 total hours of usage data. From this set of data the students were asked to select a single transcript and apply the methods presented in this paper to analyze the interaction.

Along the way, a number of methodological concerns were raised, which we will discuss here before moving on to presenting the results of the experiment.

**Level of detail** Every reference in the discourse points to some referent that can be tracked. However, the analyst must choose a level of detail for the analysis that balances desired fidelity and efficiency — tracking every referent may result in improved accuracy, but can be extremely time-consuming. Choosing a level of detail is an interactive process. By iteratively increasing the level of analysis detail, analysts can perform a high-level investigation of the discourse to get an idea of general topic, and then return to the discourse to retrieve more detailed results. Likewise, an analyst may decide to pull focus back to a more general view if the analysis is proving overly laborious for the quality of results returned. Other methods, such as recurrence analysis, can be used to highlight areas to concentrate on. There is a distinct cost/benefit trade-off to be considered — while a thorough analysis of the entire dataset may provide superior results, the time investment for such an effort may outweigh the benefits.

**Referent types** During analysis, each referent is assigned a type. This allows investigation of the differences between various sorts of information. Given a set of collaborative tasks, the hypothesis embedded in the method is that the set of referent types discovered will reflect the sorts of information that participants, and help understand how they organize their activity. These types are a combination of domain-specific referent types — in VesselWorld, these types included waste, vessel, location, and barge — and domain-independent types, including plan and repair.

Identifying an appropriate set of referent types for a particular interaction is an important part of the analysis. The process of determining types provides significant insight into the sorts of information that participants exchange. Observation and definition of types are intertwined; two closely-related types that turn out to be handled the same way may be better represented as a single type, whereas a class of information whose use can be split into two or more distinct usage patterns may need to be reclassified as being made up of two or more types. As analysis progresses the analyst will generally have to iteratively reassess the choice of types until an acceptable set of types is derived for the specific goals of the analysis.

## 4.4 Reproducing results

After this practice applying the methods, students were asked to perform a referential structure analysis of four standard transcripts of GHT data to test their analytic skills. Parts of the GHT study had been discussed in class on several occasions, so the students were familiar with the domain, although they had not seen the specific data they were given. After the analyses were performed, we engaged the class in a discussion of the results and methods from this analysis, which yielded strong positive feedback about the utility of the method. In addition to providing students with unambiguous feedback about their ability to perform the analysis correctly, this exercise allowed us to test the inter-coder reliability of the methods.

Each transcript was analyzed by five pairs of students. The resulting analyses were qualitatively similar, though there were minor variations in results from group to group. To quantify the agreement, we used Cohen's Kappa (Cohen 1960), a standard method for comparing two or more analyses of a single set of data. It is meant to be applied to a situation where independent analysts are sorting items into one of a number of categories. It computes the probability that the two classifications differ from chance, and is expressed as the positive probability that the two analyses are identical.

There were some significant complications in applying Cohen's Kappa to our data because the task was not a strict category-assignment task. To apply it, we took every pair of groups for each transcript and compared their analyses. For items, we used the referent clusters described above, assigning each group a 1 for each cluster if they had tagged a referent belonging to that cluster, or a 0 otherwise. Comparing these vectors of cluster membership resulted in a Kappa rating for each pair of groups on a particular transcript.

Average kappa values for each group ranged from 48% to 71% — fair to moderate agreement — with an overall average of 62%. Given that the student groups were given

| Project | Recurring coordination | Recurring errors | Secondary structure | Referent types | Statistical measures |
|---|---|---|---|---|---|
| Class web page | x | | | x | |
| Collaborative coding | x | x | | | |
| Boston adventure | x | x | | | |
| Collaborative coding | x | x | | | |
| Country web page | x | x | | | |
| Social dinner | x | x | | | |
| Trip planner | x | x | x | | |
| Themed web page | x | x | | x | x |
| Wedding dinner | x | x | | x | x |
| Boston trip | x | x | x | x | x |

**Figure 12: Rationales offered by student groups for redesign**

incomplete transcripts, were novice analysts, and were presented the analysis task as a homework assignment with unclear goals, we feel that these results are encouraging. We feel confident that in application by more experienced analysts, and with a more thoroughly specified goal for the analysis, analysts would be able to converge on roughly similar levels of detail and tag the same set of referents, improving the strength of the result.

## 4.5 Using the methods for redesign

Students were given three weeks to generate and submit designs for new representations to improve user performance in their particular domain, with the requirement that these new designs be properly motivated using the analysis techniques discussed in class. Most groups were able to successfully apply our methods to suggest interesting redesign possibilities for their systems. Because of time concerns students were not required to implement their redesigns.

Every student group that submitted a redesign was able to successfully motivate that redesign using these methods, as summarized in Figure 12. In the next few sections we will examine these results in greater detail.

### 4.5.1 Using recurrence analysis for redesign

Every group found recurring patterns of coordination and recurring errors in the interaction, and used these observations to justify and shape their redesign. In some groups the students also identified the creation of secondary structure by the users.

**Recurring discourse** All ten groups were able to identify recurring communication about coordination in the data, and used it to justify redesign. The recurring situations identified centered around the heart of the interaction in each case. For example, in the "wedding planner" system, the students noted users spent a great deal of time discussing seating arrangements. This focused their later referential structure analysis, and led them to create representations for determining with seating charts.

**Recurring errors** Almost all groups used the appearance of recurring errors as design justifications. For the student groups, this indicator provided some of the richest data. Despite the overall paucity of data, users made many mistakes that indicated that the system required improvement. For example, the subjects in one group were asked to plan a road trip from Boston to Los Angeles. They often made errors related to problems with attention; that is, one user would enter something into the shared text area, but the other user would fail to notice, and instead duplicate the efforts of the first user. As a result the designers proposed a representation that would allow users to keep track of what task each user was working on.

**Secondary structure** The appearance of secondary structure in the data was less frequently investigated by the students — only two groups justified their redesign based on the appearance of such structure. This is in accordance with our expectations. Because of the relatively small data set collected by the students — only ten hours of data, with each group only using the tool for a few hours — there is little time for the subjects to generate useful secondary structure. In addition, significant sophistication on the part of the analyst is required to spot small-scale, procedure structure such as adjacency pairs.

The structure found by the students is nevertheless compelling. For example, in the "Boston trip" group, one of the subjects ended up filling the shared text editor pane with a highly-formatted itinerary. The subjects felt the need to create a shared representation to organize their activity; however, the tools at their disposal were minimal — only shared text editor — and so they were unable to generate a truly effective representation. The redesign for this domain addressed this and other problems by including a tabular shared itinerary representation similar to the Object List.

### 4.5.2 Using referential structure analysis for redesign

About half of the student groups were able to further refine these design ideas by pulling inferences from the referential structure analysis of their data by making assumptions based on the referent types they identified. Most of these groups employed the full method, computing and comparing various measures (such as referent lifetimes and density) derived from their data. The students made a slightly different use of the referential structure analysis than anticipated. Only half of the groups made use of the referential structure analysis in justifying their redesign. However, all of these groups used the regimen of identifying new referent types as a way to discover the information that participants discussed most frequently in their domain. Armed with this knowledge they produced designs that incorporated shared, structured external representations for these kinds of information. These new types and new representations are summarized in Figure 13.

| Project domain | New types | New representations |
|---|---|---|
| Class web page | webpage | Browser history |
| Boston Trip | event<br>location<br>price | To-do list<br>Itinerary<br>Budget calculator |
| Themed web page | requirement<br>topic | Requirement list<br>Topic list |
| Wedding dinner | constraint<br>food<br>guest | Seating Chart<br>Menu Planner<br>Guest List |
| Trip planner | event<br>time | Timeline |

**Figure 13: Students designed new representations based on finding new referent types**

Three groups drew conclusions based on the statistical analysis of referent data – that is, computing lifetime, density, and so forth. We attribute this low number to a variety of causes. Most importantly, the students were only required to perform full referential structure analyses on a subset of their complete data, and so had a relatively small data set from which to draw conclusions. Hence, whatever data they did have was likely quite noisy, making it hard to draw conclusions from. In addition to this, students who were able to come up with a plausible redesign using the easier methods shown above were unlikely to then continue on to perform a detailed statistical analysis — they felt it simply unnecessary. This was likely due both to time constraints and to the relative simplicity of the domains being investigated. However, the groups that did perform the full analysis produced designs which were of higher quality and which more accurately reflected the coordination problems seen in their data.

These latter groups were able to focus their attention on the more important referent types, and were also able to design systems that more closely matched the access patterns of the information they encoded. For example, the "wedding dinner" group examined closely the conversations their users were having while planning the (theoretical) dinner. They found exchanges about budget to be a frequent occurrence, with many brief references to referents they decided to categorize into a budget "constraint" type. From these insights they were able to design a shared representation — a budget calculator — that they felt matched the characteristics of their reference data.

### 4.5.3 Comparison to previous classes

The success stories from the Fall 2003 HCI class stands in contrast to results from prior sessions of the same class. In the Spring of 1999, we taught a similar class and asked students to complete a very similar assignment; students were given five weeks to design and implement a basic groupware system, demonstrate it to users, and redesign it based on user feedback. Students were exposed to a variety of design techniques, including GOMS analysis, a thorough treatment of distributed cognition, and coverage of Shneiderman's (1997) prescriptions for software design. In this class, only four of eleven groups were able to successfully complete the assignment. While most groups created a system, only a handful made extensive use of feedback from users to redesign the system.

In the Fall of 2000, an HCI class was also given a similar assignment; this time they were supplied with a basic chat system on which to base their designs. Again, students were given a little over a month to design and implement a system, collect user feedback, and redesign the system. Results were similar; only about one-third of groups were able to successfully defend their redesign decisions based on analysis of their data, and their design results suffered from a lack of connection to the needs of their user population. In contrast, the methods presented in this paper aided students in investigating and redesigning their domains because of the structure they provided. By telling these novice analysts what to look at and what to look for, and then what to do with it, the methods helped guide analysts through the design process.

# 5 Conclusions

We have presented a pair of analytic tools which offer complementary approaches to extracting information about group behavior from a transcript of an online interaction. The first approach focuses on recurring communication and errors generated by participants, and the conversational structure they create to structure their activity. The second approach examines the references that participants make to reveal the way information is passed around and stored in representations. Using these analytic tools, an analyst can form conclusions about how to improve a system to better suit the emergent work practice of the participants.

The methods we have presented examine the normal task interaction of the participants in a non-invasive fashion. By focusing on an external, easily-observable feature of the interaction (in this case the discourse) the methods model the interaction while avoiding tricky assumptions about the nature of cognition. Because these techniques mark up and collect individual events within the transcript into a statistical analysis of the data, they can be used to draw quantitative conclusions about an otherwise very qualitative problem. Being able to show that the introduction of a new representation reduces communication by a statistically significant margin allows the analyst to provide compelling evidence of the utility of a redesign. The discourse-based methods of interaction analysis presented in this paper not only identify problematic areas of coordination and recommend ways to redesign a system, but also provide a framework for measuring whether the redesigned interaction is significant. By reapplying the methodology after redesign, the analyst can generate conclusive evidence of the effect the redesign is having on the resulting interaction.

The methods are teachable and reproducible: students were able to successfully apply the methods in the course of analyzing their own problem domains. The methodology provided the students with a step-by-step procedure to use to refine their applications, giving them guidance as to what portions of the interaction to address. When applied to standard transcripts, students were able to come up with comparable results. Finally, the students demonstrated that the methods can be applied to and generate redesign recommendations for a wide variety of domains.

# NOTES

*Acknowledgments.* Thanks goes to Seth Landsman, Josh Introne, and others for comments and continued collaboration in support of this paper. Special thanks also goes to the Fall 2003 class of CoSi111a for providing extensive feedback on these methods.

*Support.* This research was supported by the Office of Naval Research under grants No. N00014-96-1-0440 and N66001-00-1-8965. Additional support came from NSF grant EIA-0082393.

# REFERENCES

Alterman, R., & Garland, A. (2001). Convention in joint activity. Cognitive Science, 25, 4.

Alterman, R., Feinman, A., Introne, J., & Landsman, S. (2001). Coordinating representations in computer-mediated joint activities. Proceedings of 23rd Annual Conference of the Cognitive Science Society. Hillsdale, NJ: Lawrence Erlbaum Associates.

Alterman, R. (2005). Redesigning the representation system. In Press.

Basu, A., & Blanning, R. (2000). A formal approach to workflow analysis. Information Systems Research, 11 (1), 17-36.

Bødker, S. (1990). Through the interface — a human activity approach to user interface design. Hillsdale, NJ: Lawrence Erlbaum Associates.

Borchers, J. (2001). A pattern approach to interaction design. New York: John Wiley & Sons.

Card, S., Moran, T., & Newell, A. (1983). The psychology of human-computer interaction. Hillsdale, NJ: Erlbaum.

Clark, H. (1996). Using language. New York: Cambridge University Press.

Clark, H., & Brennan, S. (1991). Grounding in communication. In J. Levine, L.B. Resnik, & S.D. Teasley (Eds.), Perspectives on Socially Shared Cognition (127-149). New York: American Psychological Association.

Clark, H. & Wilkes-Gibbs, D. (1986). Referring as a collaborative process. Cognition, 22, 1-39.

Cohen, J. (1960). A coefficient of agreement for nominal scales. Educational and Psychological measurements, 20, 37-46.

Cole, M., & Engeström, Y. (1993). A cultural-historical approach to distributed cognition. In G. Solomon (Ed.), Distributed Cognition: Psychological and Educational Considerations, pp. 1–46. New York: Cambridge University Press.

Ellis, C., Gibbs, S., & Rein, G. (1991). Groupware: some issues and experiences. Communications of the ACM, 34, pages 38-58.

Feinman, A., & Alterman, R. (2003). Discourse analysis techniques for modeling group interaction. Proceedings of the Ninth International Conference on User Modeling, 228-237. New York: Springer-Verlag.

Foster, G., & Stefik, M. (1986) Cognoter: theory and practice of a colaborative tool. In Proceedings of the 1986 ACM conference on Computer-supported cooperative work. Austin, Texas: ACM Press, 7-15.

Garfinkel, H. (1967). Studies in ethnomethodology. Englewood Cliffs, NJ: Prentice-Hall.

Gergle, D. Kraut, R. E., Fussell, S. R. (2004). Communicating with Action. CSCW'04: Proceedings of the ACM Conference on Computer Supported Cooperative Work, 487-496. New York: ACM Press.

Goodman, B., Linton, F., Gaimari, R., Hitzeman, J., Ross, H., & Zarrella, G. (2005). Using Dialogue Fetures to Predict Trouble During Collaborative Learning. To appear in User Modeling and User-adapted Interaction.

Grosz, B., Joshi, A.,& Weinstein, S. (1995). Centering: A framework for modeling the local coherence of discourse. Computational Linguistics, 2 (21), 203-225.

Grosz, B. & Kraus, S. (1996). Collaborative plans for complex group action. Artificial Intelligence, 86, 269-357.

Grosz, B., & Sidner, C. (1986). Attention, Intentions, and the Structure of Discourse. Computational Linguistics, 12 (3).

Hirst, G. (1981). Anaphora in natural language understanding. New York: Springer-Verlag.

Hollan, J., Hutchins, E., & Kirsh, D. (2000). Distributed cognition: toward a new foundation for humancomputer interaction research. ACM Transactions on Computer-Human Interaction, 7(2), 174-193.

Hutchins, E. (1995a). How a cockpit remembers its speeds. Cognitive Science, 19, 265-288.

Hutchins, E. (1995b). Cognition in the wild. Cambridge, MA: MIT Press.

John, B., & Kieras, D. (1996). Using GOMS for user interface design and evaluation: which technique? ACM Transactions on Computer-Human Interaction, 3 (4), 287-319.

John, B., & Kieras, D. (1996). The GOMS family of user interface analysis techniques: comparison and contrast. ACM Transactions on Computer-Human Interaction, 3, 320-351.

Kirwan, B. & Ainsworth, L.K. (Eds.) (1992). A Guide to Task Analysis. London: Taylor and Francis.

Kraut, R., Fussell, S., & Siegel, J. (2003) Visual information as a conversational resource in collaborative physical tasks. Human-Computer Interaction, 18, 13-49.

Kuutti, K. (1996). Activity Theory as a Potential Framework for Human-Computer Interaction Research. In Nardi, B. (Ed.) Context and Consciousness: Activity Theory and Human-Computer Interaction. Cambridge, MA: The MIT Press.

Landsman, S. (2006). Building groupware on THYME. PhD Thesis, forthcoming.

Landsman, S., & Alterman, R. (2002). Analyzing usage of groupware. (Technical Report CS-02-230). Waltham, MA: Brandeis University.

Landsman, S., & Alterman, R. (2003). Building groupware on THYME. (Technical Report CS-03-234). Waltham, MA: Brandeis University.

Langton, J., Hickey, T., & Alterman, R. (2004). Integrating tools and resources: a case study in building educational groupware for collaborative programming. The Journal of Computing Sciences in Colleges, 19(5), 140-153.

Leont'ev, A. (1978). Activity, consciousness, and personality. Englewood Cliffs, NJ: Prentice-Hall.

Lockman, A., & Klappholz, A. (1978). Toward a procedural model of contextual reference solution. Discourse Processes, 3, 25-71

Norman, D. (1991). Cognitive artifacts. In J. M. Carroll (Ed.), Designing interaction: Psychology at the human-computer interface. New York: Cambridge University Press.

Olson, J., Olson, G., Storrosten, M., Carter, M. (1992). How a group-editor changes the character of a design meeting as well as its outcome. Proceedings of ACM CSCW'92, 91-98.

Perkins, D. N. (1993). Person-plus: a distributed view of thinking and learning. In Salomon, G. (Ed.), Distributed cognitions: Psychological and educational considerations, 88-110. New York: Cambridge University Press.

Pinelle, D., Gutwin, C., & Greenberg, S. (2003) Task analysis for groupware usability evaluation: Modeling shared-workspace tasks with the mechanics of collaboration. ACM Transactions on Computer-Human Interactions, 10(4): 281-311

Rasmussen, J. (1986). Information processing and human-machine interaction: an approach to cognitive engineering. New York: North-Holland.

Sacks, H. (1992). Lectures on conversation. Oxford: Basil Blackwell.

Sacks, H., Schegloff, E., & Jefferson, G. (1974). A simplest systematics for the organisation of turn-taking for conversation. Language, 50, 696-735.

Schegloff, E. (1991). Conversation analysis and socially shared cognition. In Levine, J., Resnik, L.B., & Teasley, S.D., (Eds.), Perspectives on socially shared cognition, 150-171. New York: American Psychological Association.

Schegloff, E., & Sacks, H. (1973). Opening up closings. Semiotica, 7, 289-327.

Schmidt, K., & Wagner, I. (2002). Coordinative artifacts in architectural practice. In Blay-Fornarino, M., et al. (Eds.): Cooperative systems design: a challenge of the mobility age, Proceedings of the Fifth International Conference on the Design of Cooperative Systems (COOP 2002), 257-274.

Schank, R. & Abelson, R. (1977). Scripts, plans, goals and understanding: an inquiry into human knowledge structures. Hillsdale, NJ: Erlbaum.

Shneiderman, B. (1997) Designing the User Interface, Third Edition. Reading, MA: Addison-Wesley Publishing Company.

Sierhuis, M. & Clancey. W. (2002). Modeling and simulating work practice: a method for work systems design. IEEE Intelligent Systems, 17(5):32-41.

Suchman, L. (1987). Plans and Situated Actions: the problem of human-machine communication. Cambridge: Cambridge University Press.

Suchman, L. & Trigg, R. (1991). Understanding practice: video as a medium for reflection and design. In Greenbaum, J., & Kyng, M. (Eds.), Design at Work, 65-90, Hillsdale, N.J.: Erlbaum.

Suchman, L., & Trigg, R. (1993) Artificial intelligence as craftwork. In Chaiklin, S. and Lave, J. (Eds.), Understanding Practice: Perspectives on Activity and Context, 144-178. New York: Cambridge University Press.

Vicente, K. (1999). Cognitive work analysis: toward safe, productive, and healthy computer-based work. Mahwah, NJ: Lawrence Erlbaum & Associates.

van Deemter, K., & Kibble, R. (1999) What is coreference and what should coreference annotation be? Proceedings of the ACL99 Workshop on Coreference and its applications, 90-96. College Park, MD.