

8-15-1997

Collaborative Design of Organizational Systems

Gert-Jan de Vreede

Delft University of Technology, devreede@sepa.tudelft.nl

Douglas L. Dean

University of Arizona, ddean@bpa.arizona.edu

Jim D. Lee

University of Arizona, jlee@bpa.arizona.edu

Follow this and additional works at: <http://aisel.aisnet.org/amcis1997>

Recommended Citation

Vreede, Gert-Jan de; Dean, Douglas L.; and Lee, Jim D., "Collaborative Design of Organizational Systems" (1997). *AMCIS 1997 Proceedings*. 72.

<http://aisel.aisnet.org/amcis1997/72>

This material is brought to you by the Americas Conference on Information Systems (AMCIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 1997 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

Collaborative Design of Organizational Systems

Gert-Jan de Vreede¹, Douglas L. Dean², Jim D. Lee², Douglas R. Vogel², Rich E. Orwig²

¹Delft University of Technology, PO Box 5015, 2600 GA, The Netherlands

email: devreede@sepa.tudelft.nl

²University of Arizona, Tucson, AZ 85721, USA

email: {ddean,jlee,dvogel,rorwig}@bpa.arizona.edu

Introduction

The collaborative design of an organizational system is a process in which members of an organization cooperate in making plans, policies, or decisions with respect to an organizational systems that affect them all. Organizational systems cover a broad range of systems that support the purposeful functioning of an organization, including information systems, work arrangements, and coordination mechanisms. An organizational system consist of *human* elements, such as the members of the organization and their working methods, of *technical* elements, such as hardware and software, and of *informational* elements, such as data.

Collaborative approaches for the design of organizational systems are gaining importance.

In *organizational settings*, teams are gaining importance as the dominant vehicle to accomplish organizational activities. Organizational groups have to make many decisions in an environment that is both politically and technically complex. Also, even when external analysts assist in the redesign of organizational systems, detailed understanding of opportunities, constraints, and change implications are possessed by organizational members. Thus, a collaborative process is essential.

In *inter-organizational settings*, organizations are seeking partnerships with other organizations to link their processes to mutual benefit. Electronic commerce is booming as an arena in which organizations have to work together in designing and developing inter-organizational systems that are beneficial and supportive to all parties involved. In such arenas, collaborative approaches and collaborative decision making are of vital importance in that, generally speaking, no single organization has the power nor the means to control all others.

The past decade has seen the emergence of information technologies, such as groupware, that facilitate groups of people engaging in collaborative work processes. This technology is proving its potential to increase the productivity of collaborative group work. The collaborative design of organizational systems is a fertile application area for the application of this technology. In this paper, we sketch an approach to organizational systems design that incorporates methods and technologies that facilitate stakeholder involvement. A graphical depiction of this approach is presented in figure 1. The regular lines represent general precedence while the dashed lines suggest feedback and revision.

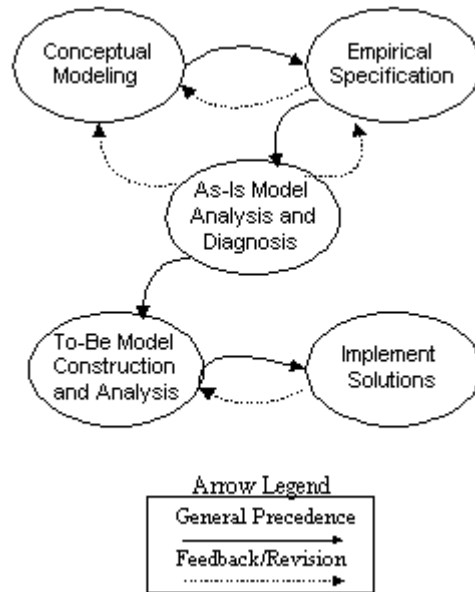


Figure 1. Analysis, Design, and Change Process.

Although a general sequence tends to be followed, iteration occurs between a number of the activities. Moreover, an activity often begins before the completion of its predecessor. The exact way in which the activities are carried out is determined by the nature of the (changing) situation in which the system is to be designed, and the style of analysts and stakeholders involved in the effort. In the remainder of this paper, we will discuss the separate activities of this process in more detail and provide some non-exhaustive samples of collaborative tasks supported by groupware that align with those activities. The authors of this paper acknowledge that additional work has been done on these topics. An exhaustive review of literature is beyond the scope of this paper.

1. Conceptual Modeling

The purpose of a conceptual model is to highlight the structure of the problem situation, i.e. the organizational system in its context. The conceptual model defines the boundaries of this situation. The following activities are carried out (in a non-linear fashion):

1. *Identify Building Concepts.* To construct the conceptual model, the relevant concepts of the problem situation are identified and expressed in a specific modeling language. For example, actors, resources, items, item stores, tasks/activities, and events [Vreede 1995].
2. *Create Aspect Models.* A conceptual model consists of a set of aspect models that each represent certain building concepts in order to highlight specific characteristics of the modeled situation. Aspect models can, for example, focus on *flow* (like IDEF3, flow charts), *functional decomposition* (like IDEF0), *communication* (like speech/act models), or *process dynamics* (like simulation, animation).
3. *Reduce Complexity.* A determination must be made as to what details will be taken into account versus which will be abstracted or ignored. This has to do both with the breadth (horizontal scope) and the depth (amount of vertical detail) included in the aspect models.
4. *Review Aspect Models.* Intermediate and final versions of aspect models are reviewed to correct incorrect representations of the situation and to create a shared understanding of this situation.

Collaborative experiences with conceptual modeling include Functional/Activity Modeling and Data Modeling using IDEF0 and IDEF1x group modeling tools [Dean 1994, 1995, 1994-95, 1996, 1997, Lee

1995, 1997, Vogel 1993], Process Modeling using TeamGraphics, a group drawing tool, and animation [Vreede 1996, 1997], and Object-Oriented Analysis [Katic 1996].

1. Empirical Specification

An empirical specification of the problem situation is a more detailed derivative of the conceptual model, so that it can be used to analyze and diagnose the situation. To this end, the following sub-activities can be distinguished:

1. *Collect empirical data on building blocks.* Empirical data is collected on each of the conceptual building blocks, describing their characteristics in more detail. E.g. the occurrence and duration of certain events, or the capacity and working schedules of actors.
2. *Reduce Complexity.* As the goal of the empirical specification activities is to build a model that is adequate for analysis and diagnosis rather than having a perfectly detailed model, the complexity of the real situation is reduced, e.g. by omitting non-key building blocks, by narrowing the scope of the process being considered, or by abstracting away from unimportant detail.
3. *Build specification.* Combine the data collected and simplify the operators to produce an empirical model based upon the conceptual model.
4. *Check Correspondence between model and real world.* Determine whether the model corresponds to key aspects of the real world process. This can be done, for example, with structured walk throughs, expert reviews, and collecting additional, independent data through observation.

Collaborative experiences with the construction of empirical specifications include Systems evaluations [Orwig 1995], and using animated simulations to facilitate group discussions aiming (1) to collect empirical data and expert estimations, and (2) validate models [Meel 1994, Vreede 1995].

3. As-Is Model Analysis and Diagnosis

The previous two activities were aimed at creating an As-Is model of the problem situation. The purpose of the next activity, As-Is model analysis and diagnosis, is to find the real causes of the problem(s) and to define when the problem can be considered as solved. The following steps can be distinguished:

1. *Define Problem Indications.* By reviewing the model, e.g. using structured walk throughs, simulation experiments, group discussions with stakeholders, or comparisons to reference models, various indications of problems can be identified and marked down.
2. *Set cause hypotheses.* Based on the problem indications, a number of underlying causes of these problems are hypothesized.
3. *Test cause hypotheses.* The As-Is model may be used to test the cause hypotheses, e.g. in case of a simulation model by making modifications and running an experiment. The test outcomes may lead to modified cause hypotheses or conclusions of the diagnosis.
4. *Define Problem Resolution Criteria.* It is stated what has to be done in order for the problem to be considered fixed. In other words, state the requirements of the organizational system to be designed.

Our collaborative experiences with respect to As-Is model analysis and diagnosis include (electronic) discussions with stakeholders of simulation results [Eijck 1996, Meel 1994, Vreede 1995].

1. To-Be Model Construction and Analysis

When the causes of the problems are identified, and hence the requirements of the organizational systems are determined, models of the future system have to be developed and analyzed. This includes the following activities:

1. *Identify potential solutions.* Identify ways to handle the causes underlying the problems in the current situation. Use theories, reference models or situations, creativity exercises, and improvement walk throughs where each step is scrutinized for possible improvements.
2. *Conceptualize solutions.* For a description see Section 1.
3. *Empirically specify solutions.* For a description see Section 2.
4. *Analyze solution models.* Evaluation of individual solutions.

Collaborative experiences in this area include the participative identification of potential solutions [Dennis 1994; Eijck 1996; Vreede 1997], the evaluation of prototypes of organizational systems [Vreede 1995], and the testing of working procedures in GroupSystems sessions [Vreede et al. 1994].

1. Implement Solutions

The final activity in the design of organizational systems comprises of the roll out of one or more To-Be models into an actual system implementation. The following activities are carried out:

1. *Perform comparative analysis of system alternatives.* Determine which (set of) system alternatives (solutions) satisfies the solution requirements best.
2. *Select alternative(s) to be implemented.* Based on the comparative analysis of the alternatives, a selection has to be made which solutions are going to be implemented in the current situation.
3. *Prepare implementation.* To prepare the implementation of the selected solutions, the necessary training, change scenarios, and implementation plans are set up.
4. *Enact Change.* Finally, the actual implementation of the organizational system is enacted.

Our collaborative experiences in this area include the collaborative writing of policy documents [Mittleman 1996], and groupware (GroupSystems/Lotus Notes) support for gaming [Smit 1996, Vreede and Vogel 1997].

Lessons Learned

Numerous lessons have been learned from years of doing collaborative systems design. These include: practicing what we preach works, user involvement pays off, forms and graphics are synergistic, modelers/facilitators are crucial, usability studies are more productive than traditional laboratory experiments in this type of research, and field group feedback is critical.

Usability and Field Studies vs. Laboratory experiments. Real world groups and successive refinement through iteration lead to improved collaborative processes and tools. Laboratory experiments are not sufficient for research on collaborative processes and tools intended for use with real groups.

Stakeholder involvement pays off. Stakeholder involvement pays off in many ways. As subjects, stakeholders provide the realistic user environment that is critical to successful evaluation of tool features and facilitation procedures. Stakeholder involvement also pays off in terms of generating additional groups and often financial resources to enable further method and tool development and continued cumulative testing.

Forms and graphics are synergistic. Exclusive use of forms or graphics for modeling tasks have proved to be inadequate. Forms alone are easy to use but inadequate to illustrate the bigger picture of the model. Graphics alone are useful for representation and feedback but cumbersome and difficult to use by many stakeholders, particularly those with fewer computer skills.

Modelers/facilitators are crucial. Historically, modelers have worked as individuals or in small group environments. The dynamics associated with larger groups place many demands on modelers that extend

into the facilitation domain. Modelers need to become comfortable working in group environments with a wide variety of organizational levels and interests. Traditional group facilitators also need to develop modeling skills to be effective in these activities to sustain credibility with the group. Overall the need to develop combined modelers/facilitators is a continuing challenge.

Future Efforts

Future efforts in collaborative systems design are heading in several complementary directions. First, efforts are being expended to evaluate the implications of doing this activity in distributed domains. Second, modeling research is moving from activity and data modeling to process modeling driven from scenarios. Third, work is progressing to reduce the costs of constructing animated simulation models and hence more efficiently integrate simulation and animation into various aspects of collaborative systems design. Finally, efforts are being extended to support operational level group activities relevant to software developers including design and code inspections.

References

A full list of references can be obtained from the authors.