Telecooperation Experience with the WinWin System

Alexander Egyed and Barry Boehm

Published in the Proceedings of the 15th IFIP World Computer Congress, 1998

Abstract

WinWin is a telecooperation system supporting the definition of software-based applications as negotiated stakeholder win conditions. Our experience in using WinWin in defining over 30 digital library applications, including several telecooperation systems, is that it is important to supplement negotiation support systems such as WinWin with such capabilities as prototyping, tradeoff analysis tools, email, and videoconferencing. We also found that WinWin's social orientation around considering other stakeholders' win conditions has enabled stakeholders to achieve high levels of shared vision and mutual trust. Our subsequent experience in implementing the specified digital library systems in a rapidly changing web-based milieu indicated that achieving these social conditions among system stakeholders was more important than achieving precise requirements specifications, due to the need for team adaptability to requirements change. Finally, we found that the WinWin approach provides an effective set of methods of integrating ethical considerations into practical system definition processes via Rawls' stakeholder negotiation-based Theory of Justice.

1. Introduction

By their very nature, telecooperation systems are people-intensive. Moreover, they project people into situations for which their social and ethical instincts are not well prepared: email ethics and etiquette; softcopy copyrights and licenses; privacy vs. freedom of information.

Given the increasing criticality of telecooperation systems, one would like to have system definition methods and tools which recognize the importance of social and ethical considerations, and integrate them within their system definition processes and frameworks.

However, with a few exceptions such as the European Participatory Design approach [11][18], most system development methodologies focus on information structures and dynamics. They generally consider social and ethical (and even economic) considerations as at best orthogonal to the job of specifying a system. More approaches are needed which integrate social, economic, and ethical considerations into the normal process of system definition.

The WinWin system discussed here [5] is an attempt to provide such a capability. It is both a telecooperation system and an approach to appropriately specifying telecooperation systems. Section 2 discusses the WinWin system components. Section 3 summarizes our experience with WinWin as a telecooperation system. Section 4 explains how its stakeholder win-win approach can integrate ethical considerations into system specification via Rawls' *Theory of Justice* [17]. Section 5 provides our conclusions.

2. WinWin System Components

2.1. Theory W

Theory W states that your project will succeed *if and only if you make winners of all the critical stakeholders* [4]. It includes a number of key principles and practices such as identifying critical stakeholders and their win conditions, system requirements as negotiated win conditions, expectations management, inventing options for mutual gain [11], and risk management of win-lose and lose-lose risks.

2.2. The WinWin Negotiation Model

The main purpose of a negotiation model [5] is to provide a stepwise approach for stakeholders to use in reconciling their individual win conditions.

The WinWin Model, as depicted in Figure 1, achieves this. The model contains four major artifact types – *Win Condition*, *Issue*, *Option*, and *Agreement* – and their interrelationships, as well as a *Domain Taxonomy*.

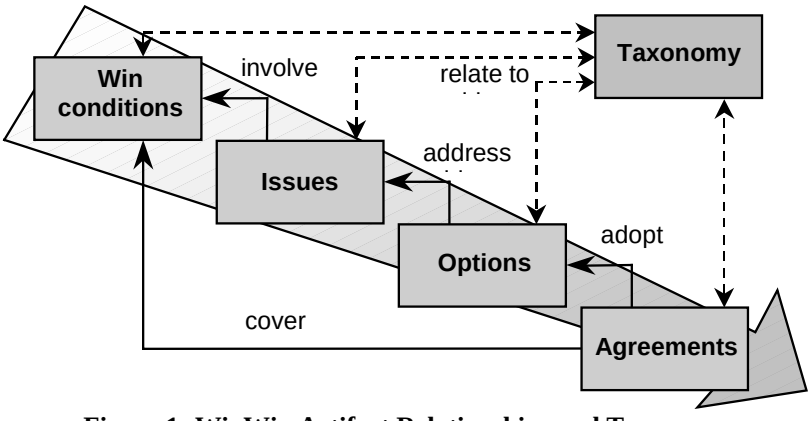


Figure 1: WinWin Artifact Relationships and Taxonomy

Win Conditions capture each stakeholder's goals and concerns with respect to the new system. If a Win Condition is non-controversial, it is covered by an Agreement. Otherwise, an Issue artifact is

created to record the resulting conflict among Win Conditions (and stakeholders). Options allow stakeholders to suggest alternative solutions, which address Issues. Options are explored and refined via tradeoff analysis, expectations management, and negotiation, eventually leading to an Agreement to adopt an Option, which resolves the Issue.

The WinWin Model also includes a tailorable *Domain Taxonomy*, which enables stakeholders to link artifacts to taxonomy items and to access those artifacts via the taxonomy. In the WinWin approach the taxonomy structure follows closely the table of contents of requirements documents. Thus, the negotiators are able use the taxonomy as a checklist for ensuring sufficient coverage of the problem, and the resulting Agreements can be mapped straightforwardly into a requirements specification.

2.3. WinWin Spiral Model and the Iterative WinWin Model

The WinWin Spiral Model [6] is an extension to the ordinary spiral model [3] which answers the question of where the next level objectives, constraints, and alternatives come from. Thus, three activities were added upfront in order to incorporate the WinWin Negotiation Model into a more general development model (see Figure 2):

- Identify the system or subsystem's key stakeholders
- Identify the stakeholders' win conditions for the system or subsystem
- Negotiate win-win reconciliation of the stakeholders' win conditions.

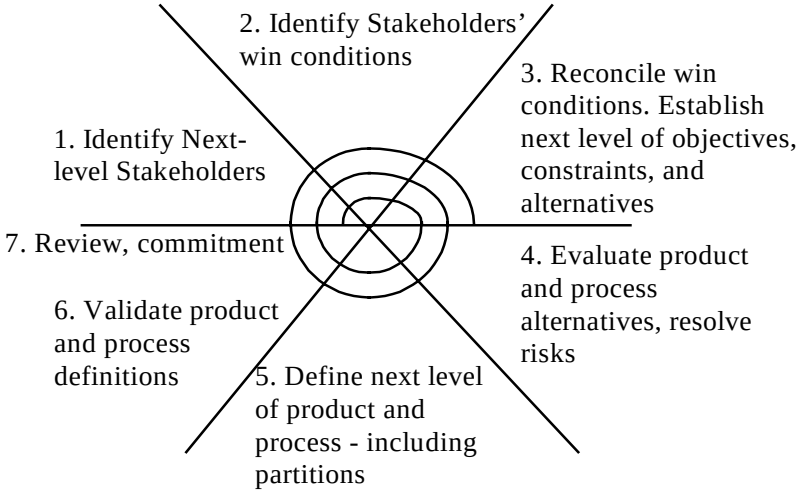


Figure 2: WinWin Spiral Model [6]

We have observed in practice that these three initial steps do indeed yield the objectives, constraints, and alternatives needed for each cycle of the spiral. During each spiral cycle, the

stakeholders concurrently and consistently refine the definitions of the problem (requirements) and its solution (design and plans). We have also defined a set of life cycle anchor points at which the consistency and feasibility of the problem and solution definitions are reviewed as the basis for management go/no-go decisions [6].

2.4. The WinWin Tool

The WinWin System [5][14] is a telecooperation tool that was built to support the WinWin negotiation model (see Figure 3). The tool uses Inter- and Intranet support to enable collaboration between distributed stakeholders. It may be used both synchronously and asynchronously, meaning that stakeholders may negotiate using the tool at the same time, but they may also use it at different times. Further, a number of support tools are integrated with WinWin to assist in the negotiation, especially in order to support tradeoff analyses, and to identify and resolve risks. The following are a few examples:

- A4 (Architecture Attribute Analysis Aid): Architecture-based analysis of cost, schedule, performance, and reliability.
- Rapide: A architecture tool for modeling and simulating systems and identifying problems (deadlocks, bottlenecks, etc.) in the architecture.
- COCOMO (Constructive Cost Model) II: Cost/Schedule estimation tool [2].

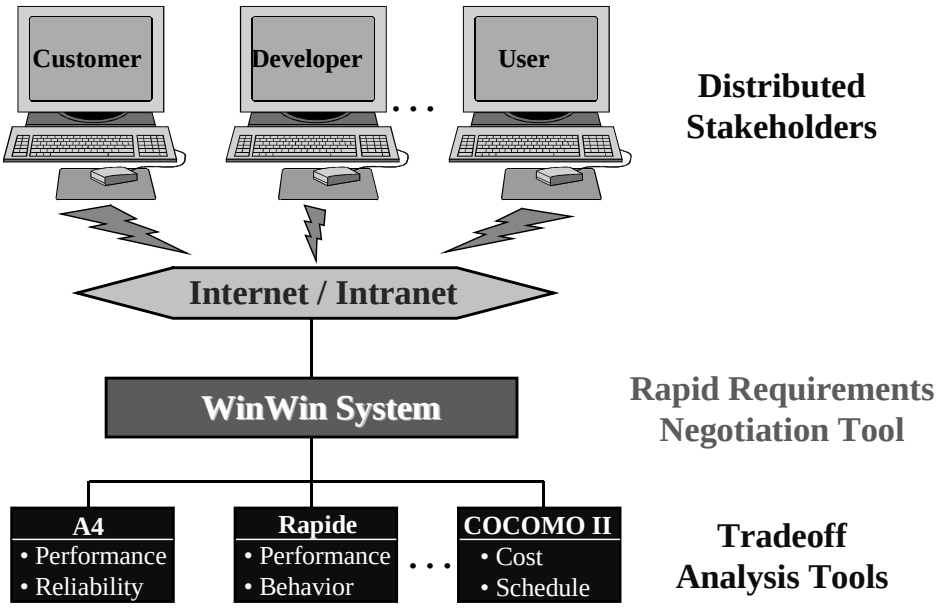


Figure 3: WinWin System Overview

A screen hardcopy of the WinWin tool is given in Figure 4. In the foreground a customer Win Condition of one of the Library projects is visible. To the right is the Taxonomy window and to the

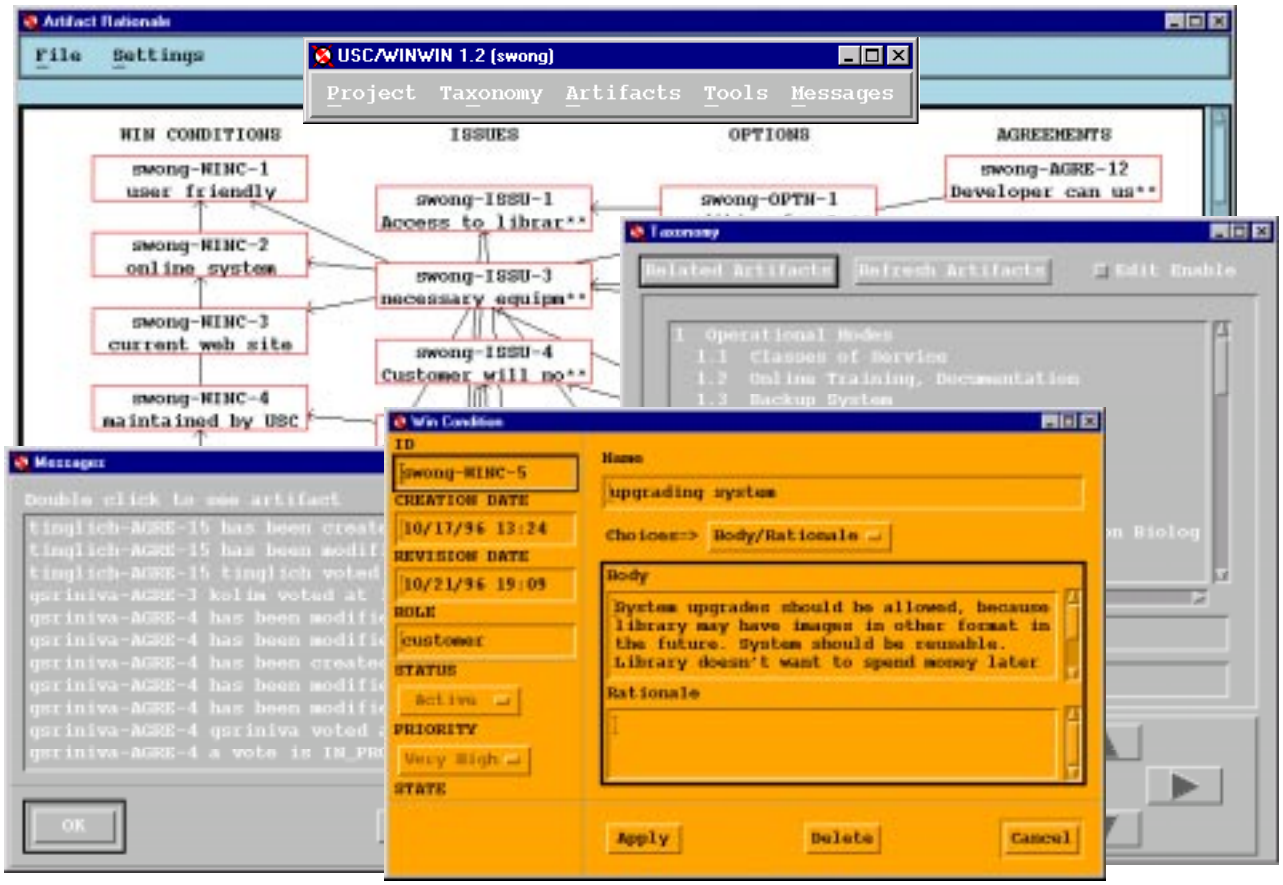


Figure 4: The WinWin Tool

Table 1: WinWin System Capabilities

Capability	WinWin System
Components	Win Conditions, Issues, Options, Agreements, Terms, Taxonomy Items
Connectors	Connections between Artifacts: • *Cover*, *Resolve*, *Adopt* (e.g. option resolves issue) • Simple *Relates To* showing some inter/intra artifact dependency • *Replace* (e.g. Agreement replaces an older one) Connections between Artifacts and Taxonomy Connections with external Tools (e.g. Analysis tools)
Views	Taxonomy view reflecting domain categorization Rationale view reflecting dependency and decision tree Message views reflecting the change history
Navigation	Hypertext Style browsing between Artifacts, Messages, and Taxonomy
Change History	Implicit through Artifact types Explicit through Messages describing nature and extend of changes
Information Sharing	Semi-automatic update (update only when requested by user)
Security	Artifact ownership; artifacts are frozen once voting is initiated
Completeness	Taxonomy (domain coverage) Artifact Flags
Group Control / Collaboration	Artifacts, Messages, and Comments Taxonomy, Terms, Rationale Graph, Status Summary External tools (Attachments)

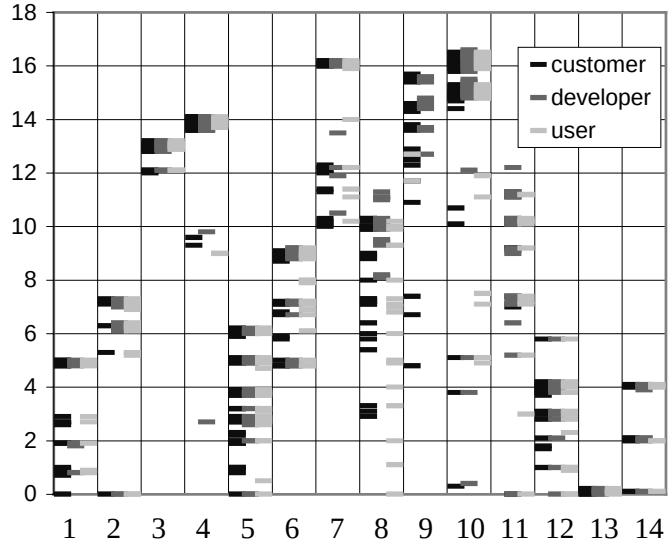
left the Message window. This window contains short descriptions of stakeholder changes in the order they were made. The background of the figure contains a graphical box-and-arrow diagram reflecting the current state of artifacts and their connections. Table 1 summarizes the tool's capabilities. More detailed information about the tool is given in [14].

3. Telecooperation Experience on Digital Library Projects

The USC Digital Library projects [7][8] constitute a major proof of the success and feasibility of the WinWin Development Model (Negotiation Model and Spiral Model). There 15 six-member teams were evaluated while they negotiated and built a series of multimedia related projects, which had to do with stereoscopic slides, medieval manuscripts, technical reports, planning documents, Web-based business data, and others. The developers were graduate students at the University of Southern California. The clients were librarians for USC's various discipline libraries and special collections.

Each project was conceived by a real customer from the USC Library and was derived out of a need in his or her community. Besides proposing the system, the library customers were also involved in negotiating the requirements with their student developer team(s), exercising prototypes, and reviewing project plans. The projects were planned and performed over a period of two semesters. All 15 teams produced satisfactory plans and specifications in the first semester. Six teams then delivered a real product with highly satisfactory initial capabilities (with one exception) at the end of the second semester.

Figure 5: Times people used WinWin (18 day period) by team number



Elaborate metrics were gathered about the negotiation, development process, people, and the deliverables. Results of those metrics were reported in a number of publications [7], [8], [10]. Some of the main results related to the telecooperation aspects of WinWin are shown in Figure 5 and Table 2.

Figure 5 shows the WinWin system usage patterns for each team, in terms of the times in which the customer, developer, and user stakeholder representatives were using WinWin. The teams with a mix of on-campus and off-campus students were Teams 1, 7, 9, 10, and 11. Their usage patterns tend to be more asynchronous than those of the on-campus teams.

Table 2 summarizes the results from the student critiques at the end of the semester. The strongest positive effects of using the WinWin approach were increasing cooperativeness, focusing participants on key issues, reducing friction, and facilitating distributed collaboration. The major improvements for the WinWin approach (now implemented) were increasing WinWin training, reducing usage overhead, and concurrent negotiation and prototyping. In addition, the Librarian critiques strongly indicated that the WinWin approach had increased mutual confidence and trust, and a willingness to participate in future projects.

Table 2: Student Critiques Summary

Positive Comments about WinWin	Count	Negative Comments about WinWin	Count
Should continue use of WinWin	9	Need more pre-WinWin homework	14
Promoted more cooperativeness and mutual understanding	9	Too much overhead in WinWin mechanics, bugs decreased negotiability	10
Focused team on key issues	8	Prototype concurrently w. WinWin conflict identification	10
Objective artifacts reduced friction, equalized loud and quiet participants	6	Should have direct Librarian involvement	6
Helped in distributed collaboration	5	Complement WinWin with email, whiteboards, video conferencing, etc.	5
Helped create better requirements	4		
Helped in understanding the requirements process	2	Need more time to do thoroughly	4
		UNIX platform limitations	2
Helped in adopting to changes	2	Need easier discussion support	2

In the Fall of 1997, 16 more teams successfully produced plans and specifications for additional Digital Library systems. Two of the teams represented more ambitious telecooperation experiments with respect to our campus and libraries in Los Angeles, one team being in San Diego and one team in Tucson, Arizona. For these teams, we supplemented WinWin and web-based project artifacts with videoconferenced Architecture Review Board meetings, including remote exercise of prototypes. After some startup difficulties, both teams were able to successfully collaborate with and satisfy their Library clients. Also some of the 1997 applications were telecooperation systems such as on-line network consultation support and semi-automated reference librarians. For these, the concurrent exercise of WinWin and prototyping were particularly important.

4. Integrating Ethics into Software Engineering Practice via WinWin

Some good software engineering ethical guidelines are provided in such publications as the *Software Engineering Code of Ethics* [13] and the *ACM Code of Ethics and Professional Conduct* [1]. However, as mentioned above, such guidelines are not generally integrated into system definition methods and tools.

We have performed such an integration by linking the stakeholder win-win approach in WinWin with Rawls' *Theory of Justice* [17]. Rawls' theory is based on performing rational negotiation of issues by participants of society. It attempts to remove special-interest biases via a "veil of ignorance," in which participants address representative issue scenarios without knowledge of which participants may be most-favored or least-favored parties.

Collins et al. [9] have translated Rawls' theory of justice into the software domain, situating negotiations among a software system's *provider*, *buyer*, *user*, and a representative of public interest called the *software penumbra*. They provide a set of canonical ethical obligations of these stakeholders, and an example of negotiation considerations for a hospital information system, but do not identify methods or tools for supporting the negotiation process.

The WinWin approach and toolset does provide such support for integrating a Rawls-based ethics approach into the system definition process, by involving a stakeholder called an ombudsman, representing Collins et al.'s penumbra. We found it impractical to support the "veil of ignorance" in practical system definition negotiations, but otherwise found the approach workable.

An example of this integration occurred in applying the WinWin approach to the definition of an urban fire dispatching system. A win condition was entered indicating that dispatching decisions should be made primarily to minimize the loss of human life and health primarily, and secondarily to minimize the dollar loss due to property damage. An ombudsman stakeholder representing the general public entered an issue stating that basing dispatching decisions on dollar loss would discriminate against responding to fires in poor neighborhoods. The resulting agreement specified that minimizing dollar loss should be considered in responding to individual file incidents, but not in prioritizing across fire incidents.

Other ethical issues surfaced in the WinWin approach for the fire dispatching system included multilingual telephone operators, record keeping and accountability, and nominal vs. crisis performance priorities.

5. Conclusions

Our experience on over 30 projects using WinWin as a telecooperation system has indicated that its support for asynchronous negotiation and its task-oriented set of negotiation artifacts have made it an effective tool for rapid, distributed requirements negotiation. We found, however, that it is important to supplement negotiation support systems such as WinWin with such capabilities as prototyping, tradeoff analysis tools, email, and videoconferencing.

We also found that WinWin's social orientation around considering other stakeholders' win conditions has enabled stakeholders to achieve high levels of shared vision and mutual trust. Our subsequent experience in implementing the specified digital library systems in a rapidly changing web-based milieu indicated that achieving these social conditions among system stakeholders was more important than achieving precise requirements specifications, due to the need for team adaptability to requirements change.

Finally, we found that the WinWin approach provides an effective set of methods of integrating ethical considerations into practical system definition processes via Rawls' stakeholder negotiation-based Theory of Justice.

6. Acknowledgements

This research is sponsored by DARPA through Rome Laboratory under contract F30602-94-C-0195 and by the Affiliates of the USC Center for Software Engineering: Aerospace Corp., Air Force Cost Analysis Agency, Allied Signal, Bellcore, Boeing, Electronic Data Systems, E-Systems, GDE Systems, Hughes Aircraft, Interactive Development Environments, Institute for Defense Analysis, Jet Propulsion Laboratory, Litton Data Systems, Lockheed Martin, Loral Federal Systems, MCC, Motorola, Network Programs, Northrop Grumman, Rational Software, Science Applications International, Software Engineering Institute, Software Productivity Consortium, Sun Microsystems, TI, TRW, USAF, Rome Laboratory, US Army Research Laboratory, and Xerox.

7. References

[1] Anderson, R., et al., "Using the New ACM Code of Ethics in Decision Making," Communications of the ACM, February 1993, pp. 98-105.

[2] Boehm, B.W., Clark, B., Horowitz, E., Madachy, R., Selby, R., Westland, C., "Cost Models for Future Software Processes: COCOMO 2.0," *Annals of Software Engineering*, Vol.1, 1995.

[3] Boehm, B. W., "A Spiral Model of Software Development and Enhancement," *Computer*, May 1988, v. 21 no. 5, pp. 61-72.

[4] Boehm, B.W. and Ross, R. "Theory W Software Project Management: Principles and Examples," *IEEE Transactions on Software Engineering*, July 1989, pp.902-916.

[5] Boehm, B.W., Bose, P., Horowitz, E., Lee, M.J. "Software Requirements Negotiation and Renegotiation Aids: A Theory-W Based Spiral Approach", *Proceedings of ICSE-17*, April 1995, pp.243-253.

[6] Boehm, B.W., "Anchoring the Software Process," *IEEE Software*, July 1996, pp.73-82.

[7] Boehm, B.W., Egyed, A., "WinWin Requirements Negotiation Processes: A Multi-Project Analysis," *Proceedings, ICSP5*, August 1998.

[8] Boehm, B.W., Egyed, A., Kwan, J., Madachy, R., "Developing Multimedia Applications with the WinWin Spiral Model," *Proceedings, ESEC/FSE 97*, 1997.

[9] Collins, W.R., Miller, K.W., Spielman, B.J., Wherry, P., "How Good is Good Enough?," *Communications of the ACM*, Vol.37, No.1, January 1994, pp. 81-91.

[10] Egyed, A., Boehm, B., "A Comparison Study in Software Requirements Negotiation," *Proceedings, INCOSE'98*, July 1998.

[11] Ehn, P., "Work-Oriented Design of Computer Artifacts," Arbetslivscentrum, Stockholm, 1988.

[12] Fisher, R., Ury W., *Getting to Yes*, Penguin Books, 1981.

[13] Gotterbarn, D., Miller, K., Rogerson, S., "Software Engineering Code of Ethics," *Communications of the ACM*, November 1997, pp. 110-118.

[14] Horowitz, E. "WinWin Reference Manual: A System for Collaboration and Negotiation of Requirements", Center for Software Engineering, University of Southern California Technical Report, March 1996.

[15] Lee, M.J., "Foundations of the WinWin Requirements Negotiation System," Ph.D. Dissertation, Center for Software Engineering, University of Southern California Technical Report, May 1996.

[16] Monge, P., Baskin, A., Fulk, J., Contractor, N., Signer, J., Gibbs, J., Lee, B., Wilson, M., "Report on WinWin Study Findings: Executive Summary", Annenberg School Report, May, 1997.

[17] Rawls, J., *A Theory of Justice*, Harvard University Press, 1971.

[18] Schuler, D., Namioka, A. (editors), "Participatory Design: Principles and Practices," Lawrence Erlbaum Association, Hillsdale, NJ, 1993.

[19] Weiser, M., "Software Engineering that Matters to People," Keynote Address, ICSE 97, IEEE/ACM, May 1997.