

12-31-2006

# Defining Key Concepts for Collaboration Engineering

Robert Briggs

*Center for Distance Education University of Alaska at Fairbanks- Delft University of Technology*

Gwendolyn Kolfschoten

*Delft University of Technology*

Vreede Gert-Jan

*University of Nebraska at Omaha- Delft University of Technology*

Dean Douglas

*Brigham Young University*

Follow this and additional works at: <http://aisel.aisnet.org/amcis2006>

---

## Recommended Citation

Briggs, Robert; Kolfschoten, Gwendolyn; Gert-Jan, Vreede; and Douglas, Dean, "Defining Key Concepts for Collaboration Engineering" (2006). *AMCIS 2006 Proceedings*. Paper 17.

<http://aisel.aisnet.org/amcis2006/17>

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 2006 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact [elibrary@aisnet.org](mailto:elibrary@aisnet.org).

# Defining Key Concepts for Collaboration Engineering

**Robert O. Briggs**

Center for Distance Education  
University of Alaska at Fairbanks  
Delft University of Technology  
bob.briggs@uaf.edu

**Gwendolyn L. Kolfshoten**

Delft University of Technology  
g.l.kolfshoten@tbm.tudelft.nl

**Gert-Jan de Vreede**

University of Nebraska at Omaha  
Delft University of Technology  
gdevreede@mail.unomaha.edu

**Douglas L. Dean**

Brigham Young University  
doug\_dean@byu.edu

## ABSTRACT

Collaboration Engineering (CE) is an approach to designing collaborative work practices for high-value recurring tasks, and deploying those designs for practitioners to execute for themselves without ongoing support from professional facilitators. CE is a fast-moving field of research and practice that has not yet reached maturity. A growing community of researchers is actively contributing to the CE literature. To establish common language among CE researchers in, and to focus the research efforts in this domain, this paper documents the current definitions of key concepts in the CE domain. This paper also examines the scope of CE as a domain of research and practice.

## Keywords

Collaboration Engineering, ThinkLets, Patterns of Collaboration, Facilitation, Collaboration

## INTRODUCTION

Many organizations now face tasks that are sufficiently complex that no individual can accomplish them alone. To increase the efficiency and effectiveness of collaboration, many organizations turn to collaboration technologies. However, good technologies are not enough; their value depends on how skillfully and purposefully they are used. Collaboration technologies used unskillfully may not yield increased group productivity (Agres, Vreede and Briggs, 2005; Vreede, Davison and Briggs, 2003). Many groups therefore turn to professional facilitators with expertise in designing and executing group processes, who know how to use collaboration technologies to improve group productivity (Dickson, Limayem, Lee Partridge and DeSanctis, 1996; Griffith, Fuller and Northcraft, 1998; Niederman, Beise and Beranek, 1996). Facilitators, however, can be expensive and because of their skills, they may be promoted quickly to new positions (Munkvold and Anson, 2001). Thus, many groups who could benefit from skilled facilitation cannot avail themselves of it.

*Collaboration Engineering* (CE) is an approach to designing collaborative work practices for high-value recurring tasks, and deploying those designs for practitioners to execute for themselves without ongoing support from professional facilitators (Vreede and Briggs, 2005). CE aims to provide some benefits of professional facilitation to groups without access to professional facilitators. CE researchers therefore seek to address two key challenges: 1) developing more complete understandings of how teams accomplish their tasks by going through various patterns of collaboration and how these patterns may be invoked predictably; and 2) developing better understandings of how collaboration designs may be transferred to collaboration practitioners in ways that result in self-sustaining and growing communities of practice.

The CE research community uses the Five Ways model (Seligmann, Wijers and Sol, 1989) to organize the many research questions it addresses. The Five Ways model posits that an engineering approach can be described through its:

1. Ways of Thinking (concepts and theoretical foundations)
2. Ways of Working (structured design methods)
3. Ways of Modeling (conventions for representing aspects of the domain and the approach)
4. Ways of Controlling (measures and methods for managing the engineering process)
5. Ways of Supporting (tools, approaches and techniques to support the designer)

As the Collaboration Engineering approach has begun to formalize, several papers have been published on the Way of Thinking (e.g. (Briggs, 1994; Briggs, Kolfshoten and Vreede, 2005; Briggs, Qureshi and Reinig, 2004; Briggs, Vreede and Nunamaker, 2003a; Santanen, Vreede and Briggs, 2004)), the Way of Working (e.g. (Briggs et al., 2003a; Briggs, Vreede, Nunamaker and David, 2001; Kolfshoten, Vreede, Chakrapani and Koneri, 2006b; Vreede and Briggs, 2005; Vreede, Briggs and Kolfshoten, 2006), the Way of Modeling (e.g. (Kolfshoten, Briggs, Vreede, Jacobs and Appelman, 2006a; Vreede et al., 2006), and the Way of Supporting (e.g. (Briggs and Vreede, 2001; Kolfshoten and Veen, 2005)). Several cases have also been published about efforts to apply CE principles to problems in the field (e.g. (Agres et al., 2005; Appelman and Driel, 2005; Harder, Keeter, Woodcock, Ferguson and Wills, 2005; Vreede, Fruhling and Chakrapani, 2005)).

As the CE research community grows, it is useful to establish a common definition and understanding of key concepts as a useful frame of reference for both researchers and practitioners. This provides the following specific benefits:

- To maintain a common language
- To position research in the Collaboration Engineering field
- To minimize redundant research efforts
- To further clarify the scope and key focus of the newly-emerging research field.

This paper therefore aims to formalize, unify and document current definitions for commonly used terms in Collaboration Engineering, and to standardize the labels for these concepts. It addresses patterns of collaboration, the CE approach and roles, and the thinkLet concept.

## COLLABORATION

The term *collaboration* comes from the Latin word *collaborare* (Harper, 2001). *collaborare* means "work with," which is derived from *com*, meaning "with" + *labore*, meaning "to work." Thus, collaborative efforts are joint, rather than individual. If collaborative efforts are joint, then they must be directed toward a group goal. A goal is *a desired state or outcome* (Locke and Latham, 1990). Thus, collaboration involves multiple individuals who combine their efforts to achieve mutually desired outcomes. Therefore we define *collaboration* as joint effort towards a group goal (Vreede and Briggs, 2005) Note that this definition does not require that members concur on the merits of a group goal, nor that they necessarily feel happy about the group goal; it only requires that members for whatever reason, make effort to achieve the group goal.

### General Patterns of Collaboration

CE researchers use *general patterns of collaboration* to classify group activities based on the changes-of-state they produce. Patterns of collaboration characterize the ways in which group activities can move a group toward its goal(s). Early work in Collaboration Engineering identified five general patterns of collaboration. To achieve its goals, a group could *diverge*, *converge*, *organize*, *evaluate*, and *build consensus* (Briggs et al., 2003a; Vreede and Briggs, 2005). However, discussions among researchers and practitioners revealed that the labels, *diverge* and *converge* bred confusion, that the definition of consensus-building was inconsistent with new theoretical insights. Researchers also identified several sub-patterns for the six general patterns listed below:

- **Generate:** Move from having fewer to having more concepts in the pool of concepts shared by the group
  - *Gather* – Collect and share known concepts from individual group members.
  - *Create* – Produce and share new ideas that were not previously known to group members.
  - *Elaborate* – Add details to concepts that are already shared by the group.
    - § *Decompose* – To characterize a concept in terms of its components and sub-components.
    - § *Expand* – To add details to more fully explain or describe a concept.
- **Reduce:** Move from having many concepts to a focus on fewer concepts that the group deems worthy of further attention
  - *Select* – Choose a subset of existing concepts
  - *Abstract* – Derive more-general concepts from specific instances in the existing set
  - *Summarize* – Capture the essence of the concepts without eliminating unique concepts.
- **Clarify:** Move from having less to having more shared understanding of concepts and of the words and phrases used to express them.
  - *Describe* – Propose alternative explanations and formulations of a concept.
- **Organize:** Move from less to more understanding of the relationships among concepts the group is considering
  - *Classify* – Arrange concepts into labeled clusters.
  - *Structure* – Create spatial arrangements among concepts to represent their conceptual relationships

- **Evaluate:** Move from less to more understanding of the relative value of the concepts under consideration
  - *Poll* – assess the group opinion with respect to the concepts
  - *Rank* – identify an order of preference among concepts
  - *Assess* – specify and elaborate on the value of concepts
- **Build consensus:** Move from having fewer to having more group members who are willing to commit to a proposal.
  - *Measure* – Assess the degree to which stakeholders are willing to commit to a proposal
  - *Diagnose* – Seek understanding of the underlying causes of dissensus
  - *Advocate* – Seek to persuade others to adopt and accept a position
  - *Resolve* – Seek ways to overcome the underlying causes of dissensus

Even though this classification scheme is meant to be comprehensive, it is not taxonomic. Frequently, a single group activity will produce more than one of these patterns. This scheme is useful as it serves several purposes. First, its concepts serve as high-level building blocks for collaboration process design. For example, for an operational risk assessment process a group can first *generate* a list of risks in a brainstorming activity, and then *reduce* the list to eliminate redundancy, and then *evaluate* the likelihood and impact of each risk. Second, it is a useful scheme for selecting among group process techniques. For example, if the group must *generate*, then one might choose to have them use either brainstorming or Nominal Group Technique, both of which are idea generation techniques. However, one would not choose to use a straw poll, because this activity produces evaluations rather than new ideas.

### The Collaboration Engineering Approach

This section labels and defines a collection of concepts that are commonly used in the CE literature. In the introduction to this paper we define *Collaboration Engineering* as an approach to designing collaborative work practices for high-value recurring tasks, and deploying those designs for practitioners to execute for themselves without ongoing support from professional facilitators. Let us examine the concepts embedded in this definition.

- A *work practice* is a set of actions carried out repeatedly to accomplish a particular organizational task.
- A task is said to be *collaborative* if its successful completion depends on joint effort among multiple individuals.
- A task is said to be *high-value* if the organization derives substantial benefit or forestalls substantial loss or risk by completing the task successfully. This is not meant to imply that an organization must derive substantial value from each instance of task execution, but rather that the practice of the task over time produces high value.
- A task is said to be *recurring* if the task must be conducted repeatedly, and if a similar process can be used each time it is executed. This does not mean that all aspects of every instance of the task must be similar; only that a similar approach can be used each time the task is executed, despite variations in its parameters.

Collaboration Engineering focuses on recurring tasks because the return on the resources devoted to the CE effort increases each time the work practice is executed, and because if a task does not recur, it would not benefit practitioners to learn a method of execution for the task. In CE collaboration processes are *designed* and *deployed*.

The core activity of Collaboration Engineering is the design of collaborative work practices. In the context of Collaboration Engineering, *to design* (verb) means to create, document and validate a prescription for a collaborative work practice. The product of a CE design effort is called a CE design. We define a *CE design* (noun) as an artifact (usually a document) that defines the sequence and logic of a set of steps for attaining some set of goals, and the conditions under which these steps will be executed. Each step of a completed work practice design should incorporate everything group members and practitioner need to do and say to complete an activity that moves them closer to their goal. A CE design must be sufficiently simple and clear to be transferable to practitioners. The patterns of collaboration the execution of the design creates must be predictable, and it must be reusable for multiple instances of the task.

*Deployment* means implementation of a CE design in an organization. The goal of deployment is to create a self-sustaining community of practice for the CE design. Indications that a self-sustaining community of practice exists include, but are not limited to a) the CE design is deemed to be the standard way the organization executes the task; b) new practitioners are trained to execute the design as a matter of course, and c) if the work practice or the technologies that support it fail, the practitioners seek a remedy instead of abandoning the work practice (Briggs, Adkins, Mittleman, Kruse, Miller and Nunamaker, 1999). Deploying a CE design includes, among other things, motivating practitioners to change from their current practices to the new collaborative work practice, training practitioners to execute the work practice without further

support of facilitators or other collaboration experts, assuring that practitioners have access to adequate resources to execute the work practice as frequently as they need to, and establishing management and control mechanisms for the work practice.

In table 1 we summarize definitions for concepts related to the CE approach. Next, we consider the implications of these definitions for the conduct of CE projects.

Collaboration	<i>Joint effort towards a group goal.</i>
Goal	<i>A desired state or outcome</i>
Collaboration Engineering	<i>An approach to designing collaborative work practices for high-value recurring tasks, and deploying those designs for practitioners to execute for themselves without ongoing support from professional facilitators.</i>
To Design (verb)	<i>To create, document and validate A CE process design</i>
CE Design (noun)	<i>A written statement defining a structured set of steps for attaining objectives, and the conditions under which these steps will be executed</i>
High-value task	<i>A task from which an organization derives substantial benefit or forestalls substantial loss by successful completion.</i>
Recurring task	<i>A task that must be conducted repeatedly, and that can be completed using a similar process design each time it is executed.</i>
Deployment	<i>Implement the collaboration process and support in a way that it becomes a self-sustaining practice within an organization.</i>

**Table 1: Definitions of Key Concepts for Collaboration Engineering**

## ROLES IN COLLABORATION ENGINEERING

There are two key roles in the CE approach: the *collaboration engineer* and the *practitioner* (Vreede and Briggs, 2005; Vreede et al., 2006). CE has its roots in group facilitation, but differs from facilitation in significant ways. A *facilitator* designs and conducts collaboration processes to support a specific group in achieving its specific goals (Griffith et al., 1998; Schwarz, 2002). Facilitators execute their own designs, so it is not necessary that their designs be transferable. Facilitators monitor the needs of the group and the requirements of the task, and when necessary, devise new interventions on-the-fly to make the group more productive. Their professional skills give them the flexibility to adapt to changing conditions during the course of an engagement.

A *collaboration engineer* is a person who designs collaborative work practices and deploys them in organizations such that the practitioners can execute the design for themselves without the ongoing support of an expert facilitator. Thus, a collaboration engineer faces design challenges beyond those faced by a facilitator. Like the facilitator, the collaboration engineer must assure that the CE design will successfully move a group to its goal. However, unlike the facilitator, the collaboration engineer must also make the design sufficiently simple, flexible, and robust that practitioners, who are not group process professionals, can nonetheless execute the design successfully under a variety of circumstances with a variety of participants. The design must be sufficiently predictable that it consistently yields high quality outcomes when executed by the practitioner.

A *practitioner* is a domain expert who learns to execute a particular work practice designed by a collaboration engineer. However, unlike facilitators and collaboration engineers, practitioners need not have the skills to design or conduct collaborative processes other than their own.

## THINKLET DESIGN PATTERNS

CE researchers have begun to create, collect, document, and test collaboration process design patterns for collaborative work practices, which now comprise design pattern language called *ThinkLets*. *Design patterns* are descriptions of known, reusable solutions to recurring problems. The concept of patterns is old, but became popular through the work of Alexander and the gang of four. (Alexander, Ishikawa, Silverstein, Jacobson, Fiksdahl-King and Angel, 1977; Gamma, Helm, Johnson and Vlissides, 1995). *ThinkLets*, are named, scripted, reusable, and transferable collaborative activities that give rise to specific known variations of the general patterns of collaboration among people working together toward a goal. Collaboration engineers select among thinkLets in part based on the variations they produce. For example, one type of *generate* thinkLet may give rise to a diversity of creative ideas, accompanied by discussion, elaboration, and lots of noise, while another may give rise to ideas that conform to known standards of quality, but very little discussion and elaboration. A given thinkLet may also give rise to several general patterns of collaboration simultaneously. For example, a thinkLet may cause a group to *generate* comments that *evaluate* the relative merits of a set of concepts, or to *organize* their comments as they *generate* them by placing each comment to an appropriate category.

Collaboration engineers use thinkLets as reusable building blocks to construct logical designs for collaborative work practices. For an organizational team to achieve a particular goal, several steps may be required, and consequently a CE design is typically composed of several thinkLets. The term, *thinkLet sequence* (Kolfschoten et al., 2006a; Vreede et al., 2006), refers to recurring combinations of multiple thinkLets that may be reused as a single unit in work practice designs.

In many cases, changing the script for a thinkLet will modify the variation of the general patterns it creates. Sometimes the same change-of-script can be applied to a whole collection of thinkLets to create the same predictable variation in the general patterns they produce. This is referred to as a *thinkLet modifier*, a named, scripted variation that can be applied to a set of thinkLets to create a predictable change in the dynamics of those thinkLets. For example, the OneUp modifier can be applied to any *generate* thinkLet. Under the OneUp variation, participants may only contribute new ideas that are arguably better in some way than the ideas that have already been contributed.

The most essential components of a thinkLet specification are rules. Rules describe the *actions* participants must take, the *constraints* under which they must act, and the *capabilities* they will require to execute the actions. (Kolfschoten et al., 2006a; Vreede et al., 2006). *Actions* are things participants must do individually, such as add, edit, move, delete, judge, or associate concepts. *Capabilities* are the means necessary to contribute, record, read, and manipulate concepts. For example, in the FreeBrainstorm thinkLet, the capabilities are “one page for each participant, with view and add rights, and with the ability for participants to exchange pages after each contribution.” A *constraint* is a limitation or guideline on how an action is to be performed. For example, brainstorming comments are constrained to be responsive to the brainstorming question; judgments rendered in polls are constrained by the evaluation criteria, and so on.

Thus, the rules of a thinkLet are the basis for instructions to participants about what they must do and say to complete the activity successfully. Some thinkLets require that different participants take different roles in the activity, and so have multiple sets of rules. For example, some *generate* thinkLets, have one set of rules for contributors, a different set of rules for a scribe, and yet a different set of rules for a devil’s advocate. In some cases, minor alternations of the rules can give rise to significant changes in the behavior of the participants and the results they produce (Briggs et al., 2003a; Santanen et al., 2004). Although the script for a thinkLet must invoke its rules, scripts can be adapted to suit the personal preferences of the people who execute them, so long as the revised script still invokes the rules of the thinkLet. ThinkLet rules also provide a strong basis for comparative research on collaboration support (Santanen, 2005).

Table 2 summarizes the definitions of the key concepts relating to thinkLets. For more details about the formal conventions for documenting thinkLets, see (Kolfschoten et al., 2006a; Vreede et al., 2006).

Design Patterns	<i>Descriptions of known, reusable solutions to recurring problems</i>
Pattern of Collaboration	<i>A description of the nature of the group’s collaborative activities when observed over a period of time as they move from a starting state to some end state.</i>
ThinkLet	<i>A named, scripted collaborative activity that gives rise to a known pattern of collaboration among people working together toward a goal. ThinkLets are design patterns for collaborative work practices</i>
Sequence	<i>A named combination of multiple thinkLets that is reused in a variety of CE designs.</i>
Modifier	<i>A repeatable variation that can be applied to a set of thinkLets to create a predictable change in the group dynamics those thinkLets produce</i>
Actions	<i>Individual efforts made by participants, such as add, edit, move, delete, judge, or associate concepts</i>
Capability	<i>The means necessary to contribute, record, read, and manipulate concepts</i>
Constraints	<i>Limitations on or guidelines on how an action must be executed.</i>
Rule	<i>An instruction to perform an action with a certain capability and under one or more specified constraints.</i>

**Table 2: Definitions Relating to ThinkLets**

**OUTCOMES OR EFFECTS OF INTEREST TO COLLABORATION ENGINEERING RESEARCHERS**

CE researchers must concern themselves with several outcomes of the CE effort (Hengst, Kolfschoten, Dean and Chakrapani, 2006). These include but are not limited to the concepts in Table 3. We offer this set of definitions not to declare it complete, but rather to suggest it is only a small start on the much larger set of phenomena that will be of interest and importance to collaboration engineers.

Sustained Use	<i>The extent to which a thinkLet or work practice can become the standard way of executing the task without ongoing support from experts outside the organization (Briggs et al., 1999)</i>
Predictability	<i>The extent to which a thinkLet or work practice, when executed as prescribed, creates similar variations on the general patterns of collaboration and deliverables across a variety of teams, tasks, and circumstances.</i>
Transferability	<i>The degree to which individuals who did not create a thinkLet or work practice design can learn, remember, and successfully execute it.</i>
Reusability	<i>The extent to which a pattern or design can be used to solve problems other than those for which it was originally developed.</i>
Satisfaction	<i>An affective response a positive or negative valence with respect to attainment of goals (Briggs, Vreede and Reinig, 2003b).</i>
Commitment	<i>The extent to which an individual assumes an obligation to expend resources to fulfill the terms of a proposal (Briggs et al., 2005)</i>
Creativity	<i>The degree to which solutions ideas or solutions are both feasible and novel (fall outside the set of known solutions). (Santanen et al., 2004)</i>

**Table 3. Definitions related to the effects relevant to Collaboration Engineering**

## CONCLUSIONS

The definitions offered in this paper have been derived from the progress made by the CE research community over the last 3 years. While meaningful progress has been made, additional CE research remains to be accomplished. For the Way of Thinking, much work remains to be done to establish theoretical foundations for the general patterns and sub-patterns of collaboration. A challenge also remains to develop a deeper understanding of change-of-work-practice to support the deployment of new CE work practices (Agres et al., 2005). Some theories have been proposed for idea generation, consensus building, and change-of-work-practice, and early validation efforts are in progress. Additional research is required to develop better understandings of the patterns of collaboration, in particular reduction, clarification, organization, and evaluation.

With respect to the Way of Working, preliminary versions of structured methodologies for CE have been proposed (Vreede and Briggs, 2005), but require more rigorous validation in the field. Further, a foundation should be laid to accurately demonstrate that practitioners to whom CE designs have been transferred can successfully execute them over the long term. Researchers must also focus on the predictability, robustness, and re-usability of CE design objects.

For the Way of Modeling, models have been proposed for several aspects of collaboration, but need additional validation (Kolfschoten et al., 2006b).

For the Way of Controlling several evaluation frameworks could be borrowed from other fields (Hengst et al., 2006; Santanen, Kolfschoten and Golla, 2006), but they should be further analyzed and adjusted to obtain metrics and methods for managing CE projects. Further, there is a need for more detailed models to evaluate the effect of thinkLets. Moreover, it will be particularly important to derive metrics for comparing the efficiency and effectiveness of CE designs.

For the Way of Supporting, work has been done on a first prototype of a support tool for CE design (Kolfschoten and Veen, 2005). Experiences with this prototype offered insights for future support tools in terms of requirements and conceptual definitions of collaboration process design elements.

It is our hope that the definitions of key concepts offered in this paper will make it easier for rigorous research and discourse on CE to progress. These definitions will give the CE community a common ground for further development of theories and to further detail the approach and the methods and techniques used. Further discussion among researchers and comparison with other fields is required to further establish these definitions and demonstrate their usefulness in research and practice.

## ACKNOWLEDGEMENT

We gratefully acknowledge the contributions of the CE research community, whose joint efforts and combined intellects led to the understandings captured in this paper.

## REFERENCES

1. Agres, A., Vreede, G.J., de , and Briggs, R.O. (2005) A Tale of Two Cities: Case Studies of GSS Transition in Two Organizations, *Group Decision and Negotiation*, 14, 4.
2. Alexander, C., Ishikawa, S., Silverstein, M., Jacobson, M., Fiksdahl-King, I, and Angel, S. (1977) *A Pattern Language*, Towns, Buildings, Construction, Oxford University Press, New York.
3. Appelman, J.H., and Driel, J. van (2005) Crisis-Response in the Port of Rotterdam: Can we do Without a Facilitator in Distributed Settings?, *Proceedings of the Hawaii International Conference on System Science*, Los Alamitos, IEEE Computer Society Press.
4. Briggs, R.O. (1994) *The Focus Theory of Team Productivity and its Application to Development and Testing of Electronic Group Support Systems*, Tucson.
5. Briggs, R.O., Adkins, M., Mittleman, D.D., Kruse, J., Miller, S., and Nunamaker, J.F. Jr. (1999) A Technology Transition Model Derived from Qualitative Field Investigation of GSS use aboard the U.S.S. Coronado, *Journal of Management Information Systems*, 15, 3, 151-196.
6. Briggs, R.O., Kolfschoten, G.L., and Vreede, G.J. de (2005) Toward a Theoretical Model of Consensus Building, *Proceedings of the Americas Conference on Information Systems*, Omaha.
7. Briggs, R.O., Qureshi, S., and Reinig, B. (2004) Satisfaction Attainment Theory as a Model for Value Creation, *Proceedings of the Hawaii International Conference On System Sciences*, Los Alamitos, IEEE Computer Society Press.
8. Briggs, R.O., and Vreede, G.J., de (2001) *ThinkLets, Building Blocks for Concerted Collaboration*, Delft University of Technology, Delft.

9. Briggs, R.O., Vreede, G.J. de, and Nunamaker, J.F. Jr (2003a) Collaboration Engineering with ThinkLets to Pursue Sustained Success with Group Support Systems, *Journal of Management Information Systems*, 19, 4, 31-63.
10. Briggs, R.O., Vreede, G.J. de, Nunamaker, J.F. Jr., and David, T.H. (2001) ThinkLets: Achieving Predictable, Repeatable Patterns of Group Interaction with Group Support Systems, *Proceedings of the Hawaii International Conference on System Sciences*, Los Alamitos, IEEE Computer Society Press.
11. Briggs, R.O., Vreede, G.J. de, and Reinig, B. (2003b) A Theory and Measurement of Meeting Satisfaction, *Proceedings of the Hawaii International Conference on System Sciences*, Los Alamitos, IEEE Computer Society Press.
12. Dickson, G.W., Limayem, M., Lee Partridge, J., and DeSanctis, G. (1996) Facilitating Computer Supported Meetings: A Cumulative Analysis In A Multiple Criteria Task Environment, *Group Decision and Negotiation*, 5, 1, 51-72.
13. Gamma, E., Helm, R., Johnson, R., and Vlissides, J. (1995) Elements of Reusable Object-Oriented Software, Addison-Wesley Publishing Company.
14. Griffith, T.L., Fuller, M.A. , and Northcraft, G.B. (1998) Facilitator Influence in Group Support Systems, *Information Systems Research*, 9, 1, 20-36.
15. Harder, R.J., Keeter, J.M., Woodcock, B.W., Ferguson, J.W., and Wills, F.W. (2005) Insights in Implementing Collaboration Engineering, *Proceedings of the Hawaii International Conference on System Science*, Los Alamitos, IEEE Computer Society Press.
16. Harper, D (2001) Online Etymology Dictionary, <http://www.etymonline.com/>.
17. Hengst, M. den, Kolfschoten, G.L., Dean, D.L., and Chakrapani, A. (2006) Assessing the Quality of Collaborative Processes, *Proceedings of the Hawaii International Conference on System Science*, Los Alamitos, IEEE computer society press.
18. Kolfschoten, G.L., Briggs, R.O., Vreede, G.J., de, Jacobs, P.H.M., and Appelman, J.H. (2006a) Conceptual Foundation of the ThinkLet Concept for Collaboration Engineering, *International Journal of Human Computer Science*, 64, 7, 611-621.
19. Kolfschoten, G.L., and Veen, W. (2005) Tool Support for GSS Session Design, *Proceedings of the Hawaii International Conference on System Sciences*, Los Alamitos, IEEE Computer Society Press.
20. Kolfschoten, G.L., Vreede, G.J. de, Chakrapani, A., and Koneri, P. (2006b) A Design Approach for Collaboration Engineering, *Proceedings of the First HICSS Symposium on Case and Field Studies of Collaboration*, Kauai, in press.
21. Locke, E.A., and Latham, G.P. (1990) A Theory of Goal Setting and Task Performance, Prentice Hall, Englewood Cliffs.
22. Munkvold, B.E., and Anson, R. (2001) Organizational Adoption and Diffusion of Electronic Meeting Systems: A Case Study, *Proceedings of the Proceedings of ACM GROUP01*, New York, ACM, 279-287.
23. Niederman, F., Beise, C.M., and Beranek, P.M. (1996) Issues and Concerns About Computer-Supported Meetings: The Facilitator's Perspective, *Management Information Systems Quarterly*, 20, 1, 1-22.
24. Santanen, E.L. (2005) Resolving Ideation Paradoxes: Seeing Apples as Oranges through the Clarity of ThinkLets, *Proceedings of the Hawaii International Conference on System Sciences*, Los Alamitos, IEEE Computer Society Press.
25. Santanen, E.L., Kolfschoten, G.L., and Golla, K. (2006) The Collaboration Engineering Maturity Model, *Proceedings of the Hawaii International Conference on System Science*, Los Alamitos, IEEE Computer Society Press.
26. Santanen, E.L., Vreede, G.J. de, and Briggs, R.O. (2004) Causal Relationships in Creative Problem Solving: Comparing Facilitation Interventions for Ideation, *Journal Of Management Information Systems*, 20, 4, 167 -197.
27. Schwarz, R.M. (2002) The Skilled Facilitator, Jossey-Bass, San Francisco.
28. Seligmann, P.S., Wijers, G.M., and Sol, H.G. (1989) Analysing the Structure of IS Methodologies, *Proceedings of the Proceedings of the 1st Dutch Conference on Information Systems*, Amersfoort, the Netherlands.
29. Vreede, G.J. de , Fruhling, A. , and Chakrapani, A. (2005) A Repeatable Collaboration Process for Usability Testing, *Proceedings of the Hawaii International Conference on System Sciences*, Los Alamitos, IEEE Computer Society Press.
30. Vreede, G.J. de, and Briggs, R.O. (2005) Collaboration Engineering: Designing Repeatable Processes for High-Value Collaborative Tasks, *Proceedings of the Hawaii International Conference on System Science*, Los Alamitos, IEEE Computer Society Press.
31. Vreede, G.J., de, Briggs, R.O., and Kolfschoten, G.L. (2006) ThinkLets: A Pattern Language for Facilitated and Practitioner-Guided Collaboration Processes, *International Journal of Computer Applications in Technology*, 25, 2/3, 140-154.
32. Vreede, G.J. de, Davison, R., and Briggs, R.O. (2003) How a Silver Bullet May Lose its Shine - Learning from Failures with Group Support Systems, *Communications of the ACM*, 46, 8, 96-101.