Research in Engineering Design (1999)11:133–144

Research in
Engineering
Design

Developing Strategic Alliances: A Framework for Collaborative Negotiation in Design

Beth Adelson

Department of Psychology, Rutgers University, NJ, USA

Abstract. *This work applies a theory-based framework of collaborative negotiation to some of the disputes that regularly arise during group design. Although the framework was developed to provide general support for group work, this paper focuses on its use as a design tool. The framework, embodied in our system NegotiationLens, has four facets. It:*

1. *Provides a negotiation method intended to produce gain for all parties.*
2. *Provides an efficient process for conflict resolution.*
3. *Develops working alliances.*
4. *Lets parties decide quickly when they should go their separate ways.*

The framework produces the above results by:

- *Helping parties develop well-reasoned and clearly articulated points of view (Adelson and Jordan, 1991; Conklin and Yakemovic, 1991; Conklin and Begeman, 1988; MacLean et al*., 1991).
- *Creating a context of committment and respect.*
- *Moving negotiating parties away from an adversarial stance and into a collaboration.*
- *Allowing joint construction of solutions that are more beneficial than the unilateral solutions each party initially brought to the table.*

We present our framework for collaborative negotiation, describe NegotiationLens, and present two cases in which it was used. We present a third case, a large design project with recurrent design conflicts, and argue how NegotiationLens could have been of benefit there.

Keywords: CSCW; Design; Group work; HCI; Negotiation; Work process

Correspondence and offprint requests to: Professor B. Adelson, 135 Elfreth's Alley, Philadelphia, PA 19106, USA. Email: adelson@cs.rutgers.edu

1. Introduction

This work applies a theory-based framework of collaborative negotiation to some of the disputes that regularly arise during design. Although the framework was developed to provide general support for group work, this paper focuses on the framework as a design tool. The framework, embodied in our system NegotiationLens, is implemented by providing a conceptual organization for all the data on the table, appropriate work spaces and information comparison tools. The work spaces allow easy comparison of various pieces of information that are relevant now, while holding separate information that would muddy things now, but be critical later. The information comparison tools allow negotiators to explore non-obvious solutions and to make better use of available resources.

1.1. Effects of Group Process on Technical Considerations

In our practice and teaching, we have found that group design has certain recurrent, non-technical conflicts that have powerful effects on technical decisions (Adelson and Jordan, 1991; Crowston et al., 1988; Lai et al., 1989; Crowston, 1990; Lee and Malone, 1988; Grudin, 1991). As we will see in the three case studies presented in this paper, the following sorts of work process issues can underlie unarticulated but strongly held positions:

- *Goal Selection.*
  Goal selection problems can arise when two groups with differing roles and therefore differing agendas, work jointly on a project within an organization. In these situations, the parties need to construct goals that are technically and professionally beneficial.

As an example[1], a well-established academic researcher took a sabbatical at an industrial lab to share research interests. The researcher brought along three graduate research assistants to work with her. Work was started on a first project, but a disagreement arose when she told the manager of the group she was visiting that she wanted to start up two new projects. The manager replied that she should not do so until she had completed the one project she had already started.

Behind each position lay a set of legitimate concerns. However, neither side felt able to express them in the context of the tension which suddenly sprang up as a result of their opposing positions. The researcher's goal was to finish, by previously established deadlines, a set of talks on works-in-progress. She needed to start the new projects in order to meet the talk deadlines. Further, she had handed the first project off to her research assistants, and she intended to keep working while she waited for their results. On the other side, the group manager was worried that the researcher's desire to start new projects signaled a loss of interest on the part of the researcher, and so he was worried that his goals would not be met: the current project would not be finished; the work done so far would be lost; and the time invested in training the researcher to use the lab's facilities would turn out to have been wasted. In his experience, some visitors were very productive, although others left projects unfinished, and he found it hard to tell into which category this visitor would ultimately fall.

The relationship which previously had been strongly positive became tense, and work slowed while the researcher considered how to get her work done and maintain the working relationship.

- *Goal Conflicts.*
  These can occur when two groups with substantially different mandates/organizational roles have been asked to work collaboratively. In these cases, the differences between the groups' goals can be sufficient to cause disagreements about design decisions. Goal conflicts also arise when either groups or individuals feel the need to establish and/or maintain areas of responsibility and control, either for personal or professional reasons.
- *Role Conflicts.*
  These are related to the issue of goal conflicts, in that these disputes concern issues of who is best qualified to make certain decisions or carry out specific tasks.
- *Allocation of Limited Resources.*
  This includes problems concerning scheduling and management of time and financial constraints.

In what follows, we hope to show how our framework and negotiation tool can be used to help designers benefit from their design goals or interests. In doing so, we will begin with a discussion of the framework on which NegotiationLens was based. This will be followed by a description of NegotiationLens, along with two examples of its use. Finally, we present a case study of a large design project and discuss ways in which the case could have benefited from the use of the negotiation tool. As we will see, the project initially showed real promise, but then repeatedly fell behind schedule, elements of the system specification were not implemented, and ultimately the project was put aside. One reason for the setbacks was that the project suffered from both technical and non-technical inter-group conflicts. We will argue that the conflicts are both typical in design situations and could have been resolved by the type of framework which we describe here.

2. Collaborative Negotiation: A Framework and a Method

In this section we present a framework useful for negotiating conflicts in large collaborative design projects. Use of the framework is intended to result in a gain for all parties whether they decide to create a strategic alliance or go their separate ways. Additionally, the framework is designed to develop working relationships in either case.

The framework helps attain good negotiation results by moving the diverse groups typically involved in a large design project, away from an adversarial stance and back into a collaborative relationship. This is accomplished by creating a context in which the groups present well-reasoned and fair considerations. The fairness of the views enables each party to be heard, and thus to feel respected. This supports the joint construction of mutually beneficial solutions and a commitment to implementation.

Our negotiation framework stems from both the theoretical and empirical work on negotiation of the last decade, as well as from our own practice and teaching (Susskind and Cruikshank, 1987; Brockner and Rubin, 1985; Pruitt and Rubin, 1986; Fisher and

[1]Here and below, the names or task domain of the examples have been changed to ensure the anonymity of the participants.

Uri, 1981; Fisher and Brown, 1988; Kolb, 1983; Adelson et al., 1991).

We begin with a description of the framework. We then present the tool which grows out of the framework by providing two examples of its use.

2.1. Steps Taken Privately

2.1.1. How the parties currently see things
The process for understanding the current state includes:

(a) Allowing all groups to state the problem as they see it. Each party privately creates their description of the impasse.
(b) Stating the solution they have in mind. Once they are clear on the reason for the impasse, each party then describes what they believe is the right solution to the impasse.

2.1.2. Purpose
Giving a concrete voice to their views allows the parties to back off of feeling being bull-dozed. Rather, each party can begin to feel that they have a clearly articulated view that will be heard and respected during the negotiation process.

What motivated the current view
(a) Finding underlying interests and resources. The parties then use their initial solution to draw up a list of design considerations or interests which need to be included. Items on the list are then prioritized. The parties also list what they are contributing to the design.
(b) Developing objective criteria for stated interests. The parties then reflect on the roots of their interests, in an attempt to see if they are well-founded. They check the validity of each interest by finding legitimating criteria. This often involves finding relevant and well-known standards or practices. This is done not to create a defensive air or to try to force an unrealistic rationality. Rather, it helps the negotiators come to the table with views that are hard to discount and can be incorporated in a mutual gain solution. For example, if I go into a negotiation over the sale of my house I will look foolish if I ask for $500,000 when the going price in my neighborhood is $250,000. If I come in and ask for $265,000 because I've recently put in a hot tub and sauna, I have given us something to talk about.

2.1.3. Purpose
Listing interests and resources allows the parties to put aside adversarial positions, and instead to come to the negotiating table with action items and potential solution elements. Having developed objective criteria, the parties can feel that they are presenting and being presented with well-motivated rather than arbitrary lists of concerns.

This piece of the framework continues to create a sense of a fair process. As this happens the two parties can increase their inclination to explore and generate new and creative solution elements. And the exploration can lead to increasingly mutual beneficial solutions.

Note that this process is designed to be *efficient*. Other parties have no ability to engage in a prolonged bargaining-down process if I have established objective criteria and insist that they do the same.

2.1.4. Difficulties
Parties may come up with multiple or conflicting criteria for a given interest. The parties then have to decide which to accept. Here the framework suggests taking the recursive route; this new issue can again be resolved through the application of objective criteria. In fact, this happens all the time, in negotiating the price for a used car a dealer will hold up his Blue Book while the buyer carries his Consumer's Report. Here the parties have to decide on the appropriateness of each standard in determining price.

Note the process does *not* cause this sort of secondary disagreement, rather it provides an efficient process for its resolution.

2.2. A Collaborative Step

2.2.1. Developing a collaborative solution
Having established their interest and resource lists, the parties now have several techniques available which contribute to the development of a joint solution: They may revisit their interests to create solutions in which both sides are satisfied. They may look at their interests and resources to find out how the interests of one can be met by the resources of the other. For example, I need you to develop an elegant package and you need me for a graphic design. Additionally, the parties can look for ways in which resources can be combined to obtain leveraged joint solutions.

Importantly, this process is applied both iteratively and jointly. As a first step, many solutions are created. As a second step, the best elements are refined and

then combined. I can make pasta with pesto. I can make couscous. Or I can make couscous with a pistachio pesto.

2.2.2. Purpose

This process has similarity to brainstorming in that both parties can produce unedited and therefore perhaps novel ideas. That this is done jointly, allows the parties to take advantage of each other's expertise. It further allows the parties to see each others' views without making any decisions, and so may increase mutual understanding which can strengthen the collaborative process. The last stage in which parties jointly draw on their interests, resources and candidate solutions both can produce solutions which are better than either party expected and allow the parties to again become a single team.

2.2.3. Difficulties

If the antipathy between the parties is long-standing either party may slip into an adversarial stance. Parties who have experience with the method can reframe the other side's emotionally charged statements as straightforward points to be considered. 'You never look at side effects!' can be met with the reply, 'Let's list the side effects that could give rise to trouble.'

If adversarial moves are not caught quickly, they can turn into power plays in which, for example, one side demands that they be taken at their word, while insisting that the other side provide documentation for their statements. Again, the method can be applied recursively. The party on the receiving end can insist that the framework be used to quickly establish a fair working process.

Current negotiation theory stresses the importance of having parties look for ways in which the resources of one side match the interests of the other. However, the discovery of these opportunities has been found to be difficult. It is hard for a group to feel, and therefore to see, how they might benefit from another group with whom they currently are at serious odds. However, our framework is designed to decrease this problem, the parties systematically match listed interests against listed resources. This brings us to the value added by NegotiationLens. As we will see in the following section, in implementing our method, NegotiationLens keeps parties focussed on their negotiation goals. Resources have been listed, they are visible and have been committed. Needs are listed in the same place. The negotiators have before them the concrete task of matching needs to resources.

This brings us to another difficulty. Setting aside interpersonal issues, in a complex negotiation it is just plain hard to see how interests interlock, how resources meet interests or how resources can be combined. Again, NegotiationLens adds value. It organizes potentially large amounts of data. It keeps some things separate and joins others; while joint solutions are being generated needs and resources are listed but initial solutions are put aside. Additionally, NegotiationLens does this automatically for the parties, they do not have to wrestle with markers and flip charts. Another advantage of NegotiationLens over paper is that it expects parties will want to look at an impasse in various ways. NegotiationLens has features that support exploration of interests and resources and their various combinations. We would not think of developing a complex budget needing comparisons and revisions without a spreadsheet. Similarly, there is no need to conduct negotiations without appropriate computational support.

2.3. Private Steps

2.3.3. Developing alternatives

In addition to developing possible negotiated solutions, the parties are asked to individually examine their alternatives to working together. This serves several functions. When good alternatives to working together do exist, the members of one party will not feel pressured and therefore may feel an increased desire to work with the other side. However, if both sides do have better alternatives they may quickly decide that the current collaboration should be abandoned. When made early on, this decision often preserves the collaborative relationship, allowing future joint efforts to succeed.

In the case where good alternatives are not found, it can increase the parties committment to the negotiation process, thereby motivating the parties to construct a joint solution.

2.3.4. Evaluating the joint solution

In this final step, the parties review their joint solution in light of their interests and resources. They may be satisfied, or they may find they should pursue a resolution with a different set of parties. If their alternatives are not strong, and they are not satisfied they may try to negotiate further. If their alternatives are weak and the other side's are strong, they may decide to go with the joint solution.

3. Using the Framework: Walking Through a NegotiationLens Case Study

In what follows we illustrate our framework by walking the reader through the system as it was used in the visiting researcher goal selection conflict (Section 1.1.). Recall that in this case study, a disagreement arose when the visiting researcher (Rebecca) told the group manager (Jose) that she wanted to start up two new projects, and Jose replied that she should not do so until she had completed the one project she had already started. Eventually, Rebecca and Jose (along with Ursula, the researcher who had been collaborating with Rebecca on the first project and training her to use the lab equipment) decided to use NegotiationLens to work through the dispute.

We begin by giving the reader an overview of the user's experience of NegotiationLens. As to what users see: Fig. 3 shows a window from the 'Visiting Researcher' case study. This window, which lists the interests ('needs') and resources of the parties represents a typical NegotiationLens window. The banner across the top of the window tells the user she is looking at a need and resource list. Below that, there is a menu bar across the top of the window which allows the user to either select the default action associated with each menu name or to pull down the menu and find related actions. Inside the window, the top two rows tell the user what proposed solution is being evaluated with respect to needs and resources. Below this there are two sets of rows, `Needs` and then `Resources`. Looking at needs, to understand each row read across the columns from left to right. First, we see the number of the `Need`, in the next field we see its contents. In the three rightmost columns, we see how well the proposal satisfies the need '`Sat/Util`', who entered the need

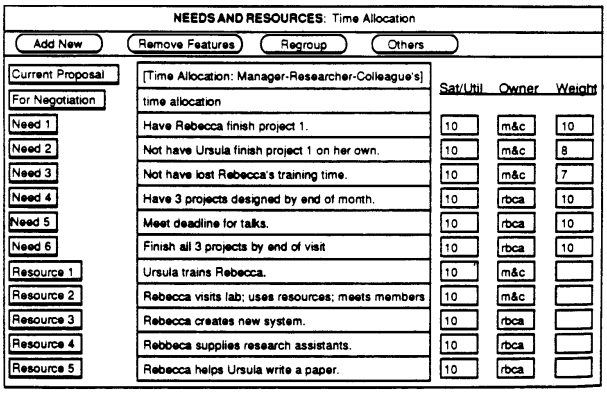

Fig. 1. Needs and resources with decision-making information shown.

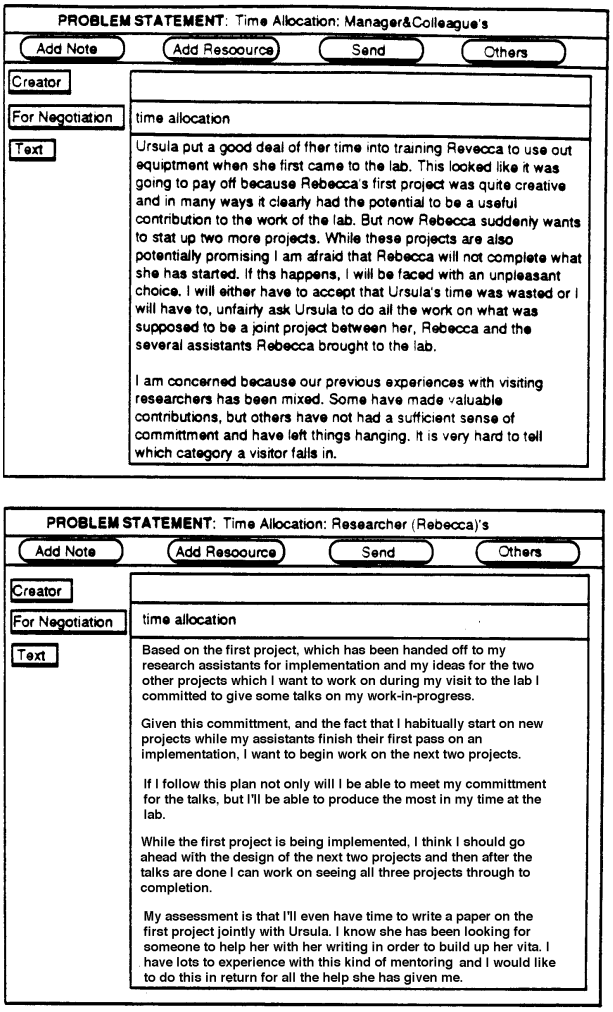

Fig. 2. Manager's (top) and researcher's (bottom) problem statements.

(`Owner`), and how important it is (`Weight`). Resources are interpreted similarly, although the `Sat/Util` column now tells us how well the resource is being utilized.

As to how this window was created: Negotiation-Lens has a 'home' window with the equivalent of a `File` menu allowing users to create `Needs` and `Resources`, `Problem Statement` and `Problem Solution` windows. Through these windows, NegotiationLens allows users to create workspaces as needed and to keep track of all information on the table.

One difficulty with process-support software is that users will not enter information if they feel it wastes their time. However, when working in a `Needs and Resources` window, users can ask the system to transfer information that has already been entered into previously created windows (e.g. from Problem

Statements). Conversely, when working in a `Problem Statement` or `Proposal` window, information can automatically be transferred to `Needs and Resources`. This does cut down on typing, but the general solution to the problem is an open HCI question.

A full description of NegotiationLens is presented by Adelson and Jordan (1991), but it should be noted here that the system is implemented on top of and integrated with Object Lens, a collaborative work system for email, bug reports, writing, etc., developed and then used by Malone and his colleagues (Crowston, Malone and Lin, 1988, Lee and Malone, 1988, 1990; Lai, Malone and Yu, 1989). We gave ourselves this integration constraint, because as Grudin (1988) points out, systems which are not part of daily work practice have little chance of being adopted.

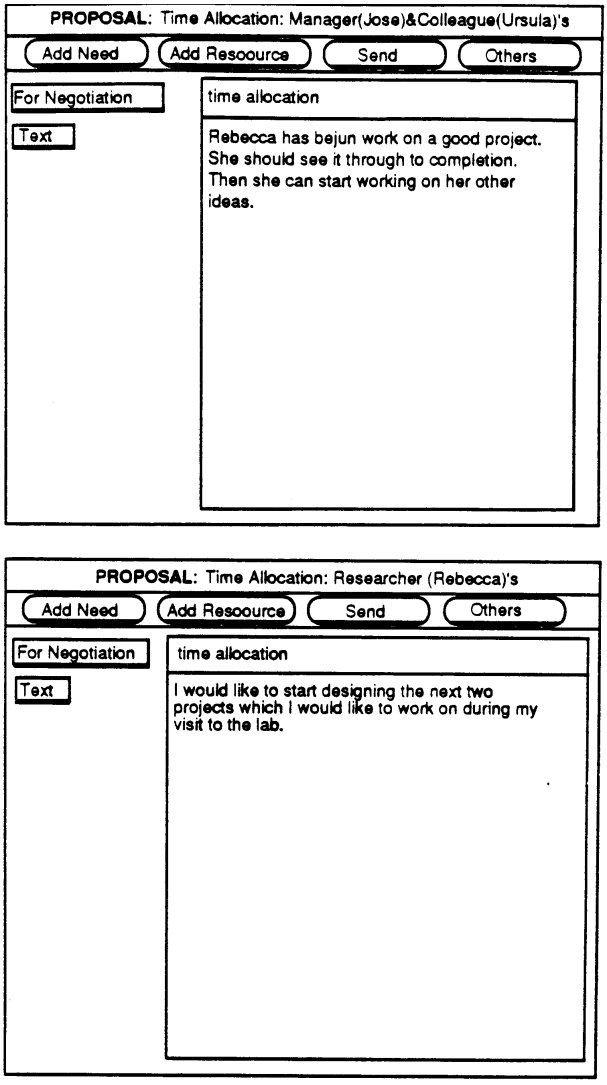

Fig. 3. Manager's (top) and researcher's (bottom) initial solutions.

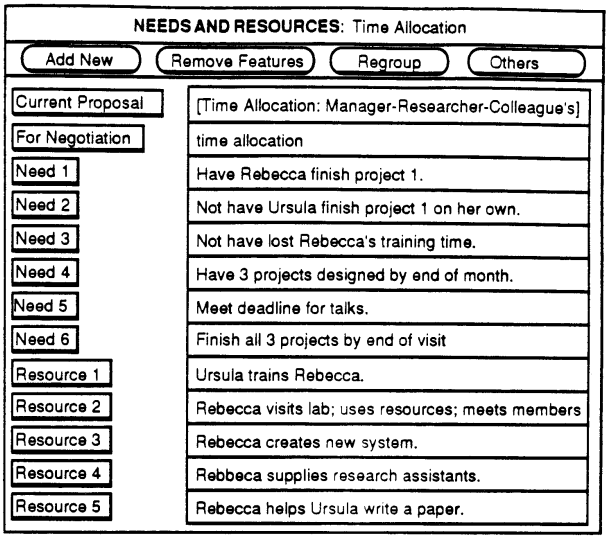

Fig. 4. List of needs and resources.

3.1. Developing a Problem Statement

Using NegotiationLens to create a problem statement, Rebecca was given the opportunity to express her concern over meeting her deadlines, and to state her desire to use her and her assistants' time most efficiently in order to have three projects completed during her visit. Jose and Ursula, in a second problem statement, also had a chance to express their fear that either Rebecca would not finish the project and so the time Ursula had already put in would turn out to have been wasted or alternatively Ursula would be left to finish the project on her own thereby adding to her already considerable workload (Fig. 3).

3.2. Developing an Initial Solution

Both parties then separately proposed an initial solution which was satisfying to their side (Fig. 3). Jose reiterates that he wants serial processing, Rebecca states that she should work in parallel.

3.3. Thinking in a Collaborative Mode

3.3.1. Deriving underlying interests and resources
Backing off from their initial solutions, the parties then turned their attention to their original `Problem Statements`, using the explanations contained there to list their interests and resources (Fig. 2).

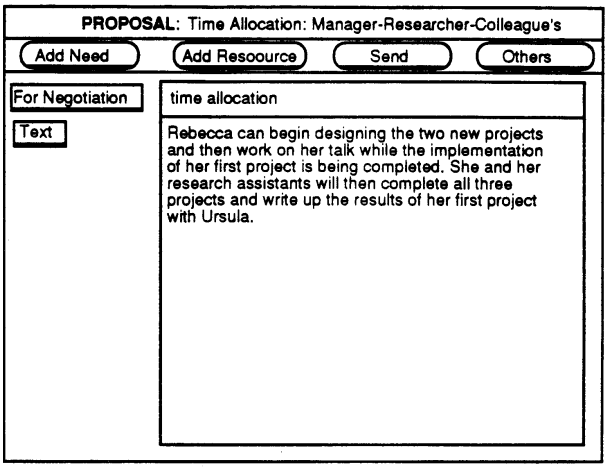

Fig. 5. Initial joint solution.

The two parties also entered a weight for each interest (not shown).

3.3.2. Matching interests to resources

At this point, using the `Needs and Resources` list (Fig. 2), the parties were able to create a joint solution in a newly created `Problem Solution` window (Fig. 2). This solution contained a schedule and work assignments which allowed for the immediate design and eventual implementation of three systems which Rebecca wanted to create for the lab. It also contained a committment to complete all three projects.

This joint solution was constructed by matching each need against the listed resources. As mentioned above, systematic matching is critical both in discovering solutions based on non-obvious ways in which parties can help each other and in increasing a sense of collaboration between the parties. The systematic matching process is facilitated by the system's grouping of needs separately from the resources. (In Fig. 3), we see needs grouped above resources. But see point (4) for a discussion of the `'Regroup'` or sorting feature on the menu bar.)

3.4. Evaluating the Joint Solution

In the final stage the parties considered the goodness of the solution by entering a value indicating the extent to which each need was satisfied and each resource was utilized (Fig. 1, third column from left).

This example turned out to be one which had a simple solution in which there was clearly a mutual gain and so it may not seem surprising that the satisfaction values were high and that both parties felt the initial joint solution was satisfactory. However, it must be stressed that the situation did not appear that way at the *outset* of the negotiation. It started out in a charged atmosphere which followed a period of stalemate and frustration. Because of the explosive atmosphere, it did not have the feeling of a problem which was going to be solved easily. It was only when the parties extracted their needs and resources from their problem statement that the solution presented itself as simple.

Another result of the negotiation was one which is highly desirable to those interested in group work. Each party reached a better understanding of the needs, strengths and concerns of the other, this resulted in a more relaxed group dynamic, and allowed them to avoid future conflicts around these sorts of issues. Additionally, it strengthened the relationship between Rebecca and Ursula in that they agreed to (and did) jointly write a paper on the first project upon its completion.

In the case of more complex negotiations, if the parties feel dissatisfied with a newly developed solution, the system provides them with facilities for finding the source(s) of their dissatisfaction. The parties can turn to the `Needs and Resources` window for the new solution and look at who put forth each need and resource; how important each need was; how well each need is being satisfied by the solution currently under consideration and how well each resource is being utilized. Additionally, selecting the `regroup` option on the menu bar in Fig. 3 allows the parties to request that the needs and resources list be resorted by weight; by weight for each owner; by satisfaction/utilization; or by satisfaction/utilization for each owner.

Several situations can call for sorting and inspecting weights and satisfaction values. For example, if one or both of the parties are not satisfied with a solution, but are not sure why, they can first sort the needs by weight and then inspect the satisfaction values, allowing them to see whether important needs are *both* listed and being met. A new solution can then be developed by revising the needs and/or resources or by making better use of the existing resources.

As a second example, if one party feels that the current solution is more favorable to the other side, it can sort the list by owner and then within that by satisfaction values. The parties can then see if the solution is addressing the interests of both sides. If not, a new solution can then be developed. This can be done by looking at the utilization values for the resources and either revising or making better use of them.

NegotiationLens results in making available the interests and resources of both sides. As a result, it

can lead to the development of solutions which are more satisfying and more sound than unilateral solutions based on unarticulated criteria. Additionally, and for the same reason, the process can move the parties towards a better long term relationship.

3.1. Case Study: Matching Criteria for Success

3.1.1. The New Faculty Member

In this second example, the negotiation resulted in mutual gain as a result of one party reconsidering the legitimacy of part of his rationale. In this negotiation a young researcher (Dennis) was trying to negotiate the terms of a first faculty appointment with the help of a more senior colleague (Karen). In this example NegotiationLens was used as a *planning tool* to help the junior colleague work out his best approach. The second party, Dennis' new department head, Isaac, was not directly involved in the use of the tool, although he was affected by the rethinking that resulted from its use. Initially Dennis wanted Isaac to allow him to buy out of teaching with some research funding he had been offered. More specifically, he wanted Isaac to use the buy out money to bring in one of Dennis' friends to teach his courses. He was particularly eager to have this friend as an intellectual companion in his new job.

Dennis and Karen used the tool to create a problem statement, an initial solution and an interest and resource list both for Dennis and, to the extent possible, for Isaac. Dennis and Karen then reviewed the list in an effort to construct a proposal which would be acceptable to Isaac, since he had initially exhibited resistance to the idea of Dennis' buying out of teaching. Isaac believed that having the faculty teach the students provided the students with the best education. He therefore saw teaching as an important responsibility both to the students and to the university community as a whole.

In reviewing the list of needs it came out that Dennis' rationale for wanting to buy out was that he wanted to do well at his new job and that underlying this rationale was the criterion of doing as much research (and therefore as little teaching) as possible. However, it also became clear in considering the department head's rationale that doing well included being willing (if not down right eager) to teach. This suggested that Dennis should change his criterion as to what constituted doing well at the new job, and as a result, change his stance on the buy out. Having done this, Dennis and Karen then reviewed the extent to which the resources in the situation were being utilized. They noticed that if Dennis did not buy out he could use his research money to bring in his friend as a visiting professor and possibly in the long term change the department head's attitude concerning the potential contribution of visiting faculty.

This process, in which Dennis reviewed the legitimacy of his rationale led to a solution which benefited both parties.

4. Case Study: The Integrated Work Set Project

NegotiationLens was conceived of as a tool to aid negotiation in a wide range of domains. And because design negotiation is driven by a process in which the interests of the parties provide the basis for the resolution of differences we argue that NegotiationLens also applies to the domain of design.

In this section we provide a retrospective account of a large design project and show how the resolutions for the cases above can provide partial models for the conflicts described here.

4.1. Overview

Several years ago, a major software company instituted a project whose goal was to create an integrated set of development tools, the 'IWS', or Integrated Work Set. Two years later, after continual setbacks, the project was ended without the release of the software. At its inception it was envisaged that the IWS would comprise a wide range of applications; ones for which the company was already known, as well as ones to be developed specially for the project. The company was striving for a product in which a graphical interface would provide a vehicle for the easy and integrated use of both familiar and innovative applications.

Upper management therefore gave the project high priority. With input from marketing and high level development management, upper management set down a high level functional specification for the product. It then assembled several groups of leading employees; each described below. As a reflection of the project's high priority the groups were allowed much more say than usual in the part of the design for which they were responsible. They were also given a much looser time schedule than was usual; typically groups were asked only to revise pre-existing products under tightly defined time schedules.

4.2. Groups Involved in the Development of the Integrated Work Set

Responsibilities and conflicts: Not all of the project's design decisions generated conflicts and not all of the conflicts were intractable. In what follows we focus on the difficulties which we believe could have been resolved through the use of the NegotiationLens framework.

4.2.1. User Interaction (UI)

Reflecting the importance of the user's interaction with the system, the UI team was assembled at the beginning of the project from the set of software engineers considered to be the company's 'hot shots'.

The group's responsibilities, fell into two categories. At a high level the group was charged with ensuring that the diverse set of applications would have a unified look and feel, giving the software a global coherence. At the level of implementation the group was responsible for producing the window manager, the graphics utilities, the I/O handlers, etc. The team included among the software engineers one member with extensive training in interface and human-centered system design.

Two of the intra-group conflicts that arose are interesting in looking at the value of a negotiation framework using objective criteria. Both involve situations in which designers supported sub-optimal choices as a result of unarticulated design considerations (Grudin, 1991).

(a) *Grouping similar applications:* the system was slated to contain a set of applications which were similar but not identical and which, in addition, had names that were sufficiently similar that even members of the development group often got confused and launched the wrong application. Because of their confusability the software engineer with the user interface background proposed that this set of applications should be grouped on a single menu. His rationale was that putting the set under one menu would allow users to focus on the differences between the members of the set and as a result confuse them less often.

The proposal met with strong but not clearly explained resistance. Repeated and lengthy discussions within the group did not resolve the issue. Finally one of the newer software engineers confided to the interface designer that the resistance came from the other software engineers' rationale that an interface should reflect its implementation.

The example of the junior faculty member who reviewed his criteria for job success provides us with an analog for looking at this dispute. Had the UI group engaged in a NegotiationLens-like process the interest of wanting interfaces to map on to architectures might have been brought before the whole group. The group could then have decided whether they wanted to retain this constraint in this situation. That is, the interest does have some legitimacy. It makes systems easier to maintain and modify. However, had the tradeoff between maintenance usability been brought out, a decision could have been reached more quickly and with less acrimony.

(b) *Nested menus:* a second conflict concerned whether nested menus should be used. Here the software engineer with the interface background felt that some of the applications would have profited from nested menus. His reasoning was that for applications such as `print`, which had several options, if a user did not want to use the default options, a second menu should unfurl allowing the user to quickly specify the desired options. The counter proposal contained no nested menu and as a result constrained the user to accept a set of default values for the `print` command. The interface person felt that this was particularly problematic since it was not clear to him what the default options should be. (For example, should the action of printing a folder produce a listing of the folder, or a printout of its contents?) Again prolonged arguments ensued and the issue was difficult to resolve. Several years later, looking back on the dispute it became clear that because in its presentation on paper, the design without nested menus looked elegant and was easy to understand the majority of the group assumed that when implemented, it would better suit the needs of the system's users. Again had the criteria for and against the 'no nesting' design been made explicit the difficulties might have been resolved more easily.

4.2.2. Applications Environment (AE)

This group worked most closely with the UI group, having similar concerns and responsibilities. At the level of implementation concerns the AE group was responsible for producing the libraries which would support the specific applications. Related to its implementation level responsibilities, its high level charge was to ensure that the individual applications were supported in a way which was globally consistent both at an implementation level and with respect to look and feel.

4.2.3. Performance Analysis (PA)

The usual responsibility of the Performance Analysis group was to stress newly built systems in order to ensure that they could meet minimum performance

requirements. For this project it was decided that, contrary to the usual arrangement, the PA group would be involved at the outset in order to avoid finding hard to deal with after-the-fact problems such as finding that systems which had been targeted to support 24 PC networks could in practice only support, and be marketed with, 18. The PA analysis group had over time developed many tools for assessing system prototypes and the nature of their input was potentially valuable to the UI and AE groups. However, the performance analysts were accustomed to making assessments once they had a working prototype. In this situation, where they were being asked to produce an evaluation in the absence of a prototype they decided to come up with a minimal list of requirements in the form of response times for basic operations like cut, paste, copy, delete, etc.

When PA presented their proposal for performance requirements to the UI group, the proposal was greeted with resistance which was sufficient to cause the performance analysis group to withdraw from the project. Both groups ultimately lost out in this situation. A successful contribution to the high visibility IWS project would have benefited PA. Additionally, had PA been able to uncover an inadequacy in the performance of the system before it was cast in code it would have been helpful to UI. Looking back to our two earlier case studies, it seems as though NegotiationLens could have helped uncover the sources of friction preventing mutually beneficial solutions.

The resistance of the UI group came from two sources: The performance analysis group had not explained how they had come by the performance requirements in their proposal; the requirements seemed arbitrary. Further, UI felt that the advice was presumptuous, they believed that they were continually trying to optimize their implementations and that the PA group had not spoken to their hard problems. What we see here is a conflict in which each side had something that would have benefited the other but neither side was able to make that clear. As a result the two sides withdrew and lost the opportunity for mutual gain. If we view the visiting researcher and new faculty examples as partial analogs to this situation we can use elements of both negotiations to construct a scenario with a mutually beneficial outcome.

Had the UI group been encouraged to make a needs and resource list along with a set of objective criteria, it could have become clear to PA that they were from the outset aware of optimization considerations. Had the PA group also been encouraged to make the same list UI could have been made aware of the tools PA had for doing evaluations on prototypes. Then, when looking for opportunities for mutual gain, the parties could jointly have made decisions as to how they could best use of each others' resources. Under this scenario it seems that the two sides might have come to an agreement that PA could provide input useful to UI once an early prototype had been built (something which the UI group was able to do). And had such a plan been implemented the work of both groups might have progressed more rapidly and/or effectively.

4.2.4. Technical Writing (TW)

Typically, technical writers only wrote external documentation, the documentation to be seen by the end users, rather than the internal documentation used to communicate detailed functional specifications to the groups involved in a project. This meant that the TWs became involved in projects only when the product was completed and the rush for shipment had begun. It left them very little time to become agile users of the systems they were documenting and so although they were talented writers, they were forced into a position of producing user manuals which only provided a catalog of the system features. It did not allow them to produce a document which explained how to use the system in a full, clever and creative way in a variety of typical scenarios. Additionally, it did not allow the TWs, who constituted the first end users in this off-the-shelf development situation to provide the developers with feedback on the system's usability (Grudin, 1988, 1991).

To remedy these two recurring problems, upper management decided to involve members of the technical writing staff at the outset of the IWS project. To begin with, they assigned one junior technical writer to work under the management of a second senior and particularly talented technical writer. The senior writer had a clear desire to move into a management role. She expected that this situation would provide her with a vehicle to do so both because the project had high visibility and because the TW group was expected to grow with the project.

However, problems arose from this well-intentioned although not thoroughly worked out time line. At the very outset of the project there was not enough documentation work to occupy two full-time writers. As a result, the junior writer went back to the main TW division and the senior writer was assigned to do internal documentation. This meant that her professional goals were being very poorly served causing her eventually to leave the project as well.

It therefore turned out that the situation ended up serving no one's goals. The senior TW did not get the

opportunity to prove herself as a manager; upper management did not manage to get rich external documentation or early user feedback and TW was once again put in the position of having to create documentation after, rather than during development, which meant further delays for the project.

Had UI and the TWs made explicit what they did want from each other, what they did not want and what they were willing to give each other, it might have been possible once again to create a situation from which both the parties and the project would have benefited. This is what we saw in the visiting researcher example, where three systems were developed rather than just one, once the intentions of the parties were examined.

As to the needs, limitations and resources of the groups, UI wanted fuller documentation but felt ambivalent about feedback from TW. They felt that engineers had better understanding of systems, and therefore better intuitions about usability than did TWs. The senior TW also wanted to be able to provide fuller documentation but she did not want to be working alone on internal documentation because it gave her no chance to prove herself as a manager or to produce a piece of work which would have been appreciated in her division which produced and therefore valued only external documentation.

Had these needs and resources been surfaced and systematically considered, a joint solution might have been reached similar to the one we envisaged for the PA and UI groups. The two TWs could have attended some early design meetings and then come on when there was a working prototype. They could then have produced prototype-level documentation and so the final documentation would have been fuller.

However, as to having UI accept the TWs feedback, it is not clear whether the engineers would have come to respect the TWs to an extent which would have allowed them to accept their feedback. Lack of respect and therefore acceptance of non-technical professionals is a difficult issue when it comes to user feedback. On the other hand, if the TWs, as a result of attending early design meetings, had been able to produce prototype documentation which reflected a sufficiently deep understanding of the system, the software engineers might have been willing to accept the TWs as a legitimate population of users and therefore taken account of their experience. Additionally the career goals of the talented TW might have been realized.

4.2.5. Marketing (MK)

Marketing has the central responsibility of maintaining and expanding the company's market share. Marketing attempted to maintain market share by ensuring that new products were backwardly compatible with existing products. They attempted to expand market share by requesting functionalities which competing companies featured in their advertising. This meant that marketing would frequently demand that UI/AE change functional and/or interface elements of a system. Often this would occur at a late date, since that is when the product tended to be evaluated by marketing.

These disputes are inherently difficult to resolve. There are real differences between the goals of the two parties and both are legitimate. Although NegotiationLens may not be able to get the parties to find a mutually pleasing solution it does ease the tension accompanying inter-group conflicts as we saw in the visiting researcher example.

The process fostered by the use of NegotiationLens eases tensions by encouraging a working through of differences within a framework of respect and committment. That is, within this framework the parties make explicit the criteria underlying their interests and provide explanations for the weight which they are giving to their needs. This means that although the parties may still disagree at the end of the process, they have seen that the disagreement is not a result of one side discounting the other, nor of one side being arbitrarily stubborn. We have found that, for example, customers who cannot be accommodated have a better reaction to the news when they know that their needs are not a matter of indifference.

4.2.6. Applications Development (AD)

Once an architecture and a prototype was in place, development groups were assigned to develop each of the set of applications which the IWS was to support. The conflict which arose between this set of groups and UI/AE was quite similar, at an abstract level, to the conflict between marketing and UI. It was in the interest of each of the AD groups to have the interface and the architecture designed in a way which would best support their particular application. However, the goal of the UI and AE groups was to design a system which had both a coherent look and feel and gave sufficient and balanced support to all of the applications. In this case, as in the the case with marketing, the goals of the two parties were in conflict. Again there is no easy solution to this recurrent problem, but a NegotiationLens process could at least have allowed the AD groups to feel that their needs were being seriously heard and considered. It is possible that this would again have helped to decrease the delays which were caused by

prolonged design disputes and which ultimately hampered the project quite seriously.

Looking at the case study presented here, it seems that a collaborative negotiation tool like NegotiationLens may have been able to resolve a number of the conflicts which arose within and between the groups working on the Integrated Work Set project.

5. Summary and Implications

We have presented a framework and a tool which we argue allows designers to uncover and use their concerns in order to resolve a wide variety of design disputes. The tool does so by allowing the designers to freely express their view of the problem and how it should be solved, make their underlying interests explicit, state and review the criteria behind their interests and then jointly create mutually beneficial solutions based on the interests and resources of each side. Additionally, the framework specifies an efficient process in focusing the negotiators on clear and well-reasoned interests, obviating lengthy bargaining sessions.

Of course, there is a cost to negotiating a dispute. In many cases the cost will not be worth the benefit and negotiation should not take place. But in other cases, like some of the ones presented here we believe that it will be clear that negotiation will result in a savings. For example, if a group like Performance Analysis or Technical Writing, whose contribution is both valuable and cannot easily be replaced or offset is withdrawing from the collaboration, entering into a negotiation is worthwhile. Similarly, if a dispute concerning the implementation of a central feature of a system is continuing for a period of time which is equal to the time it would have taken to implement the feature it again seems that a several hour negotiation will produce a savings.

Acknowledgements

This work was supported by SROA awards from Rutgers University and NSF award number HRD990-6200. It was also made possible by the generous support of Tom Malone at The MIT Sloane School Center for Coordination Science. Special thanks go to Troy Jordan.

References

Adelson B, Jordan T (1991) The need for negotiation in cooperative work. In: E Barrett (ed), The Social Creation of Knowledge. MIT Press, Cambridge, MA

Brockner J, Rubin J (1985) Entrapment in Escalating Conflicts. Springer-Verlag, New York

Conklin EJ, Yakemovic KB (1991) A process-oriented approach to design rationale. Human-Computer Interaction 6 (3,4)

Conklin J, Begeman M (1990) gIBIS: A hypertext tool for exploratory policy discussion. In: D. Tatar (ed), Proceedings of the Second Conference on Computer-Supported Co-operative Work. ACM Press

Crowston K (1990) Towards a Coordination Cookbook: Recipes for Multi-Agent Action. Doctoral Dissertation, MIT Sloan School

Crowston K, Malone T, Lin F (1988) Cognitive science and organizational design. Human Computer Interaction 3:59–85

Fisher R, Uri W (1981) Getting to Yes. Penguin, New York

Fisher R, Brown S (1988) Getting Together. Penguin, New York

Grudin J (1988) Why CSCW applications fail. In: D. Tatar (ed), Proceedings of the Second Conference on Computer-Supported Cooperative Work. ACM Press

Grudin J (1991) Systematic sources of suboptimal interface design in large product development organization. Human-Computer Interaction, June

Kolb D (1983) The Mediators. MIT Press, Cambridge, MA

Lai K, Malone T, Yu K (1989) Object Lens: A ‘spreadsheet’ for cooperative work. ACM Trans Office Infor Syst 6(4):332–353

Lee J, Malone T (1988) How can groups communicate when they use different languages? In: R Allen (ed), Proceedings of the ACM Conference on Office Information Systems. Palo Alto, CA

Lee J, Malone T (1990) Partially shared views. ACM Trans Infor Syst 15(3):221–33

MacLean A, Young RM, Bellotti VME, Moran TP (1991) Questions, options and criteria: Elements of design space analysis. Human-Computer Interaction 6 (3,4)

Pruitt D, Rubin J (1986) Social Conflict: Escalation, stalemate and settlement. Random House, New York

Susskind L, Cruikshank J (1987) Breaking the Impasse: Consensual Approaches to Resolving Public Disputes. Basic Books, New York