

# System Dynamics Modeling and Simulation of Collaborative Requirements Engineering

Friedrich Stallinger

Systems Engineering and Automation  
Johannes Kepler University Linz  
Altenberger Straße 69, A-4040 Linz, Austria  
[fs@sea.uni-linz.ac.at](mailto:fs@sea.uni-linz.ac.at)

Paul Grünbacher

Computer Science Department  
University of Southern California (USC)  
941 W. 37th Place, Los Angeles, CA 90089-0781  
[gruenbac@sunset.usc.edu](mailto:gruenbac@sunset.usc.edu)

## Extended Abstract

### 1. Introduction, Motivation and Background

Inspired by A. M. Christie's ProSim'99 contribution [Christie-99] the proposed paper presents selected aspects of the modeling and simulation of the EasyWinWin requirements elicitation and negotiation process that is currently under development at the Computer Science Department at the University of Southern California (USC-CSE). The process has so far been used in six real-client requirements negotiations at the USC-CSE software engineering class and one real-world COTS product and release planning process. The approach is based on the off-the-shelf product GroupSystems by *GroupSystems.Com*, that provides a set of generic group support tools facilitating collaborative activities like idea generation, idea categorization and prioritization.

Requirements engineering as an early activity in the overall software development process is seen as crucial to the success of any software project as it involves a variety of social and behavioral aspects that significantly effect the quality of the resulting requirements. The modeling effort therefore focuses on these social and behavioral aspects related to requirements elicitation and negotiation and their dynamics. Due to the "smooth" nature of these aspects a continuous modeling approach is used.

The goal of the modeling effort is to gain insights into the behavioral characteristics of the EasyWinWin process, to identify the relevant relationships and causalities, and to validate these insights using the data gathered on the projects mentioned above. The results of the modeling and simulation effort are intended to be used for future process planning and process guidance and thus represent an example of how process modeling and simulation can address real-world software process design, management and improvement issues.

### 2. The EasyWinWin Process

According to the WinWin approach the requirements engineering process depends heavily on the collaboration of stakeholders with different backgrounds and objectives. The approach aims at making winners of all success-critical stakeholders in software development by capturing and reconciling their win conditions in a negotiation process.

The major goals of the EasyWinWin approach were to develop a 'light-weight' implementation of the WinWin approach [Boehm-98], to lower the entry barrier for all stakeholders by taking advantage of the intuitive user interface of a groupware environment, and to enhance the directness, extent, and frequency of interactions by additionally allowing synchronous, co-located sessions. EasyWinWin therefore defines a set of activities guiding stakeholders (clients, users, and developers) through the process of gathering, eliciting, prioritizing, and negotiating requirements [BGr2000, Gr2000a, Gr2000b]:

- **Refine domain taxonomy:** A domain taxonomy is refined and tailored to the specific needs of a project using the GroupOutliner tool. The MBASE guidelines provide a high-level taxonomy comprising *application capabilities* (system fea-

features and services), *interfaces* (user interface, interfaces to other software or hardware systems), *properties* ('non-functional' requirements), *project & process* (cost, schedule, development tools, support) and *evolution* (expected long-term development of the system) [MBASE-99]. The collaborative refinement of the taxonomy is supposed to help stakeholders understand and influence the scope of the negotiation process.

- **Brainstorm win conditions:** The goal of this activity is to collect stakeholder win conditions in an electronic brainstorming session. The Electronic Brainstorming tool allows gathering ideas simultaneously and – if desired – anonymously.
- **Categorize and consolidate ideas to win conditions:** The objectives of this step are to create a non-redundant list of unambiguous descriptions of win conditions from the brainstorming ideas and to organize the win conditions into the high-level taxonomy elements. Based on the Categorizer tool two techniques are eligible: a) *offline* categorization carried out jointly by the project manager and the requirements engineer, b) *fast focus* categorization supported by a facilitator in a joint meeting by all stakeholders through summarizing and selecting the most important ideas.
- **Capture domain language:** The objective of defining and sharing important terms of the domain language is achieved during the categorization and consolidation of win conditions by the facilitator through collecting a list of terms appearing in the ideas. The electronic brainstorming tool supports this activity by automated grouping of ideas based on a list of keywords and synonyms. The Topic Commenter tool is then used to review the ideas and craft definitions for the different terms. All Stakeholders review and comment the definitions.
- **Prioritize win conditions:** In this step the collected win conditions are classified from two perspectives: the *business importance* expressing the relevance of a win condition to clients and users and the *ease of realization* indicating the perceived and expected difficulties of realizing or implementing a win condition (e.g., technical or financial constraints). In the corresponding voting process developers focus on technical issues while clients and users rate the business relevance. Based on the voting results the win conditions are grouped into four priority categories: *Low hanging fruits*, i.e. win conditions showing a high business importance and low expected difficulties; *Important with hurdles*, i.e. high-priority win conditions that are difficult to implement; *Maybe later*, i.e. low-priority win conditions that may be considered later because of their low difficulty of realization; and *Forget them*, i.e. unimportant win conditions that are difficult to achieve.
- **Negotiate issues, options, and agreements:** Conflict identification and resolution in the WinWin negotiation model is based on the Issues, Options and Agreement artifacts. These artifacts are organized in a tree using GroupOutliner where the level in the tree determines the type of an artifact: *priority category* on level one, *win conditions* on level two, *issues or agreements* on level three, *options* on level four, and *agreements* on level five.
- **Mapping to the taxonomy:** In this final activity, stakeholders associate WinWin artifacts to taxonomy elements in the Categorizer tool. In the first step, the team collaboratively relates the artifacts to the taxonomy elements. In the second step, the resulting relations are reviewed to address one to many relationships and incorrect associations.

The current recommendation for carrying out the activities relevant in the context of this paper are *same time/same place* or *same time/different place* for the brainstorming and *same time/same place* or *different time/different place* for the negotiation activity. The experiments used for data gathering mentioned above have been carried out according these recommendations.

### 3. A System Dynamics Model of the EasyWinWin Requirements Engineering Process

The process steps agreed to be most critical regarding their effect on the overall “quality” of the resulting requirements definition are the requirements gathering, i.e. the *brainstorm win conditions*, and the requirements negotiation, i.e. *negotiate issues, options, and agreements* activity. The overall strategy of modeling and validating the EasyWinWin process therefore currently focuses on modeling the behavioral and social characteristics of these two processes and on using empirical data also related to the intermediate or subsequent process steps to “calibrate” the output variables of these models. The remaining process steps are currently not modeled in detail regarding their social and behavioral characteristics.

The paper lays out the general concept of the system dynamics models related to these two processes. As an example to demonstrate the modeling approach, details on the requirements gathering process are provided.

The general scenario of this requirements gathering session is that three basic groups of stakeholders – clients, users, and developers – meet for an electronic, time-limited brainstorming session to gather ideas for their potential win conditions.

The effects of a preceding *refine domain taxonomy* session on the brainstorming process are represented by one input variable for each stakeholder expressing the extent of understanding of the negotiation process and the high level taxonomy. The major individuals' characteristics modeled are *technical capability*, *domain knowledge*, *ability to understand/learn* the anonymously posted ideas of others, and a *creativity factor* representing the "inspiration effect" related to the brainstorming technique. The behavioral model of the *brainstorm win condition* session itself is implemented as a time-limited dynamic process where individuals are sharing their time between posting ideas and reviewing/learning/creating own ideas.

The major output variables produced by the requirements gathering model are the *completeness* of the ideas gathered, i.e. to which degrees have the participants been able to bring in their win conditions, and - harder to identify in real world empirical data - the *overall quality* of the gathered ideas.

As indicators for the *completeness* of the win conditions the *idea generation rate* (ideas per person per minute) – directly observable in the brainstorming process - together with the *unique ideas ratio* (percentage of *WinConditions* remaining after the subsequent facilitated categorizing and consolidation of ideas) are used as well as the coverage of taxonomy elements by win conditions.

The *overall quality* is computed as the average understanding of the posted ideas by all stakeholders at the end of the brainstorming session. As real-world indicators for this *overall quality* we currently use the average *time spent for categorizing and consolidating* the ideas to features related to the *unique ideas ratio* as well as voting results from the *prioritize win conditions* process step, namely the vote spread and discrepancy regarding the business importance dimension, i.e., how important are the consolidated features and to which degree to the stakeholders agree on this importance.

The model is implemented in the Extend tool primarily due to the potential of combining and enlarging the sub-models currently under development to represent organizational aspects of the EasyWinWin process (e.g. splitting of sessions, modeling of resource constraints, etc.).

#### 4. First Results, Conclusions, and Future Work

Beside the ongoing encouraging calibration and validation activities related to the major model components – including additional trial applications of the process -, the simulation model is used for experiments like determining the effects of varying numbers of participants for each stakeholder group or the influence of variations in intra-group stakeholder understanding due to pre-coordination activities. Although the development of realistic behavioral models of group or behavioral phenomena and their validation is a non-trivial undertaking, the analysis of empirical data and simulation results clearly indicates that these phenomena can be important in determining the outcome of software processes and that modeling and simulation of these phenomena can be useful for process or project planning and guidance.

#### References

- [Christie-99] Christie A. M., "Organizational and Social Simulation of a Software Requirements Development Process", ProSim'99 Software Process Simulation Modeling, June 27-29, 1999, Silver Falls, Oregon, Oregon Center for Advanced Technology Education (OCATE), Portland OR, 1999.
- [MBASE-99] Boehm B., Port D., Egyed A., Abi-Antoun M., "The MBASE Life Cycle Architecture Package", In: Donohoe P. (ed.), Software Architecture, pp. 511-528, Kluwer Academic Publishers, 1999.
- [Boehm-98] Boehm B., Egyed A., Kwan J., Port D., Shah A., Madachy, R.: Using the WinWin Spiral Model: A Case Study, IEEE Computer, p. 33-44, July 1998.
- [BGr2000] Boehm B., Grünbacher P., "Supporting Collaborative Requirements Negotiation: The EasyWinWin Approach", International Conference on Virtual Worlds and Simulation, San Diego 2000 (to appear).
- [Gr2000a] Collaborative Requirements Negotiation with EasyWinWin, USC-CSE TR, 2000.
- [Gr2000b] Grünbacher P., Egyed A., Medvidovic N., "Dimensions of Concerns in Requirements Negotiation and Architecture Modeling", Technical Report, USC-Center for Software Engineering, Los Angeles, CA.