Theoretical Foundations
for
Collaboration Engineering

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus Prof. dr. ir. J.T. Fokkema
voorzitter van het College voor Promoties,
in het openbaar te verdedigen
op 18 december 2007 om 12:30 uur

door
Gwendolyn Laetitia KOLFSCHOTEN
bestuurskundig ingenieur
geboren te Amsterdam.

Dit proefschrift is goedgekeurd door de promotor:

Prof dr.ir. G.J. de Vreede

Samenstelling promotiecommissie:

Rector Magnificus, Technische Universiteit Delft, voorzitter

Prof. dr.ir. G.J. de Vreede, University of Nebraska at Omaha, promotor

Prof. dr. R.O. Briggs, University of Nebraska at Omaha

Prof. dr. H.G. Sol, Technische Universiteit Delft

Prof. dr. S. French, University of Manchester

Prof. dr. J.J.G. van Merrienboer, Open Universiteit Nederland

Prof. dr. W. Veen, Technische Universiteit Delft

Mr. J.C. Jenkins, Hanze Hogeschool Groningen, adviseur

Professor dr. R.O. Briggs heeft als begeleider in belangrijke mate aan de totstandkoming van het proefschrift bijgedragen.

Colofon

Published and distributed by:

Gwendolyn L. Kolfschoten
Department of Systems Engineering
Faculty of Technology Policy and Management
Delft University of Technology
Jaffalaan 5, 2628 BX Delft, the Netherlands
PO box 5015, 2600 GA Delft, the Netherlands
Work: +31 (0)15-2783567
Mobile: +31 (0)6-41270849
E-mail: g.l.kolfschoten@tudelft.nl
www: http://www.tbm.tudelft.nl/webstaf/gwenk

Cover design: Nicoline Kolfschoten
Printed by: Proefschriftmaken.nl

ISBN: 9789056381820

Content

Colofon ... iii
Preface ... vii
Chapter 1. Introduction ... 1
1.1 Collaboration in the knowledge economy ... 1
1.2 What is collaboration ... 2
1.3 Collaboration support ... 5
1.4 Collaboration Engineering ... 7
1.5 Design of collaboration processes ... 10
1.6 Research objective ... 12
Chapter 2. Collaboration Engineering ... 15
2.1 Defining Collaboration Engineering ... 15
2.2 Roles in Collaboration Engineering ... 17
2.2.1 The task of the collaboration engineer ... 19
2.2.2 The task of the practitioner ... 20
2.3 A process overview of the Collaboration Engineering approach ... 22
2.3.1 Investment decision ... 23
2.3.2 Task analysis ... 26
2.3.3 Design ... 27
2.3.4 Transfer ... 27
2.3.5 Implementation ... 28
2.3.6 Sustained use ... 28
2.4 Challenges in Collaboration Engineering: research questions ... 28
Chapter 3. Quality in Collaboration Engineering ... 31
3.1 Quality of collaboration ... 32
3.2 Quality dimensions for collaboration ... 33
3.3 A theory on collaboration ... 37
3.4 A theory of collaboration process design ... 41
3.5 Design support for Collaboration Engineering ... 43
Chapter 4. Research approach ... 45
4.1 The research and application domain ... 45
4.2 The research philosophy ... 46
4.3 The research strategy ... 48
4.4 The research instruments ... 51
4.5 The research outline ... 59
Chapter 5. Transferability: reducing the cognitive load of the practitioner task ... 63
5.1 Theoretical basis: cognitive load theory ... 63
5.1.1 What is cognitive load? ... 63
5.1.2 Cognitive load and transferability ... 65
5.2 Transferability of the practitioner task ... 69
5.3 Increasing transferability: the process prescription template ... 72
5.3.1 Requirements to the process prescription template ... 73
5.3.2 Conceptual design of the process prescription template ... 78
5.3.3 Case study: transferability of the process prescription template ... 82
5.3.4 Conclusions about the process prescription template ... 89
5.4 Increasing transferability: the training approach ... 90
5.4.1 Requirements to the training approach ... 90
5.4.2 Conceptual design of the training approach ... 94
5.4.3 Case study: transferability effect of the training approach ... 95
5.4.4 Conclusions about the training approach ... 101

5.5 Transferability support in Collaboration Engineering 102
Chapter 6. Efficaciousness and acceptance: design support 105
6.1 Theoretical basis: design and engineering 105
6.2 Design in Collaboration Engineering 106
6.3 Increasing efficaciousness and acceptance: design support 109
6.3.1 Requirements to the design support 109
6.3.2 Conceptual design: design support for Collaboration Engineering 120
6.3.3 Case study: evaluation of the design support 131
6.4 Efficaciousness and acceptance support for Collaboration Engineering 138
Chapter 7. Predictability and reusability: the pattern language 141
7.1 Theoretical basis: the pattern language and rule based interventions 142
7.1.1 Design patterns 142
7.1.2 Rule based interventions 145
7.2 Increasing design quality: the thinkLet concept 149
7.2.1 Requirements to the thinkLet concept 149
7.2.2 Conceptual design of the thinkLet concept 152
7.3 ThinkLet content: patterns of collaboration 159
7.4 ThinkLet support for Collaboration Engineering 164
Chapter 8. The effect of Collaboration Engineering on the quality of collaboration 167
8.1 Hypotheses 167
8.2 Research method 167
8.3 Transfer case 1: Half way project evaluation at a ministry 170
8.4 Transfer case 2: Integrity assessment in government organizations 176
8.5 Conclusions from the case studies 182
Chapter 9. Engineerability and transferability 185
9.1 Engineerability 186
9.2 Transferability 188
9.3 Implications for research 188
9.4 Implications for practice 189
9.5 Limitations and future research 190
Appendix 1 The task of the facilitator 193
Appendix 2 Quality of collaboration 196
Appendix 3 Roles in collaboration support 197
Appendix 4 Design challenges 202
Appendix 5 Choice criteria interview protocol 205
Appendix 6 Process prescription example 207
Appendix 7 Lessons learned from the process prescription evaluation 217
Appendix 8 Design evaluation 227
Appendix 9 Master ThinkLet 233
Appendix 10 Research instruments for transfer evaluation 238
References 245
Author index 255
Subject Index 257
Summary 259
Samenvatting 261
Curriculum Vitae 265
Publications 267

Preface

Collaboration is often presented as the solution to numerous problems in business and society. However, collaboration is challenging, and collaboration support is not an off-the-shelf-product. This research offers theoretical foundations for Collaboration Engineering; an approach to design and deploy high value recurring collaborative work practices that can be transferred to practitioners to execute for themselves without ongoing support from professionals. We present a theory on the quality of a collaboration process design for Collaboration Engineering and offer support to design and transfer such process design. The supporting concepts have been evaluated. We found that practitioners were able to facilitate collaboration processes of similar quality as professionals.

This research can be used to support collaboration, but would not have accomplished without support from and collaboration with a number of friend and colleagues that I would like to acknowledge. Firstly I would like to thank Gert-Jan de Vreede and Robert Briggs, for introducing me to the world of academia, for teaching me, for involving me in every possible way, for their patience, and for their friendship. Furthermore, I would like to thank Henk Sol for his advice and support, Simon French and Linda Macaulay, for inviting me to Manchester for a case study, and Jon and Maureen Jenkins for offering their support in collecting data among the International Association of Facilitators.

During my research I had the opportunity to write and collaborate with a number of colleagues. I would like to thank every one of them for their insights and their collaboration: Etienne Rouwette, Eric Santanen, Doug Dean, Fred Niederman, Paul Lowry, Linda Macaulay, Marielle den Hengst, Jaco Appelman, Alexander Verbraeck, Wim Veen, Leah Pietron, Aref Abdulalkarim, Mehruz Kamal, Anita Chakrapani, Kasturi Golla, Pushpa Koneri, Stijn-Pieter van Houten, Edwin Valentin, Peter Jacobs Jeroen Wien, and Sandra van der Hulst.

Several organizations supported my research projects. First of all ‘het buitenhuis’ Jur Kosterbok and Herman Meines supported a case study, in collaboration with Alain Hoekstra and Nicole de Bree at ‘het BIOS’. Next, Bram Castelein supported a case study at ‘de Academie’ at ‘Het ministerie van Sociale Zaken en Werkgelegenheid’ Third, PRC, especially Menno v.d. Velde supported this research and Finally Rene Stad from ING offered his support to a case study. I also had the opportunity to interview a great number of facilitators and professionals in the field of collaboration and collaboration support and to work with many novice facilitators and practitioners that patiently answered all my questions. Many thanks.

During my research I had a lot of fun with my colleagues at the department of Systems Engineering. I thank them for their support and friendship, particularly my roommates Nong Chen and Stijn-Pieter van Houten.

Most of all, I would like to thank Rinske en Richard, Jasper, Nicoline and Ultima, for their love and support.

Delft, November 2007

Gwendolyn Kolfschoten

Chapter 1. Introduction

In this research we will offer theoretical foundations for Collaboration Engineering, an approach to implement collaboration support in organizations. We will first discuss the need for collaboration in society; next we will define collaboration and its challenges. In the third section we will discuss different types of collaboration support, followed by an introduction to the Collaboration Engineering approach discussed in this research. In section five we will discuss the design of collaboration processes. We will end this chapter with the research objective and research questions.

1.1 Collaboration in the knowledge economy

Organizations face the challenge of increasingly complex processes and tasks (Huber, 1984). To deal with this complexity, organizations often need the expertise of several people to solve problems, make decisions, and accomplish tasks. Processes where this is particularly of importance, are innovation, knowledge creation and knowledge activation (putting knowledge to use) (Qureshi and Keen, 2004). Innovation and knowledge creation/activation are important for the competitiveness of organizations (Grossman and Helpman, 1991). Once products or services are on the market, they can be imitated, and competition will evolve. In order to remain competitive, products and the product development process needs constant improvement (innovation). In order to be more innovative than the competition, new knowledge is required. To organize knowledge and to foster productivity and creativity of knowledge workers, group work can be beneficial. The outcomes and results of the knowledge economy often need to be created trough interdisciplinary groups, and consequently require collaboration (Nonaka, 1994, Drucker, 1969). A few examples of such need for interdisciplinary solutions are described below.

A consortium of Finish universities together developed a system for student information. The system has been used for five years, and new functions and updates are required. To get innovative feedback from experienced end-users, they invited them for a collaboration process to gain this feedback and to creatively explore a set of development ideas that can be used to innovate the system (Bragge and Merisalo-Rantanen, 2005).

A group of experts on wind energy had to brainstorm scenario's in which there would be a large opportunity for small-scale wind-energy turbines (Meijers, 2005). This relatively new technology for wind energy can be implemented on the roof of large buildings. Different experts on construction, construction legislation, marketing, the energy market and wind-energy were invited to participate in a collaboration process to develop scenario's in which these turbines could be implemented on a large scale. Scenarios are visions on the future development of trends in society, economy, politics, etc. They represent a possible future and therefore scenario building can be regarded as knowledge creation.

Collaboration and innovation

Collaboration is often essential for innovation; the improvement of products and production processes (Evans and Wolf, 2005). Innovation comes from a creative and problem solving organizational attitude, which can be fostered in an organization where decisions are based on knowledge and experts from different disciplines collaborate to choose the best solution for the complex problems they encounter. Combining the expertise of people is necessary when decisions are too complex for one individual to understand all implications. A design for an organization that fosters collaboration is Mintzberg's adhocracy (Mintzberg, 1983). Mintzberg describes that an adhocracy organization structure causes a pull to collaboration. For instance building the wind-turbine scenarios is a complex, multidisciplinary task. In order to explore such scenarios expertise on technical, political, economic, and societal developments needs to be combined, and thus collaboration among these experts is required.

Collaboration and knowledge creation

Collaboration is often required for knowledge creation in organizations. Knowledge creation is the process in which information is processed to become "justified true belief"(Nonaka, 1994). Knowledge creation can be done with inquiry methods. These are methods to identify and establish new knowledge. Courtney used Churchman's inquiry methods (Churchman, 1971), to explain how organizations can create knowledge (Courtney, 2001). All of these inquiry methods benefit from collaboration and especially the last three because the basis of these inquiry methods is a group process:

- mathematical or logical analysis (Leibniz)
- decision making based on multiple models (Kant)
- consensus building among people, (Locke)
- synthesis of a conflict between people, (Hegel)
- combining multiple perspectives of different people (Singer)

In complex situations and within organizations, often shared responsibility requires a group decision or group analysis, and it is often difficult to make decisions or do analysis without consulting experts. Consensus building, synthesis of conflict and analysis of multiple perspectives require collaboration, as they are all based on the interaction and contribution of multiple individuals.

Consequently, in complex problem solving situations collaboration is essential for the creation of knowledge and innovation, which are described above as two important success factors of organizations in the knowledge economy. Therefore, effective and efficient collaboration is an important factor for the success of such organizations. However, collaboration is never the ultimate goal of an organization; it always has an objective that contributes to organizational value.

1.2 What is collaboration

An often used definition of collaboration from psychology, is from Roschelle and Teasley: Collaboration is *coordinated synchronous activity* that is the result of continued attempt to construct and maintain a *shared conception* of a problem (p. 70) (Roschelle and Teasley, 1995). Another known definition is from Schrage who defines collaboration as the process of shared creation: two or more individuals with complementary skills *interacting* to create a *shared understanding* (Schrage, 1990). In the software and computer science world, definitions of collaboration are: A process in which two or more agents *work together to achieve* shared *goals* (Terveen, 1994) (p.67), and *joint effort* where in each party provides specific products and services *toward* a common *goal* (Beckman et al., 1997) (p. 50).

When we analyze these definitions we identify the following similarities:

- Most definitions specify the activity in collaboration as interaction, working together or joint effort
- Most specify a goal or outcome to which this activity is directed
- Most use an adjective that implies a shared, common status of the goal.

We will discuss these 3 elements of the definitions to determine our own definition.

Joint effort

Joint effort or interaction can involve different levels of interdependency (Thompson, 1967). In collaboration the effort of the individuals is dependent and based on the effort of others. We state that collaboration is a joint effort which implies significant interdependency, as opposed to (combined) individual effort with low interdependency.

Goal

Joint effort among individuals can have a variety of outcomes. However, when effort is not directed or channeled, it is not likely to contribute to the objective of the organization. Direction of effort can be accomplished by goal setting, and goal setting increases effort (Locke and Latham, 1990). Thus, for organizations, joint effort is more useful when directed to a goal. In order to distinguish joint effort from individual effort, we need to set a goal. This allows us to distinguish joint effort, to the same goal, from individual effort, to different goals. Consequently we define collaboration as joint effort towards a goal. Locke and Latham define a goal as a desired state or outcome (Locke and Latham, 1990).

Shared

The definitions above have one more element in common, most use the word shared with respect to the goal. A shared goal means that the people know, understand, and agree to make effort toward the goal that is set. A shared or agreed goal implies full commitment of participants. Although shared and supported goals are more likely when people make joint effort, it is not guaranteed, people can be “forced” to contribute to a goal, or they can contribute because this is required to achieve other goals, and this does not mean that they share such goal. Therefore we will leave adjectives as shared and agreed out of the definition of collaboration.

Thus, for the purpose of this dissertation we define collaboration as *joint effort toward a goal*. This implies that *all participants make effort, combine it (joint) and direct it (towards) to achieve a desired state or outcome (goal).* We do not require that people support or even like the goal. When people direct their effort towards the same goal, we consider their effort collaborative. This accommodates that people make effort to a goal because they are assigned to do so, or because they think the goal is instrumental to other goals. People can have different perceptions of the goal. When progress is made towards the goal, differences in goal-perception can become clear. If conflicting expectations with respect to the goal cannot be resolved, it will be difficult to collaborate. Some group efforts can be focused on identifying goals and required outcomes. Setting a goal, can be a goal on its own, and thus goal setting, as happens in kickoff meetings of projects, can be a collaborative activity.

Using this definition for collaboration in organizations with a focus on innovation and knowledge creation requires us to demarcate a focus on "knowledge oriented" collaboration. This implies that we do not focus on physical collaboration as for instance in sports teams or manufacturing work and collaboration in social activities. Instead we focus on collaboration in a knowledge oriented organizational setting. Furthermore, collaboration can occur on different levels in organizations; between organizations, between departments and between individuals. Our research focuses on collaboration between individuals. Naturally, the goal of the collaboration process can be related to each of these levels.

There are several concepts that have closely related meaning to collaboration, such as coordination and cooperation. If we look at the source of these words, the Latin language, we can easily distinguish these concepts (Harper, 2001).

collaborare *"work with," from* ***com-*** *"with" +* ***labore*** *"to work."*

coordinare *"to set in order, arrange," from* ***com-*** *"together" +* ***ordinatio*** *"arrangement," from* ***ordo*** *"order."*

cooperationem *"a working together," from* ***cooperari*** *"to work together," from* ***com-*** *"with" +* ***operari*** *"to work"*

Thus the difference between collaboration and coordination is that collaboration is focused on joint effort, where coordination is focused on structuring or arranging effort.

Malone and Crowston (1994), established the definition: "Coordination is managing dependencies between activities" after analyzing a set of definitions in literature. A coordinator, is someone who arranges that the efforts of others are inline with the goal while collaborators are part of the group. Coordination is thus a strategy for improving collaboration. Coordination can be encouraging, supporting or structuring and directing collaboration. Coordination, among other factors can have a positive effect on the quality of collaboration, and better collaboration can on its turn have an effect on the quality of organizational outcomes.

Cooperation also differs from collaboration. In the English language, cooperation is used to indicate the effort of the individual to the group result "he cooperated well" means that someone made the expected effort and did not deliberately obstruct other people's effort, where "he collaborated well" means that someone adjusted his effort well to the goal and to the input of others. Being cooperative is an attribute of an individual, while collaborative is an attribute of a group. Jassawalla and Sashittal (1998) (p.252) state that the epistemological domains integration and cooperation fail to adequately capture the complexity or the intensity of cross-functional collaboration.

This dissertation thus focuses on collaboration; joint effort towards a goal. We state that improving collaboration is valuable to organizations, because several success critical organizational outcomes require collaboration. When collaboration is defined as directing joint effort to a goal, then goal achievement, logically is a success factor of collaboration. Collaboration is thus not a goal by itself; it is a process, instrumental to a goal. Increasing the quality of collaboration can thus have two objectives, increasing the quality of the outcome of collaboration (goal) or increasing the quality of collaboration itself (process). We can model the goal as an output factor, which makes joint effort an input factor.

Unfortunately, joint effort as an input factor will not automatically lead to successful collaboration and high-quality results. Groups that collaborate often encounter many challenges. In our wind-turbine example, different experts might represent different stakeholders in the market, and thus conflict can arise, and trust can be broken. People show different behavior in groups, some are dominant, and others are shy and in the background. When all experts represent important stakes within the organization, it is important that the opinion and knowledge of each is taken into account. On the other side of the spectrum, groups with a shared objective can get a tunnel vision of a problem when people are not critical but confirmative (Janis, 1972). Groups, when confronted with complex and knowledge intensive tasks, are often not able to detect and overcome such challenges without support or training (Ellis et al., 1991, Nunamaker et al., 1997, Weiss and Hughes, 2005, DeSanctis and Gallupe, 1987, McGrath, 1991, Schwarz, 1994).

1.3 Collaboration support

Groups might not be able to overcome the challenges of collaboration by themselves (Nunamaker et al., 1997, Schwarz, 1994). Even if groups are able to accomplish their goals, they can often collaborate more efficient and effective using collaboration support (Fjermestad and Hiltz, 2001, Schwarz, 1994). Collaboration support can exist of tools, processes and services that support groups in their joint effort. In knowledge oriented organizations, there is often a need or demand for collaboration support. Collaboration support can be offered by stimulation of increased effort, or by better focusing or directing effort (Briggs, 1994). Collaboration support can be offered in different shapes, such as facilitation, training and tools or technology.

Tools and technology for group support exist in a variety of shapes from complex computer systems (groupware, Group Support Systems), to simple boxes with cards and pencils. Each of these tools can be used by the group to be more successful in sharing their ideas and indicating relations and preferences. However, many tools are not intuitive enough to make sense to groups without any instruction on how to use them. Like in other flow oriented problems, an important part of the solution to go from input to output is process support (Massey et al., 2003). Therefore, tools and technology are often combined with instruction, training or facilitation (Dennis et al., 2001).

Training for collaboration support can help teams by improving their communication skills and can teach them methods to overcome conflict. However, training would only be valuable if the trained team was to collaborate on a frequent basis, and even then, it is costly to train all team members. Training is often focused on changing the behavior of collaborators. For instance, Schwarz offers a set of rules that instruct people to share information and to reduce the chance of personal conflict (Schwarz, 1994). Although these approaches are likely to increase the efficiency of the group, they are not specifically designed to support the group in accomplishing their goal. For this, specific methods for group work such as brainstorming and discussion techniques can be used. There are many of such techniques and it would be very extensive to train teams in all these techniques and their different applications.

As described in section 1.1, many collaboration processes involve interdisciplinary teams. If the composition of those teams changes frequently, the entire organization should be trained in collaborative skills in order to support each team configuration. Therefore, to offer goal-focused collaboration support, facilitation; professional customized collaboration process support, is often more economic. As in many other disciplines, a tradeoff between costs and customization also exists in facilitation; creating a custom, one-off process is more expensive than using of-the-shelf techniques. Professional facilitation requires complex skills and experience to support a group, offering them the appropriate methods to achieve their goal. Becoming a professional facilitator requires extensive training and experience. Good facilitators are likely to be expensive. If an organization wants to reap the benefits of a facilitator (possibly in combination with groupware technology) they can choose to either hire an external facilitator or employ or train an internal facilitator.

Internal facilitation support is difficult to sustain; field studies describe that GSS are much depending on a champion facilitator, once such person is gone, the knowledge disappears, (Munkvold and Anson, 2001, Post, 1993) the skills required are very difficult to transfer. Next, it is difficult to create a business case for GSS and facilitation support implementation in an organization (Agres et al., 2005, Briggs et al., 2003a), although the added value is substantial (Vreede et al., 2003b, Fjermestad and Hiltz, 2001), it is difficult to predict and document this added value. Furthermore, the added value can often be a saving, rather than earnings or could consist of a reduced risk that potentially brings high costs. Such added value can be difficult to allocate as revenue of collaboration support. Collaboration often contributes to important processes in the organization, but not often to the central production process. This makes it easier to eliminate such facilities in a budget crunch (Agres et al., 2005, Briggs et al., 2003a).

As described, to become a professional facilitator require extensive training and skills and thus acquiring or training professional facilitators involves a large investment. As it is difficult enough in organizations to schedule a group meeting, a facilitator should be available, and thus it is hard to be “part-time” facilitator or to combine the function with another function. This requires the allocation of a full-time professional for this task. A facilitator is likely to become more experienced when he encounters more different groups and tasks, while an organization might have very similar and recurring tasks that need support. This can make the task of an internal facilitator monotonous, and might cause facilitators to leave, in which case the investment in training is not repaid. Last, one of the important characteristics of a facilitator is that a facilitator is expected to be neutral (Facilitators, 2004, Niederman et al., 1996). For a member of the organization, it will be more difficult to be objective than for an outsider. Therefore, few organizations have internal all round, professional facilitators, and consequently groups that would benefit from facilitation, will need to hire an external facilitator.

External facilitation would offer a solution for one-off ad-hoc tasks, however, in general and especially for recurring collaborative tasks, there are a number of substantial barriers to hire external facilitation support. First of all, additional costs are added to the project. When used for a recurring task, these costs could mountain up. Next, the budget for collaborative tasks does not always include an external facilitator or collaboration support, and obtaining such a budget often requires bureaucratic procedures. Furthermore, confidentiality might prohibit the group to involve external parties in the collaborative task. This adds to the barrier of hiring an external facilitator. Therefore, organizations are unlikely to hire external facilitators to support recurring collaboration processes that could benefit from such collaboration support.

This can make it difficult for organizations to provide their teams and groups with affordable and accessible qualitative collaboration support to help them to accomplish their goals efficiently and effectively, especially in the case of recurring tasks.

We interviewed 18 people that facilitated groups in their collaborative efforts and were employed to do so (see textbox 4.1 in chapter 4 for method). Out of these 18 facilitators, only 7 considered the organization of collaboration support, their own role and the organization of supporting technology in their organization successful. Most reported challenges were:

- The administrative organization, acquisition of sessions, allocation of facilitators and financial compensation are not arranged well, and the management does not understand the added value of collaboration support
- Facilitators do not get enough time allocated to perform their role and to maintain their skills
- The technology is out-dated or the amount and quality of workstations could be improved
- The knowledge about facilitation is scattered in the organization

1.4 Collaboration Engineering

An approach to address this problem is offered in the emerging field of Collaboration Engineering. The Collaboration Engineering approach, is an approach to designing collaborative work practices for high-value recurring tasks, and deploying those as process prescriptions for practitioners to execute for themselves without ongoing support from professional facilitators (Vreede and Briggs, 2005, Briggs et al., 2006b). Collaboration Engineering intends to enable an improvement in the quality of collaboration for a recurring mission critical task in the organization, as for instance requirements negotiation or risk assessment (Vreede and Briggs, 2005). To solve the problem explained in section 1.3 the Collaboration Engineering approach suggests splitting the role of the facilitator in two roles. Within the Collaboration Engineering approach, a distinction is made between the roles regarding the design and execution of a collaboration process. We distinguish a facilitator, a collaboration engineer and a practitioner (See Figure 1.1).

- A facilitator designs and facilitates collaboration processes. (s)He designs a collaboration process to execute for himself. (s)He is flexible and can adapt his process and activities to uncertain events in a group process. A facilitator should thus be hired for every iteration of the collaboration process.

In the Collaboration Engineering approach, this role is split up. Instead of a facilitator who designs and executes the collaboration process, there are 2 roles; the designer and executor are separate roles.

- A collaboration engineer designs a collaboration process and transfers it to practitioners. This sets different criteria for the design effort. The collaboration engineer cannot expect the practitioner to be flexible. A practitioner does not have the skills to flexibly adapt the process to the situation. Therefore the collaboration engineer should create a very high quality, robust process prescription. A collaboration engineer should be a master facilitator.
- A practitioner is a task specialist in an organization who executes a recurring collaboration process without on-going support from a facilitator or collaboration engineer. A practitioner is not required to have any general facilitation skills or experience and no experience in process design. He is however a domain expert on the content of the recurring collaboration effort. He gets a short training to perform and execute only one specific collaboration process (Vreede and Briggs, 2005).

<table>
<tr><th></th><th>Ad-hoc one off collaboration processes</th><th>Recurring high value collaboration processes</th></tr>
<tr><td>Process design</td><td rowspan="2">Facilitator</td><td>Collaboration engineer</td></tr>
<tr><td>Process execution</td><td>Practitioner</td></tr>
</table>

Figure 1.1 Collaboration support strategies.

The preparation and design of the collaboration process is a difficult step (Clawson and Bostrom, 1995, Hayne, 1999), especially when groupware is involved. In Collaboration Engineering this task is performed by a collaboration engineer. A collaboration engineer designs a collaboration process and documents this in a collaboration process prescription. The Collaboration Engineering process prescription is captured in such a way, that it can be transferred to a domain expert in the organization, called a practitioner. The process prescription is then to be executed without further support from a facilitator or collaboration engineer. A practitioner should be able to guide a group to achieve its goal using the Collaboration Engineering process prescription.

The Collaboration Engineering approach therefore places additional requirements on a collaboration process design, compared to the process design a facilitator creates to execute for himself. Since the practitioner has no design skills, and only limited facilitation skills, the resulting collaboration process prescription should be much more explicit, and should offer sufficient support to practitioners. The design effort of a collaboration engineer and the training of practitioners will consume time, effort and resources, but when the resulting process prescription is re-used and has the intended effect (efficiency, effectiveness, etc.) the organization is likely to gain from this investment.

The requirements for the Collaboration Engineering process design and the design effort to create the process prescription thus are clearly different from the requirements for process designs of facilitators. A facilitator has the skills to improvise when executing the design does not cause the intended effect, but a practitioner is not expected to have design skills, neither will he have facilitation skills, sufficient to deal with unintended effects of the design. Therefore, the design for a Collaboration Engineering process should meet a set of specific quality criteria to make it useable for the practitioner. Such criteria are that it is transferable to practitioner and that it is reusable and useful in several instances of the collaborative task for which it is re-used.

Collaboration Engineering therefore offers collaboration engineers thinkLets. A thinkLet is the smallest unit of intellectual capital to create a pattern of collaboration (Briggs, et al. 2003a). A thinkLet provides a transferable, reusable and predictable building block for the design of a collaboration process. Originally, each thinkLet was described as a combination of tool, configuration and script. A tool is the GSS tool, its configuration is the individual settings in the tool and the script is a set of instructions to the facilitator and the group that describes the method to be used. The script contains everything a facilitator has to do, say, consider and decide to create the required pattern (Briggs et al., 2003a).

Currently, expert facilitators have documented over 70 thinkLets. The design effort is simplified with the use of thinkLets; however, although thinkLets provided a palette of methods to support a group process, the choice between 70 thinkLets to select the ones appropriate to the task, still requires experience. ThinkLets are also used in the training of practitioners (Vreede and Briggs, 2005). They can therefore play a large role in increasing the quality of the collaboration process design and the design effort.

In order to improve the quality of a design we need to be able to directly measure this quality, and we need to know the factors that influence this quality. It is thus important to understand not only what the quality criteria of a collaboration process design for Collaboration Engineering are, but also how we can improve the quality of a collaboration process prescription to better support the practitioner in supporting the group to achieve its goal. To answer these questions we need to study the criteria for a high quality collaboration process design.

1.5 Design of collaboration processes

The importance of the design effort is amplified when Group Support Systems (GSS) are used. A GSS is a suite of software tools for focusing and structuring group deliberation, while reducing the cognitive costs of communication and information access among teams making a joint cognitive effort toward a goal (Davison and Briggs, 2000). Group Support Systems are networks of hardware and software that support groups in their collaborative effort with a set of tools to coordinate their effort. Group Support Systems are a specific type of groupware.

The characteristics of GSS (Bostrom et al., 1992) are:

- **Parallelism.** Participants can add, evaluate or organize topics, ideas, or point of views simultaneously. This means that the disadvantage of waiting for a speaking turn is avoided. The result is a saving on meeting time (Vreede et al., 2003b).
- **Anonymity.** The software does not identify the source of information. This means that each idea is valued on content. Dominance in a meeting is avoided in this way (Nunamaker et al., 1997). However, sometimes responsibility or the need for recognition might require identification of contributions.
- **Electronic recording and representation.** Every idea, topic or point of view is stored. This delivers more accurate and objective minutes, and allows users to use the data processing capacity of computers to aggregate voting results or to analyze text (Bostrom et al., 1992).

Because of these characteristics, various advantages of group work, e.g. synergy, mutual stimulation, knowledge sharing etc. can be enforced. The effect of potential disadvantages e.g. dominance, incomplete use of information, can be diminished. Because of this, a higher level of productivity and a better quality of results can be achieved (Nunamaker et al., 1997).
Field studies on GSS show that GSS are often perceived to be more efficient and effective than manual meetings. Furthermore, participants are more satisfied in a GSS meeting than in a manual meeting (Fjermestad and Hiltz 2001). In a benchmark study where Boeing, ING-NN, IBM and EADS-M were compared, efficiency was improved with more than 50% both on meeting time and on project time and “effectiveness compared to manual” and “user satisfaction” were rated 4.1 on average at a 5-point scale (Vreede, et al. 2003b). This increase in effectiveness, efficiency and satisfaction is caused by a combination of the characteristics of GSS and facilitation. Users believe facilitation to be a critical success factor (Vreede et al., 2002).

In facilitation literature a high quality collaboration process design is considered of high importance. Clawson and Bostrom (1995) questioned facilitators about their roles and tasks and the importance of these tasks, and concluded that preparation and design is the most important task of a facilitator. Unfortunately, we know of only few approaches for the design of collaboration processes published in literature. We will discuss the ones available in GSS literature, which are: GSS interventions, appropriation, apprenticeship or training, and facilitation methods.

Research on specific GSS interventions

In order to design predictable successful collaboration processes, we need to understand the exact effect of a specific GSS tool configuration and the instructions given. A few examples of such studies are: (Garfield et al., 2001, Hender et al., 2002, Santanen et al., 2004). Santanen et al. compared small script variations in a brainstorming technique where the group was inspired with series of prompt from the facilitator. The frequency and combination of prompts had a significant impact on the creativity of the outcome. Hender et al. also studied the impact of different creativity techniques on the amount of new and creative ideas generated (Hender et al., 2002). Garfield et al. (Garfield et al., 2001) studied the effect of stimuli and creativity techniques on idea generation. However, if we look at Fjermestad and Hiltz' meta analysis of experimental GSS research (Fjermestad and Hiltz, 1999) many factors that are measured are rather generic such as quality, time spent, perceptions of effectiveness, satisfaction and usability. Although these are all important outcomes, knowing that the use of a GSS tool increases satisfaction, does not help a designer in using the tool appropriately. More important, the interventions compared are often "with or without GSS and/ or facilitation support" rather than specific instructions to the group. An exception on this is the research based on creativity and brainstorming, which is examined in more detail. Relatively few studies focus on a very specific effect such as (level of) consensus, depth of evaluations, number of arguments, etc. It is with these detailed effects that designers can optimize their process prescription to support goal achievement.

Good guidelines and specific nuances in the process can increase (the quality of) specific outcomes (Santanen et al., 2004). However, the research to formalize and test these guidelines is limited. Without such research it is difficult to give exact guidelines for the design of collaboration process prescriptions that are applicable in multiple domains and situations. Furthermore, there is no overview of such guidelines, which would be useful when different stakeholders have different requirements.

Faithful Appropriation

In order to design a collaboration process that is likely to be successful we need to know when and how to use methods and supporting tools such as GSS (Dennis et al., 2001). Several researchers offer tools and methods to support the design of a collaboration process. Dennis et al (Dennis et al., 2001) describe two important aspects of design; a good fit between the task and the components of the GSS, and faithful appropriation of those components. Faithful appropriation of GSS, described as using the GSS as intended by the designers of the system. However, this might be complicated; the assumptions that lie underneath GSS design are only functional in some group processes, while in other processes do not fit these assumptions (Vreede and Bruijn, 1999). An example is anonymity. While groups that are challenged with dominant participants and large differences in hierarchy might benefit from this feature, groups with intellectually equals might dislike the feature, as they like to be recognized for their contributions. Dennis et al refer to facilitation, training, restriction and guidance as mediators for faithful appropriation. However, they do not point out resources for facilitators to learn to use the system appropriately.

Apprenticeship or training

Yoong (1995) describes training for GSS facilitators. Such training does not only include the basic logic of the products tool options and interface, but also building comfort with the technology and training the ability to map concrete tasks, methodologies and experiences to the product's capabilities. Besides product training, a facilitator should also learn to develop a process structure, to support group interaction and to develop and maintain relationships among the group members. Furthermore, many of the trainings for GSS facilitators assume experience in facilitation and leading meetings (Yoong, 1995). Facilitation is a skill, and thus practice and copying from the expert is important which requires apprenticeship (Post, 1993, Ackermann, 1996) and observation (Beranek et al., 1993). Although this might be valuable approach to train facilitators and collaboration engineers, it is too extensive for practitioners, and it does not offer us explicit guidelines to increase the quality of a collaboration process design.

Summarizing, we can state that GSS support and facilitation of GSS sessions requires extensive training and experience, or even apprenticeship, and that designing the collaboration process is one of the most critical skills that need to be learned. Guidelines to increase the quality of collaboration process design are tacit knowledge and further understanding of such guidelines would benefit the field of GSS research.

Facilitation methods

In the practical literature on facilitation, design is addressed as well. For instance, many facilitation techniques and methods are available in the IAF method database (Jenkins, 2005). This resource for facilitators also supports facilitators in sharing their experience with different methods. Like thinkLets these methods can be used to support the design effort of facilitators. However, we do not know whether these methods are useful for novices and practitioners and there is no assurance or evaluation of the quality of the methods.

Concluding, we can state that both practice and research will benefit from a more detailed understanding of the quality of collaboration process design and guidelines to establish such quality in design, especially when this design should be transferred as a process prescription to practitioners in an organization. Such understanding is critical for the successful implementation of sustained collaboration support according to the Collaboration Engineering approach.

1.6 Research objective

Summarizing the arguments above we found that collaboration is important for knowledge creation and innovation and therefore for the competitiveness of organizations in a knowledge economy (Drucker, 1969, Mintzberg, 1983, Qureshi and

Keen, 2004, Grossman and Helpman, 1991, Nonaka, 1994). We define collaboration as joint effort towards a goal (Harper, 2001). Collaboration is challenging and groups cannot overcome the challenges of collaboration by themselves (Ellis et al., 1991, Nunamaker et al., 1997, Weiss and Hughes, 2005, DeSanctis and Gallupe, 1987, McGrath, 1991, Schwarz, 1994). In order to increase the quality of collaboration, collaboration support is required (Nunamaker et al., 1997, Schwarz, 1994, Dennis et al., 2001).

Collaboration support can be offered by technology and by process support such as facilitation. For recurring tasks, it is difficult for organizations to implement sustained collaboration support (Agres et al., 2005, Briggs et al., 2003a). Collaboration Engineering is an approach where the traditional facilitation role is split up in a design and execution role, which is easier to sustain in organizations (Briggs et al., 2003a). Collaboration Engineering can only be successful if collaboration engineers can design high quality collaboration processes that are transferable to practitioners.

In Collaboration Engineering, the collaboration process design fulfills a key role in affording an organization collaboration support through the training of practitioners. Therefore, this research will aim to create a further understanding in the relation between the quality of collaboration process design and the quality of a collaboration process. Especially it is important to understand how the process prescription can support the practitioner, to be able to replace a professional facilitator for a single recurring collaboration process. Furthermore, we will try to develop guidelines and support for the creation of high quality collaboration process prescriptions that can be executed by practitioners. In this way we want to offer theoretical foundations for the Collaboration Engineering approach and the key challenges in this approach; engineerability and transferability.

The aim of this research is to offer a theoretical foundation for a design and transfer approach in the context of the Collaboration Engineering approach. This theoretical foundation can be used to create high-quality transferable collaboration process designs for practitioners. We will call this product a Collaboration Engineering process prescription. We will first elaborate on the Collaboration Engineering approach and the role of the practitioner. Next, we will inductively derive the factors that constitute quality of collaboration. These factors will be used to create a theory about the quality of collaboration process design, and its contribution to a high quality collaboration process. The resulting theory will be used as a basis to create design and transfer support for the collaboration engineer, and a set of metrics to determine the quality of design and the resulting collaboration process. These will be tested to evaluate the design and transfer challenge in Collaboration Engineering.

The research will help to design high quality collaboration process prescriptions that can be transferred to practitioners to enable sustained collaboration support in organizations. The implication of this research will be to:

- Gain more insight in the requirements from participants, collaboration engineers and practitioners for collaboration support in general, for recurring collaborative tasks.

- Derive theory that supports collaboration engineers to understand the critical challenges in the design and transfer of high quality collaboration processes for Collaboration Engineering
- Use these insights to create design and transfer support for Collaboration Engineering.

Chapter 2. Collaboration Engineering

In order to further understand the quality of a collaboration process design, we first need to understand the context in which such design can be developed and used. In analyzing this process we will identify the key constraints and challenges with respect to this process design.

Collaboration Engineering has developed into a growing research field (Briggs et al., 2003a, Vreede and Briggs, 2005, Santanen et al., 2006). This chapter will explain the Collaboration Engineering approach, its objectives, added value and challenges. We will first define Collaboration Engineering. Second, we will discuss the roles involved, and the tasks assigned to those roles. Third, we will explain the steps of the approach. Last, we will discuss the implications of the approach and the challenges that will be addressed in this research.

2.1 Defining Collaboration Engineering

Collaboration Engineering is an approach to designing collaboration processes. Its aim is to create sustained support for a recurring collaborative task. To indicate both the scope and the key elements of the approach we formulate the following definition for Collaboration Engineering:

Collaboration Engineering is an approach to create ***sustained collaboration support*** *by* ***designing collaborative work practices*** *for* ***high-value recurring tasks****, and* ***deploying*** *those as* ***collaboration process prescriptions*** *for practitioners to execute for themselves without ongoing support from professionals (Briggs et al., 2006b).*

We will explain each of the concepts embedded in this definition as discussed in (Briggs et al., 2006b) in detail:

In Collaboration Engineering we aim to support groups in overcoming the challenges of group work as discussed in chapter 1.3 by offering process and/or technology support in a way that enables the organization to derive value from this collaboration support on an on-going basis without the need to rely on professionals.

- We thus define **sustained collaboration support** as: *an ongoing value derived from process and technology support for groups to achieve their goal that is applied and maintained by members of the organization without support from professionals*.

Design is both a noun and a verb. While designing (verb) is one of the key activities in the Collaboration Engineering approach, a collaboration process design (noun) is the key object in the Collaboration Engineering approach. For reasons of clarity we will label the result of the design effort a collaboration process prescription. Design has many different connotations in a variety of disciplines. For instance, design in architecture is very different than designing education material. Design as a basis for an approach implies a philosophical choice for 'purposefulness' and intention as opposed to randomness and befall. Collaboration Engineering consciously poses itself as an engineering discipline, a discipline where conscious interventions are made to create a specific effect.

- In the context of Collaboration Engineering, **to design** (verb) means *to create, document and validate a prescription for a collaboration process*
- In the context of Collaboration Engineering we define a **collaboration process prescription** (noun) as *an artifact that defines the sequence and logic of a set of activities for attaining some set of goals, and the conditions under which these activities will be executed.*

Collaboration Engineering focuses on the design of collaborative work practices to accomplish a specific type of tasks in an organization; recurring high value tasks. This focus has several reasons. First of all, the return on the resources devoted to the Collaboration Engineering effort increases each time the work practice is executed. Second, if a task does not recur, it would not benefit practitioners to learn a method of execution for the task. Focusing on a recurring task will increase the chance of adoption and sustained use of the collaborative work practice; the individual cost benefit analysis from practitioners and the business case for the adoption of the process will both show a positive net value; the return is high and frequent, the initial investment is low, the variable costs will also be low, and these can easily be allocated to the groups that use the work practice. Additionally, the recurring benefits for a high value task make the work practice important, which decreases the likelihood that it will be abandoned (Agres et al., 2005, Briggs et al., 1999, Briggs et al., 2003a).

- A **work practice** is *a set of actions carried out repeatedly to accomplish a recurring task.*
- A task is **collaborative** if its *successful completion depends on joint effort from multiple individuals.*
- A task is of **high-value** if *the organization derives substantial value or forestalls substantial loss or risk by completing the task successfully*.
- A task is **recurring** if *the task must be conducted frequently in a similar manner*.

This definition implies a first key requirement to the collaboration process design; it should be reusable. If the design is to be used for a recurring task and if it is to be executed by a practitioner without design skills the process has to be reusable in multiple instances of the recurring task.

A second key process in Collaboration Engineering is the deployment of the work practice in the organization through the transfer of a collaboration process prescription, created by a collaboration engineer, to a practitioner, and the implementation of the work practice in the organization.

- **Deployment** means *the training of practitioners to independently execute a Collaboration Engineering process prescription and implementation of a Collaboration Engineering work practice in an organization.*

This insight offers us a second key requirement to the process design; it should be transferable to practitioners.

Based on this definition of Collaboration Engineering, we can now further elaborate on the roles involved in deploying collaboration support through the training of practitioners. We will explain the tasks and responsibilities involved, and the challenges of this task division.

2.2 Roles in Collaboration Engineering

Collaboration support exists of process and (optionally) technology support. For these two types of support we can distinguish a design task (to design the process and the technology), an application task (to apply the process and to use the technology) and a management task (to manage the implementation and control of the process and to manage the maintenance of the technology). A role is a job description that involves the responsibility and execution of one or more tasks. In our interviews (see textbox 4.1 chapter 4 for method) we found that in many organizations only one role is distinguished for collaboration support; a facilitator. The facilitator often does the design and execution of the collaboration process and in many cases also takes care of the project management (e.g. acquisition of sessions, management of facilitation team and business administration) and technology application (operating the technology). External roles are often the design of the technology and the maintenance of the technology (hardware and software maintenance).

In Collaboration Engineering the tasks are divided among several roles which enables outsourcing and dividing the workload of collaboration support (Kolfschoten et al., 2006b). In Collaboration Engineering the collaboration engineer designs a reusable and predictable collaboration process prescription for a recurring task, and transfers the prescription to practitioners to execute for themselves without the ongoing intervention of group process professionals. These practitioners are domain experts, and are trained to become experts in conducting one specific collaboration process, but are not necessarily experts in designing new processes for themselves or others. They execute the designed collaboration process as part of their regular work (Vreede and Briggs, 2005). When using collaboration support technology, the technical execution can be performed by a single practitioner, or two practitioners may work together, one moderating the process while the other runs the technology. However, since this would be a standardized, routine process, there would be no need for skilled professional technical facilitators (also called chauffeurs or technographers) who know all features and functions of the technology platform. Rather, practitioners need to know only the configurations and operations relevant to their specific process. The skills required for the application roles are therefore very light compared to those of the professional facilitator.

Unlike a professional facilitator, the collaboration engineer will not be on hand to correct any deficiencies in the process prescription as it is executed by the practitioner (Kolfschoten et al., 2005). Therefore, the process design skills required by the collaboration engineer are much more extensive than those required by either a facilitator or a practitioner. The processes they create must be well-tested, predictable, reusable and easily transferred to practitioners who are not group process professionals.

In Collaboration Engineering the overall responsibility for the recurring task and the roll-out of the Collaboration Engineering process is mostly not the task of the practitioner but of a process implementation manager. A process implementation manager is responsible for the organization deployment process and for monitoring progress and outcomes. Also the technology is often managed by another person. Most organizations have a special department for technology support and maintenance and such department could also maintain the technology for collaboration support. The new role division is displayed in figure 2.1.

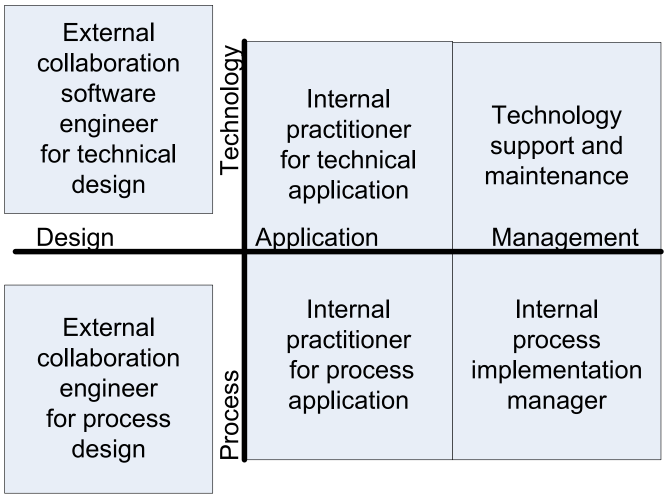


Figure 2.1. Role division in Collaboration Engineering.

In our interviews with facilitators (see textbox 4.1 in chapter 4 for method) we identified the different tasks they performed and asked them to classify themselves in one of the Collaboration Engineering roles. The results are listed in table 2.1. We found only one respondent that indicated that he/she most performed the role of collaboration engineer, however, this person indicated that he/she advised others about their processes design, but did not solely design them for others. Half of the respondents (9) identified themselves as external facilitators. Four of them mostly did technical facilitation. Interesting is that all these chauffeurs indicated that they often offered advice to other facilitators on the design of the process. We found two internal facilitators and one external practitioner, a person that ran collaboration processes for groups outside the organization based on process designs made by others. However, this one practitioner did not run just one recurring process, and also advised others in the design of collaboration processes. Therefore we conclude that the Collaboration Engineering approach is not really found in practice, at least not in relation to GSS supported processes. While facilitators council each other and their apprentices or novices, they do not often design processes for each other, and if they do this is mostly in emergency situation or in situation where a standard process is ran by multiple facilitators or where facilitators work with a team to facilitate a large group.

Question	Total % Yes	Total % No	Total % Other
use GSS	94.4	5.6	-
operate GSS	38.9	0.0	50.0
design GSS	0.0	88.9	-
design processes	88.9	0.0	5.6
use process design from others	47.1	52.9	-
design for others	38.9	5.6	55.6*
role designer	22.2	77.8	-
role internal facilitator	38.9	61.1	-
role internal practitioner	16.7	83.3	-
role external facilitator	55.6	44.4	-
role external practitioner	16.7	83.3	-
role chauffeur	61.1	38.9	-
role collaboration engineer	50.0	50.0	-
Question	**External**	**Internal**	**Both**
support internal/ external groups	66.7	11.1	22.2

Table 2.1. Role separation among interview respondents.
* 'I advice other about their process design.'

To further understand the tasks and responsibilities of the different roles in Collaboration Engineering, especially those that involve the design and application of collaboration support, we can analyze the literature that describes the facilitation task, in relation to the use of GSS. The facilitation task is described extensively in GSS literature (Vreede et al., 2002, Niederman et al., 1996, Dickson et al., 1996, Hayne, 1999, Clawson et al., 1993, Ackermann, 1996). In these descriptions the management tasks (technology support and maintenance, and internal process implementation management) are not described. We clustered these tasks, and divided them into tasks executed by the collaboration engineer and tasks executed (partly) by the practitioner. When both are indicated, the practitioner executes the task supported by the process prescription. Tasks and skills that are only done by the practitioner are thus not directly supported by the process prescription and should be either trained, or they should be used to select practitioners who are already skilled in these tasks. A last resort is to try to avoid the need for these tasks. The overview of facilitation tasks is listed in appendix 1. We will shortly summarize the task division below.

2.2.1 The task of the collaboration engineer

The collaboration engineer designs the collaboration process. He performs the design task of a facilitator. We analyzed what a facilitator does to design and prepare a collaboration process using a survey among experienced facilitators in which we verified whether they performed different steps of a design approach. We determined a generic process for design or problem solving based on design and problem solving approaches in literature such as (Couger, 1995, Ackoff, 1978, Mitroff et al., 1974, Simon, 1973, Checkland, 1981, Sol, 1982). A conceptual aggregation of such problem solving processes is presented in (Vreede and Briggs, 2005). Based on the classical descriptive model of a GSS-based collaboration process described by Nunamaker et al (Nunamaker et al., 1991) we further detailed this process. We asked the facilitators whether they performed each of the steps in the resulting approach, and whether they performed additional steps besides the ones we identified. Furthermore, we asked whether they used the input information described by Nunamaker et al. (See textbox 4.4 in chapter 4 for the method used in the survey). Based on the number of facilitators that indicated they performed each step, we distilled a process for the design of a collaboration process. In figure 2.2 this process is visualized.

The design task of a facilitator consists of analyzing the task and establishing the goal, timeframe, complexity, size and deliverables of the task. He also analyses the group to determine different group characteristics such as its size, context, education level, the stakeholders and institutionalized methods. Next, the design task involves determining the process activities and sub-activities and exploring, evaluating and choosing different facilitation techniques to match the activities in the process. The process prescription is documented with a meeting agenda and a detailed timeframe.

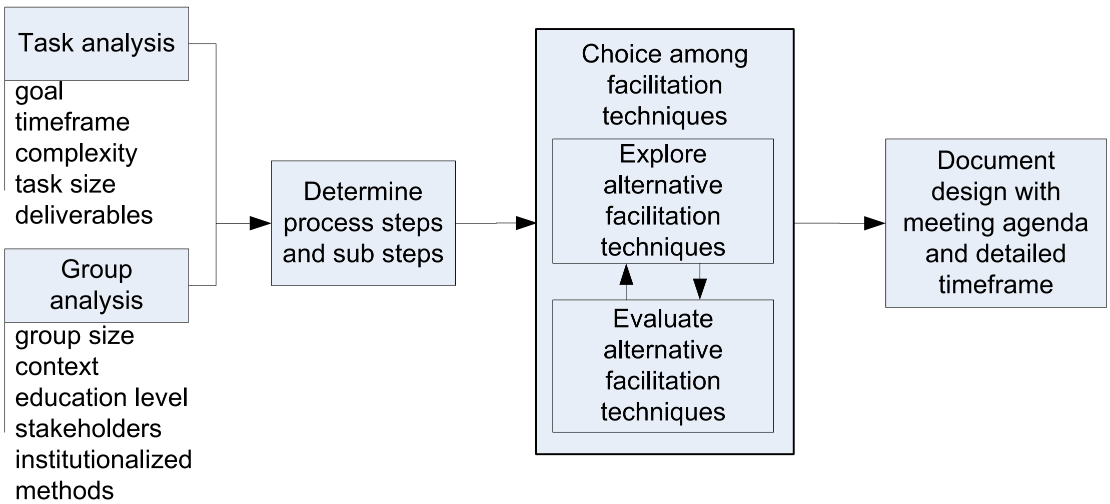


Figure 2.2. Activities performed as part of the design task of a facilitator.

Additional to these facilitation design tasks a collaboration engineer needs to take the capabilities and skills of the practitioner into account. Furthermore, the collaboration engineer should document the design more extensive as a process prescription to support the practitioner during the training and during his/her first executions of the process prescription. Last, a collaboration engineer should put more effort in the validation of a collaboration process than a facilitator; a practitioner needs to get a more predictable and robust process prescription as he/she can not rely on facilitation skills to deal with unintended effects of interventions.

2.2.2 The task of the practitioner

The task of the practitioner can be split in preparing and organizing the different instance of the process, executing the process prescription, and dealing with the dynamics of the group process. (See Appendix 1)

- **Preparation of the instances of the process**
  Preparation deals mostly with the logistics of the process, inviting participants, organizing location and preparing resources, tools and technology as instructed (Vreede et al., 2002, Niederman et al., 1996, Dickson et al., 1996,

Hayne, 1999, Clawson et al., 1993, Ackermann, 1996). Some elements of the process prescription might differ for each instance of the process; these should be filled in for the specific situation, possibly together with the problem owner. For this purpose the practitioner needs to meet with the initiator from the specific group, its leader or the problem owner. Together they discuss the information required as input before the session, and the participants of the session. The script will offer guidelines for this preparation.

- **Execution of the process prescription**
  The task of the practitioner is to execute the process prescription. This task contains activities like (Ackermann, 1996, Vreede et al., 2002, Niederman et al., 1996, Dickson et al., 1996, Hayne, 1999, Clawson et al., 1993):

  - Maintaining focus on outcomes and the goal and guiding the group towards that goal
  - Structuring and focusing discussion
  - Presenting content information, asking questions, interpreting results and decisions, and giving feedback
  - Maintain guiding rules for behavior
  - Ensuring participation of all stakeholders and inclusion of their interests
  - Be sensitive to the group and accommodate their needs
  - Support the use of tools and technology
  - Manage the time
  - Manage roles and responsibilities

- **Dealing with the dynamics of the group process**
  Tasks that are not directly supported by the process prescription but are executed during the collaboration process are (Ackermann, 1996, Vreede et al., 2002, Niederman et al., 1996, Dickson et al., 1996, Hayne, 1999, Clawson et al., 1993):

  - Motivating the group, creating enthusiasm and stimulating ownership
  - Detecting and handling conflict, disagreements and misunderstanding
  - Adapting the process when outcomes are not as expected and the process becomes inappropriate
  - Dealing with emotions
  - Being a leader
  - Resolving unclear issues, questions or results
  - Managing communication and discussion

Most of these task elements are less task- or content related, but focus on the group dynamics and unpredictable elements as conflict, emotions, and motivational issues that occur during the process. Although some frequent occurring challenges and pitfalls can be addressed during training, most of these task elements will come down to the skill and flexibility of the practitioner.

Now that we discriminated between design tasks, performed by an external professional collaboration engineer, and execution tasks, performed by an internal practitioner, we can further explore the way in which the Collaboration Engineering approach proposes the design and deployment of collaboration support in an organization.

2.3 A process overview of the Collaboration Engineering approach

In Collaboration Engineering a collaboration process is designed and deployed for a recurring task. Therefore Collaboration Engineering will have similarities with process engineering, and the approach for deployment will resemble to approaches for organizational change. The general steps in the Collaboration Engineering approach to implement collaboration support in an organization are therefore similar to the phases in a process for organizational process change or business process reengineering (BPR). Davenport (Davenport, 1993) distinguishes two approaches to process change in organizations; radical innovation and incremental improvement. According to Jarvenpaa and Stoddard (1998) the radical approach is useful for design because the activities in a collaboration process are very interrelated and designing separate activities of the process independently is often not possible. However, the implementation of the new work practice can be done in a more incremental way (step by step training small groups with the new approach) to spread costs and to create buy-in for the approach. While Davenport indicates information technology as a key driver for business process change, Grover (1999) suggests we do not assume the use of information technology in business process change.

One of the approaches to business process change that evolved as a response to the evaluations of the radical business process reengineering approach is the approach for business process change described by Kettinger and colleagues, (Kettinger and Teng, 1997, Grover and Kettinger, 1995). Business process change uses the process re-engineering life cycle to describe the process from envisioning to inauguration, to diagnosis, to (re-) design, to (re-) construction and to evaluation. In Collaboration Engineering we will use similar phases and steps to analyze, design, deploy and evaluate the new collaboration process. In Collaboration Engineering, we distinguish, similar to envisioning, an initial state in which the applicability and added value of the approach and the investment is addressed. Next, the design team is established (inauguration) and the task is analyzed. Goal setting, task diagnosis and design can begin. After these often iterative steps, the design is finished, and transfer, piloting and implementation can start. Once the process is implemented it can be adopted by the organization to eventually become a sustained work practice.

Sustained adoption of a (change of) work practice requires that its users perceive an added value of the transfer which is caused, according to Briggs by sufficient added value of the change, sufficient frequency of that value and furthermore, depends on certainty about this added value of transfer through positive testimony and initial experience (Briggs, 2006). The focus on high value recurring processes will likely result in the first two factors, but the initial success of the practitioner will have a large impact on the certainty about this added value. Therefore, the successful transfer of collaboration support skills and thus the quality of the design are of vital importance for sustained adoption.

The 6 steps described above, which can each be decomposed in smaller steps, are visualized, as displayed in figure 2.3 and explained below. As discussed above, the approach will not presume the use of technology for the implementation of a new collaboration process. Furthermore, we will indeed focus on creating a high quality process design in an iterative manner where analysis and decomposition can be revised based on insights from the choices made and in which the design is extensively validated and improved before the full transfer. Once the process design is finished, it can be transferred as a process prescription and implemented. For these phases we will use a more incremental approach. In incremental deployment, the scale of implementation is increased step by step, for instance though the training of small groups of practitioners and by gradually developing and sharing experiences.

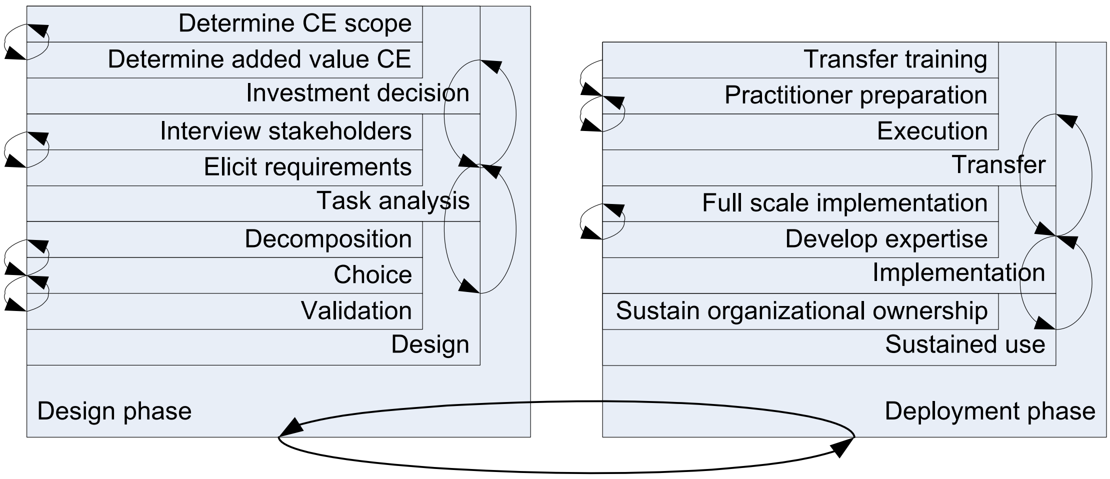


Figure 2.3. The Collaboration Engineering approach.

We will now explain each step of the process in more detail. This explanation has two purposes; explaining the Collaboration Engineering approach and demarcating the research project. The first phase in which we determine whether for a given project the Collaboration Engineering approach is applicable, will be explained in more detail than the other steps because it also explains the demarcation of the focus of this research project. The approach for analysis, design and transfer will be the product of this research and are therefore only briefly addressed. The implementation of collaboration support and the sustained use of it will be outside the scope of this research and is therefore also addressed in less detail.

2.3.1 Investment decision

In order to judge whether the Collaboration Engineering approach will improve collaboration for the anticipated task we need to make two judgments. First the approach should be applicable and second it should offer sufficient added value. The first step in the investment decision involves a check whether the process is part of the Collaboration Engineering scope. The second step addresses the added qualitative and quantitative value of the Collaboration Engineering process.

Collaboration Engineering scope
Collaboration Engineering has a rather distinct scope. This scope has 3 components; and economic component, the collaboration component and the domain of application.

Economic scope
Collaboration Engineering focuses on high value recurring tasks in the organization (Vreede and Briggs, 2005). As described above this enables recurring revenue from a single design and training investment. With this focus the success of the process change will be valuable for the organization, and this value will be created on a recurring basis. Therefore, it is more likely that stakeholders will stimulate the use of the process and guard that conditions for successful deployment of the process are in place. This will increase the likelihood of sustained use (Vreede and Briggs, 2005). If a process does not fit this economic scope, a Collaboration Engineering approach might fail or might not render sufficient revenue.

Collaboration scope
Not all group processes are collaborative tasks. Often a group process involves mostly one-way communication, such as a presentation, or a collective survey, where people are given information, or asked for information, but where there is no exchange of information. In these cases there is no (need for a) group goal, and thus no need for collaboration. Collaborative tasks involve interaction, discussion, evaluation, shared understanding, decision making, consensus building, etc. Based on the patterns of collaboration (Vreede and Briggs, 2005, Briggs et al., 2006b) we can limit this scope to (a combination of):

- ***Generate***: Move from having fewer to having more concepts in the pool of concepts shared by the group
- ***Reduce***: Move from having many concepts to a focus on fewer concepts that the group deems worthy of further attention
- ***Clarify***: Move from having less to having more shared understanding of concepts and of the words and phrases used to express them.
- ***Organize***: Move from less to more understanding of the relationships among concepts the group is considering
- ***Evaluate***: Move from less to more understanding of the relative value of the concepts under consideration
- ***Build consensus***: Move from having fewer to having more group members who are willing to commit to a proposal.

Application domain
Collaboration Engineering can be applied to many domains. However, as apparent in the patterns of collaboration, Collaboration Engineering is applied to knowledge intensive processes that require cognitive effort. It does not involve collaborative physical effort. Also Collaboration Engineering is applied in result-focused tasks. This means that it is not suitable to design processes to change behavior or build relations between people (teambuilding) or to train social or knowledge related skills. For instance, Collaboration Engineering can be used for conflict solving, but it will focus on the content of the conflict, not on training the people involved in conflict

resolution skills. Disturbing behavior and emotions are only dealt with when they interfere with the process to goal achievement.

Added value

The added value of the collaboration process can again involve 3 components. A financial component due to efficiency of the process, a quality increase in the results, and added value in the form of e.g. agreements, consensus, support, awareness and shared understanding, that will improve the quality of the results of future work among the stakeholders involved. Similar results are found when the use of Group Support Systems is researched (Vreede et al., 2003b). Most of these results can also be achieved without such system, but with a clear goal and a designed and facilitated process.

Financial added value

Especially a collaboration process can be designed to be more efficient trough parallel working, and through the use of a focused approach (Briggs, 1994). An example is a process that requires generation of a set of solutions and an evaluation of those solutions by several experts in the organization. As an alternative to a collaborative process, the experts can be interviewed to get a list of solutions; these are integrated by the interviewer, and presented in a second interview to perform the evaluation. In a parallel work process the group can brainstorm ideas and elaborate on them in the group, converge to a list of shared solutions, and evaluate these solutions on several criteria, both quantitative and qualitative. Below in table 2.1 is an estimation of the time frame of both approaches when 10 experts and 1 interviewer/practitioner are involved, and when the interviewer/practitioner is dedicated to the project.

Interview approach			***Collaborative approach***		
Activity	***Man hours***	***Project time frame***	***Activity***	***Man hours***	***Project time frame***
Plan interviews	2	3 days	Plan meeting	2	3 days
Passing time		1 week	Passing time		2 weeks
Interview 1 (1h)	20	1 week	Meeting (3h)	33	1 day
Working trough results	8	1 day			
Interview 2 (1h)	20	1 week			
Work out results	8	1 day	Work out results	8	1 day
Total man hours	**58**		**Total man hours**	**43**	
Total project time		**4 weeks**	**Total project time**		**3 weeks**

Table 2.2. Interview approach versus collaborative approach.

In this way, a large amount of man-hours are saved for the interviewer, but the participants need to spend one hour more. In project time also savings are made, because both activities are done at once, and in one day, although planning a meeting with 10 people might take some additional time.

Quality increase of results

Besides financial added value, the results of the collaborative approach are likely to score higher on several quality indicators. If we again look at the example, the collaborative approach is also likely to have higher quality results. First, the brainstorm is interactive; therefore experts will elaborate on each others ideas, and will be inspired by each other, which can result in more creative and better solutions. Second, the group convergence will give a shared group result instead of a (possibly biased) summary of the interviewer. Furthermore, it will resolve misunderstandings and create a shared language and more support for the solutions. Last, the evaluation can be rater extensive, the results of the evaluation will be directly available, and can be discussed shortly to gain increased consensus. This might result in a better solution, and a better decision about the possible solutions.

Additional added value
Besides the quality improvement of the results there are some gains in a collaborative approach that are more tacit such as a team bond, awareness of the problem, consensus on decisions, a shared language, higher chance on commitment to the implementation, and support for the results. In the example, the results of the collaborative approach might also improve the results of the experts in the future; they now have more of a shared language and they learned to look at the problem and solutions from different perspectives. Furthermore, the results are shared and owned by the group, and people will feel that their ideas are considered which might make them more willing to use and re-confirm the results in follow-up activities. Last, because the group was involved in the whole process, they might feel more responsible for follow-up tasks and they might be more inclined to use and implement the results.

Once it is determined that the task is supported by a collaborative approach, and that the project of changing this approach will benefit from the Collaboration Engineering approach, the task needs to be analyzed further in order to derive the requirements and constraints to the collaboration support and the collaboration process design.

2.3.2 Task analysis

In the task analysis phase a team is created with stakeholders from the organization among which the project manager of the Collaboration Engineering project. The team analyzes the task and defines the goal, deliverable and other requirements. Interviews or meetings with the relevant stakeholders will give insight in the goal and task. A goal can be to deliver a tangible result as for instance, to make a decision, to solve a problem, but it can also be a state or group experience, like increasing awareness about a problem or creating shared understanding. Deliverables therefore can be very straight forward, but in some cases require strong demarcation. In other cases it is important that specific requirements to the deliverables like the level of detail of a solution or the level of consensus with respect to a decision are accommodated in the design. Other requirements that need to be defined include group, context, technology and the skill level of the practitioner (s). Depending on the amount of practitioners, the design can be adjusted to the preferences and skills of the practitioner, or the design can be based on a practitioner profile, that is later used to recruit practitioners in the organization. Very important in this step is to establish the parameters that need to be instantiated individually for each specific instance in which the process should be used. The design should be flexible with respect to those aspects. Also it is important to develop some metrics to be able to assess the quality of the results and the process. Once the requirements and constraints for collaboration support are determined, we can start to design the collaboration process.

2.3.3 Design

In this phase the collaboration process is build based on the requirements established in the analysis. The approach for collaboration process design will resemble a design approach or problem solving method as described for instance by (Ackoff, 1978, Mitroff, et al. 1974, Simon, 1973, 1960, Checkland, 1981), with one key difference; instead of identifying solutions or alternatives from scratch, a library of known techniques is used as a source to select, combine techniques in a collaboration process design. There are three key steps in the design phase, the decomposition of the process in small activities, the choice of facilitation techniques for each activity and the validation of the design.

To achieve the goal, the group has to go through several activities. In order to determine these activities the process needs to be decomposed. To design a process that enables the group to accomplish each of the activities, facilitation techniques are used. The Collaboration Engineering design approach uses thinkLets to design collaboration processes. ThinkLets are formal, documented, reusable and predictable facilitation interventions, used to consciously create predictable patterns of collaboration. A thinkLet is the smallest unit of intellectual capital to create a pattern of collaboration. A thinkLet provides a transferable, reusable and predictable building block for the design of a collaboration process (Vreede and Briggs, 2005). Currently, expert facilitators have documented over 70 thinkLets. Experience has shown that practitioners and novice facilitators can use thinkLets and indeed create the intended patterns of collaboration (Vreede and Briggs, 2005, Kolfschoten et al., 2004a). The choice of a thinkLet can be made based on the pattern of collaboration it creates, its result and the previous and next thinkLet in the process (Kolfschoten et al., 2004b). Finally the design can be validated based on several criteria. Important is that the process is realistic in terms of the results, abilities of the group, abilities of the practitioner, and constraints of the resources available. Approaches to validation include a pilot session or walk-through to try the process prescription on a real group or a few stakeholders, expert validation by colleagues, and simulation with a limited number of sample contributions. The design steps have an iterative nature, similar to iterative approaches in software engineering (e.g. Boehm’s spiral model (Boehm, 1988)). The validation is however a key gate in the process; it is critical that the design has sufficient quality since flaws will result in unsuccessful transfer to practitioners, which could lead to abandonment of the project. However, insights in the transfer phase might again result in revisions to the process design.

2.3.4 Transfer

In the transfer phase, the collaboration engineer transfers the collaboration process prescription to the practitioner. This step will contain three important learning efforts. The first occurs in the training in which the practitioners learn to execute the collaboration process. The second occurs when the practitioner prepares himself for a first application of the process. He then has to apply the process prescription to a specific group in his organization and needs to prepare and instantiate different aspects of the process prescription. The last learning effort occurs in the first trials of the collaboration process execution. These might reveal problems and difficulties that require adjustments and refinements of the design. Although the collaboration engineer himself might run pilots of the process to test it, the first trials of the practitioners might reveal different problems, and thus can also lead to changes in the design. After one or more iterations of the process, the practitioners will increase their skills and expertise and will be able to run the (revised) process independently.

2.3.5 Implementation

When the transfer phase is complete the process can be implemented on a full scale, and experience is gained. This requires managerial activities, planning and organization. Like in facilitation, the success of the practitioner is key to the successful implementation of the process (Vreede et al., 2003a, Nunamaker et al., 1997). When practitioners fail they will lose credibility, the results will be of insufficient quality, the process might be inefficient and this might result in abandonment of the process and a large waste of resources. If we look back at the interview example, the effect of 1 bad interview is less likely to escalate than the failure of a collaborative process:

- Firstly, only two man-hours are wasted compared to thirty three in the collaboration process.
- Secondly, the interviewer has a bad reputation with only one expert, while the practitioner will have a bad reputation with all experts.
- Thirdly, the self-esteem of the practitioner will be more damaged in the collaboration process, since the group will witness his/her failure.
- Fourth, the interviewer may learn from the problem, and may do better in the next situation. Thus, the results of the 9 following interviews may compensate the loss at interview 1, while the practitioner might not get a second chance from the group and cannot compensate for his/her mistakes.

2.3.6 Sustained use

When the practitioners are trained and performed well at their first sessions the process should be rolled out in the organization and the organization should slowly take ownership of the process. To establish this, the management should stimulate the use of the collaboration process through controls and incentives. Additional, it should lobby for support from team leaders and lower level managers in the organization. Furthermore, when the project involves multiple practitioners, it may be valuable to set-up a community of practice to exchange experiences and lessons learned. Last, it is important that the process and its benefits are evaluated on a regular basis.

Naturally this deployment phase is a more incremental phase. Practitioners can be trained in small groups and they will not be able to conduct their first sessions all at once, thus, during a period of time the old and the new work practice will be used in parallel. In time when more controls and incentives are in place the process will become a standard in the organization.

2.4 Challenges in Collaboration Engineering: research questions

As we explained, in chapter 1, and in the first section of this chapter, the benefits of the Collaboration Engineering approach can have a large impact on an organizational scale. However, the challenge of Collaboration Engineering, became clear in the last steps of the process. A practitioner is a domain expert, without facilitation skills. Training the practitioner is an additional investment, and therefore needs to be done with a limited budget, which will only allow for a short training. Furthermore, initial success of the practitioner is essential for sustained adoption of the process. Therefore, the process prescription has to overcome the lack of facilitation skills, should be easy to transfer, and should be easy to understand, learn and use. Last, the process prescription should be as robust as possible, it should involve little risks, limited uncertainty and as few choices as possible. Additional, the process prescription should be flexible enough to be useful in different instances of the collaboration process. Therefore, a key requirement is that the process prescription is reusable for the different instances in which the task recurs.

Given the timeframe of this research, a full study on the deployment of a collaboration process design and the effect of the collaboration process design and practitioner deployment is not feasible. As discussed, certainty about the added value of the process prescription and positive testimony is critical for adoption and sustained use. Therefore, the quality of the collaboration process when executed by a practitioner for the first time is an important first indicator of the effect of the Collaboration Engineering approach on sustained and successful of collaboration support.

Given this demarcation, two key challenges of Collaboration Engineering need to be addressed in this research; the design of a robust process prescription and the successful transfer of collaboration support to practitioners, so they can use it to support groups without the support of professionals. The training of the practitioner is focused on the transfer of a single reusable process prescription for a collaborative work practice, with the use of collaboration support (tools and techniques). The collaboration process prescription has a key role in the Collaboration Engineering approach; it is the result of the design phase, and the basis for transfer. To support both the collaboration process prescription should be of high quality. To further understand the quality dimensions of the design we need to determine what high quality collaboration is, and how we can create a collaboration process prescription that offers sufficient support to the practitioner.

The objective of this research is therefore to identify, define, operationalize and test the quality dimensions of collaboration process design to create collaboration process prescriptions that can be transferred to practitioners in an organization. This implies the following research questions:

1. *What are the quality dimensions of a collaboration process design that is transferred as collaboration process prescription to be executed by a practitioner, and how can we define these quality dimensions?*
2. *Knowing these dimensions, how can they be operationalized to optimize the quality of the design, to increase practitioner performance and therewith the success of the collaborative effort?*
3. *Does the use of the design and transfer support indeed enable the support of a collaboration process by practitioners with professional quality?*

To summarize, we need theoretical foundations to overcome two key challenges in the Collaboration Engineering approach; transferability and engineerability

Chapter 3. Quality in Collaboration Engineering

Literature on collaboration support emphasizes the importance of design or preparation for the success of collaboration processes (Clawson et al., 1993, Nunamaker et al., 1997, Hayne, 1999). To properly design a collaboration process we first need to understand what "quality of design" and its objective "quality of collaboration" is, and how it can be operationalized. The purpose of a collaboration process design is to consider, predict and anticipate what will happen in the collaboration process, to improve the quality of the resulting collaboration process (Clawson and Bostrom, 1995, Nunamaker et al., 1997). This implies that, a high quality collaboration process design, should, when executed as intended, create a successful collaboration process. Quality is a container concept; it needs to be further defined in order to be measurable, and to enable deliberate quality improvement. As we discussed in chapter 1, the success of the collaboration process can be measured with respect to the outcomes (goal achievement) and the process itself, and both as an observation and as a perception of the people involved. In this chapter we will further explore the "quality of design" and "quality of collaboration" in the light of Collaboration Engineering to derive a theory to develop support for the design of transferable collaboration process prescriptions.

Based on the definitions and scope of Collaboration Engineering in the previous chapter we can state that the main purpose of the Collaboration Engineering process design is to allow the practitioner, a domain expert without facilitation experience, to offer collaboration support for a group (groups) to execute a recurring high value task. The critical function of the resulting collaboration process prescription is thus "enabling the practitioner to support the group in achieving its goal." But the design also prescribes the process that the group will use to achieve the goal and therefore it has both a direct and an indirect effect on the quality of the collaboration process. Based on this first analysis we could define the quality of a collaboration process design for Collaboration Engineering as *the degree to which the Collaboration Engineering design supports a practitioner to support the group in achieving its goal.* However, with this definition we did not yet break the concepts "goal achievement" and "ability to support the practitioner" open. We need to further understand the effect of the design on the collaboration process, its outcomes, and the role of the practitioner in this process, to be able to specify these dimensions further. When we understand the dimensions of 'quality of design' and how it affects 'quality of collaboration' we can use the dimensions of 'quality of design' as a theoretical basis for the design support that will be developed in this study. For this purpose we will first analyze the literature and survey facilitators with respect to quality and success of collaboration in general. From this analysis we will derive the main quality dimensions and distinguish dimensions of 'quality of collaboration'. This will help us to further define and operationalize the quality of a collaboration process design for Collaboration Engineering, and to derive theories on quality of collaboration and quality of a collaboration process design from a Collaboration Engineering perspective. Using these insights in the quality dimensions of a collaboration process design we can build design support to increase the quality of a collaboration process design. The dimensions of ‘quality of collaboration’ can be used to evaluate the effect of the design.

3.1 Quality of collaboration

Quality of collaboration thus has a process and a result component. *We define quality of collaboration as appreciation of joint effort by relevant stakeholders*. A first approach to analyze the quality dimensions of collaboration is to look at the outcomes of interest for research on group processes. A good source for this analysis is the use of meta-analysis studies in GSS research in which outcomes studied in this domain are listed. We analyzed the meta analyses by: (Fjermestad and Hiltz, 1999, 2001, McLeod, 1992, Baltes et al., 2002, Hwang, 1998, Tyran and Sheperherd, 1998, Dennis and Wixom, 2002) (see appendix 2). The study by Fjermestad and Hiltz is by far the most complete overview of output factors of interest. We removed the factors that are domain or context specific such as for instance ‘consensus’ and ‘agreement’ not all collaborative efforts need to result in a decision, but this can be part of the goal. Furthermore, we used the summarizing concepts of Fjermestad and Hiltz, efficiency, effectiveness and satisfaction. A striking factor that was listed in each of the meta analysis papers was participation. In order for a collaboration process to be successful, the group members need to participate. However, we want to strengthen this concept, we think that group members do not just participate, make effort towards the goal, they also spend other resources to the process such as time, effort, knowledge and physical resources. We will therefore label this dimension commitment of resources, among which making effort; participation is one type. Furthermore, we found under the dimension satisfaction factors that indicate satisfaction with the process and satisfaction with the result.

These dimensions may at times be conflicting or incompatible. For instance, if the outcome of a collaboration process has large consequences, effectiveness is more important than efficiency. Furthermore, perceptions on the quality of a collaboration process can differ among participants or stakeholders. For example, the problem owner can be very pleased with the results of a session, while the participants did not feel that all their interests were taken into account. Quality thus can be measured from different perspectives, and it varies based on the goal of the process and the intentions of the stakeholders with respect to the results.

This suggests that creating a high quality collaboration process will require the designer to make trade-offs between several important requirements and perspectives, and often it is the case that an increase of quality on one dimension decreases quality on another dimension. Therefore there is no independent objective quality measure for collaboration processes. However, we can use the quality dimension suggested above to describe the quality of a collaboration process. This will allow us to further specify the effect of the collaboration processes design on the quality of collaboration.

To corroborate these dimensions we did a second analysis. In this second analysis we asked facilitators as part of an interview (see chapter 4 textbox 4.3 for method) what they considered high quality collaboration. We condensed their answers to the following list of dimensions

- Goal/results are achieved
- Participants are committed to the goal and willing to contribute
- Participants are satisfied with the results and support them
- Process is efficient and participants were satisfied with the process
- People are willing to share and compromise
- People listen to each other
- Participants feel their contribution was useful
- Focus on the goal
- There is a bond developed, respect and trust in the group
- Participants feel free to contribute
- Fun
- Mutual learning

Some dimensions are specific versions of other like mutual learning and a team bond can be considered a type of result that should be achieved and fun and focus could be seen as an indicator of efficiency or process satisfaction. Commitment and willingness to share are also similar. Willingness to compromise and listening were mentioned several times. Listening is one specific type of effort required from participants and thus part of commitment. Willingness to compromise is more difficult. Also more complex are trust and feeling free to contribute, we think that these factors will be reflected in the perceived satisfaction of the participants. The other dimensions are similar to the dimensions we found in the meta analysis, however, some of the dimensions are indicated from different perspectives, such as goal achievement as in effectiveness in general, goal support from the participants, quality of results and satisfaction with the results.

3.2 Quality dimensions for collaboration

In order to further understand the different dimensions and perspectives on quality of collaboration we will discuss each of the dimensions from literature below to reflect on their applicability in explaining the quality of a collaboration process, and the effect of high quality design on the quality of a collaboration process. Then we will discuss the relations among these concepts. From this analysis we will derive the theory on the quality of collaboration and quality of design.

Group effectiveness

An important indicator for the quality of a collaboration process is that the group goal, specified in the design, is achieved and that the results of the collaboration process meet the requirements set in the design. A frequent requirement to outcomes is that the group achieves a consensus about an issue or proposal, other requirements can be that a decision is made, or that a deliverable is useful for its purpose. However, not all collaboration processes require consensus, nor do all collaboration processes lead to decisions. Therefore, such dimensions cannot be regarded as generic quality indicator. Effectiveness, is however generic enough, and will in all collaboration processes be an indicator of quality. We define effectiveness according to In 't Veld (Veld, 1987) as *the real result compared to the intended result, specified in the design*.

Effectiveness can be seen as a continuum, it is the extent to which a result serves accomplishment of the goal set for the collaboration process. However, the stakeholders that specify the intended result (goal) each have a different perspectives and interests. Therefore, different stakeholders can have a different perception of the effectiveness of the collaboration process, based on their expectation, and the way they value the results of the group effort.

This is visualized in figure 3.1.

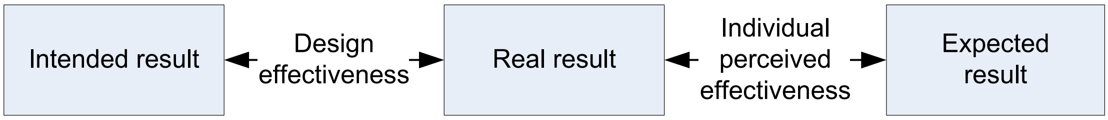


Figure 3.1. Effectiveness: each box represents a variable; the arrow labels are the quality dimensions, the arrows point towards the variables that determine the quality dimensions.

To decrease the variations in effectiveness perceptions and the intention specified in the design, we can increase the level of detail of the goal specification (Locke and Latham, 1990). The more specific the shared requirements to the results, the more focused and specific the group effort. This can lower the difference in perceived effectiveness of the process. Another dimension that influences this variation is goal congruence. If the goals and stakes of participants differ with the group goal, expectations and the utility of the results for the participants are also more likely to differ.

Group efficiency

There are several important resources that stakeholders can "spend" in a collaboration process; they can spend time, they can offer and share information or knowledge, they can make effort, focus attention and they can spend physical resources such as a money and meeting facilities. We call this factor expense of resources. For a given task, there is a specific budget of resources, the intended resource expense, which are specified in the design. Stakeholders also have an expectation about the expense they need to make. Based on the task description, comparable efforts and their expectation of the input of others they will estimate the resource expense they have to make to achieve the goal. High group efficiency occurs when fewer resources are spent than intended. We therefore define efficiency according to In 't Veld (Veld, 1987) as *the difference in the net amount of resources used (real resource expense) compared to the planned amount of resources (intended resource expense).*

Like effectiveness, efficiency is also a scale. The required resources can be determined by estimation, priority of the task compared to other tasks and the effort from the participants. However, most of the resources are controlled by the participants. Therefore, planning or allocating resources is difficult. While it is often possible to allocate time and physical resources, it will be more difficult to plan the amount of effort people make, the attention they focus on the process and the knowledge they are willing to share.

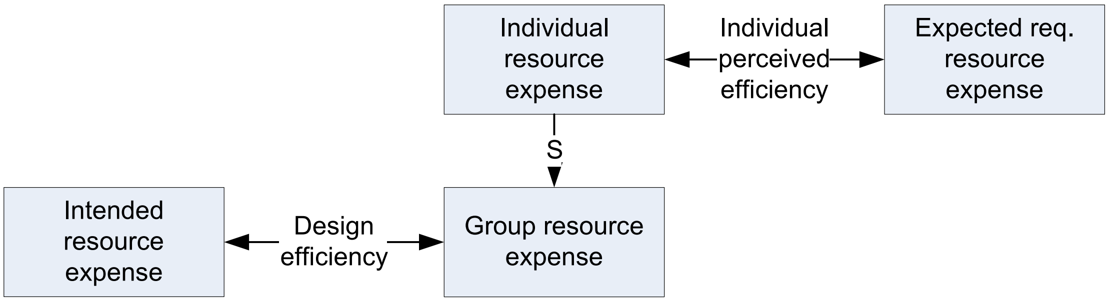


Figure 3.2. Efficiency: each box represents a variable; the arrow labels are the quality dimensions, the arrows point towards the variables that determine the quality dimensions.

Linking expense of resources and results: Group productivity

The design specifies both the intended result and the intended expense of resources. The objective of the design is that when the resources are spent as specified in the process prescription, the use of resources results in outcomes that fit the requirements for goal achievement. Thus, if we want assess the quality of the design we need to know, not only whether the results meet the requirements and whether resources are spend as intended, we also need to assess whether the results justify this expense. This is indicated with productivity. Productivity indicates whether *the results are in balance with the expense of resources*.

Productivity thus increases when fewer resources are spent or when the result is better or both. In the design productivity is specified in the intended result and resource expense. The real result can meet or exceed both intentions and expectations. Productivity can also be compared to a norm. Figure 3.4 visualizes productivity.

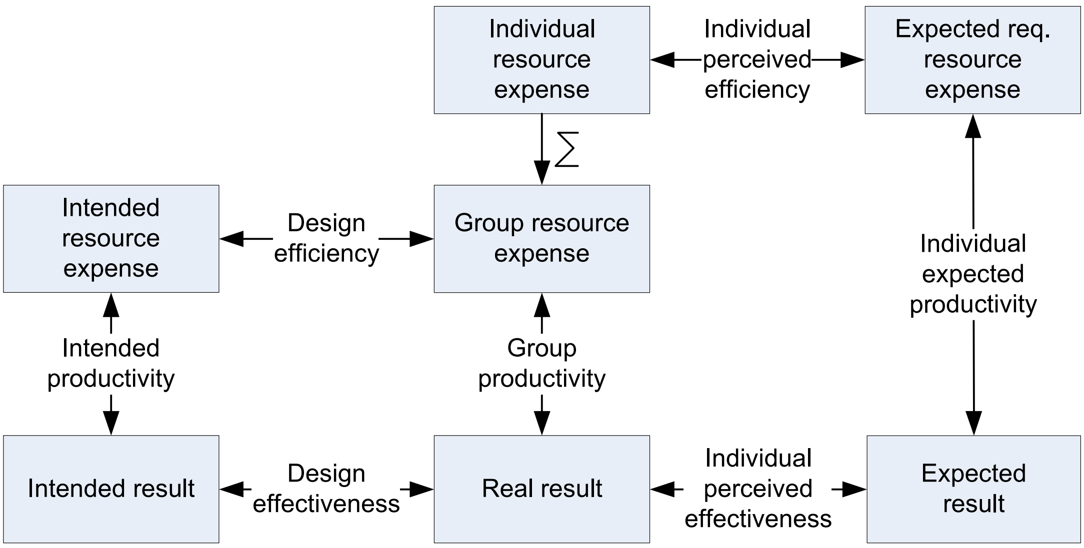


Figure 3.3. Group productivity: each box represents a variable; the arrow labels are the quality dimensions, the arrows point towards the variables that determine the quality dimensions.

Commitment of resources to the group goal

In order to be productive, the resources that are needed should be available. Stakeholders and participants in the collaboration process own the resources. As they have a stake in the process or outcome; they are somewhat willing to spend them. Efficiency indicates the required budget, compared to the real resource expense. In the ideal situation, the resources needed are allocated for the task and available. However, in reality, resources are scarce and are divided among several tasks. Furthermore, some resources such as effort and information are more difficult to allocate. An organization can assign a person to a task, but the amount of effort made, is also related to the motivation, expectations, skill and abilities of the person. Resources required do not only imply an amount, but also a type of resources. For example, to solve a problem, more information or effort does not necessarily lead to a better result; the right information and the effort of the right people is required to accomplish the results. To accommodate these dimensions we introduce the dimension 'commitment'. Commitment is researched especially in the context of human resource management, by John Meyer and colleagues. Mayer and Herscovitch did a review on commitment definitions to derive a general model of commitment. They define commitment as "a force that binds an individual to a course of action of relevance to one or more targets" (Meyer and Herscovitch, 2001 p. 301) Since we defined collaboration as joint effort towards a goal, we want to use the construct 'goal' rather than 'target', and we will focus in this perspective on the act of commitment as the 'expense of resources' instead of a 'cause of action'. We will then define commitment as *a force that binds an individual to spend resources(time, effort, knowledge and physical resources) to achieve the group goal* When designing collaboration process we will assume that the force that binds the individual to spend resources to achieve the group goal is not external pressure but rather willingness of the individual (Briggs et al., 2005). It is therefore an individual judgment based on the expectations with respect to the result, the required resource expense and the expected commitment of others. However, in some situations other forces might press the individual to commit. It will require an ethical judgment to deal with these forces.

To visualize commitment we can create the following model:

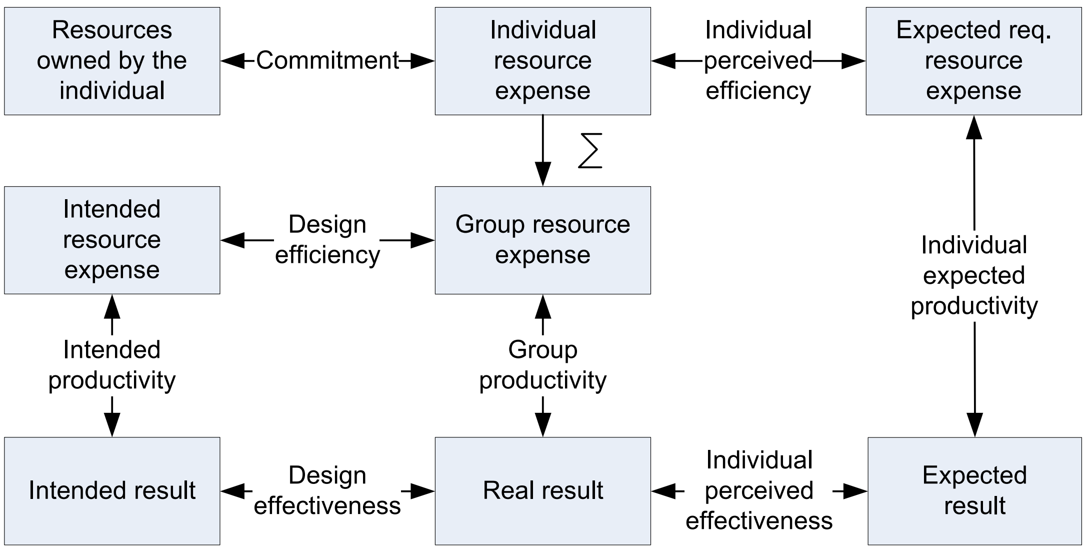


Figure 3.4. Commitment related to productivity: each box represents a variable; the arrow labels are the quality dimensions, the arrows point towards the variables that determine the quality dimensions.

Participant satisfaction

Satisfaction can be defined in different ways. Two important distinctions are satisfaction as an emotional response and satisfaction as a judgment (Briggs et al., 2004).

Emotional Satisfaction is an emotional response as a result of a perceived shift in yield with respect to personal goals (Briggs et al., 2004). Such yield is the result of a shift in utility with respect to goal attainment and/or a shift in likelihood of goal attainment. There can be many personal goals that constitute the reason for a participant to contribute in a collaboration process other than the group goal. Yield Shift Theory (YST) defines the Satisfaction Response as a valanced affective arousal (emotion) with respect to the attainment of individual goals (Briggs et al., 2004). YST posits unconscious cognitive mechanisms that automatically ascribes utility to the attainment of a goal, and automatically assesses the likelihood that a goal may be obtained. It posits that when an individual perceives a change in either the utility or likelihood of attaining a goal, a subconscious mechanism initiates an emotional response proportional to and in the direction of those shifts. Thus, if the individual perceives a positive shift in utility or likelihood, a positive emotion manifests; if the individual perceives a negative shift in utility, a negative emotion manifests. If a person perceives that the likelihood or utility of an individual goal are advanced by the results of the group effort, then a positive satisfaction response is likely to manifest.

Judgmental Satisfaction has to do with the individual cost-benefit analysis of the result and the resource expense, and can be expressed in the perceptions of efficiency, effectiveness and productivity, as shown in figure 3.5, and their relative importance for the participant.

Satisfaction thus is a quality dimension of collaboration, both as a judgment and as an emotion, and can be assessed for both the result and the process. Whether the satisfaction judgment matches the satisfaction emotion depends on the goal congruence between individual goals and the group goal.

3.3 A theory on collaboration

As we have shown in the different models explaining the quality dimensions, there is a relation between individual expectations, the resulting commitment, and the achievement of the group goal. As an individual participant has only a partial influence and effect on the group results, the relation between individual expense of resources and the achievement of the group goal is a key to successful collaboration. The question raises, what causes people to spend their resources in support of the group goal and how does this relate to the group process that leads to goal achievement. In figure 3.5 we offer a causal model that is aimed to explain these relations.

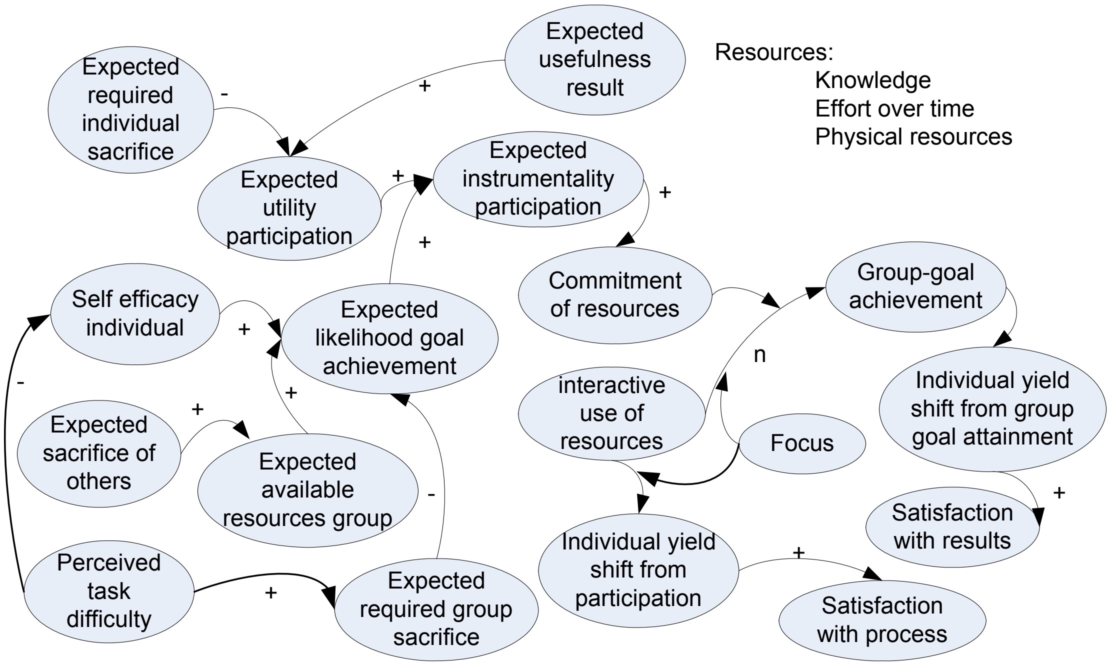


Figure 3.5. Theory on collaboration

In this model the ultimate result of a collaboration process is the achievement of the group goal. The goal can be achieved if knowledge, effort, time, attention and physical resources are used interactively by the group. The relation between resource expense and group goal achievement is not a linear relation; too little or too many resources spend will lead to inefficiency and ineffectiveness and thus to a lower degree of goal achievement. A first condition for this relation between interactive use of resources and goal achievement is what we will call focus; the use of resources efficacious to goal achievement (Briggs, 1994). In order to create and improve focus we need to be able to structure the use of resources with predictable effects.

Another condition for (focused) interactive use of resources to result in goal achievement is that those resources are committed to the group process by the individual participants and stakeholders. Locke and Latham did a meta analysis on empirical studies in which commitment and performance (caused by the use of resources) has been studied. When significant reduction in commitment was found, performance dropped accordingly (Locke and Latham, 1990). The relation between commitment and behavioral intention has been confirmed by Malhotra and Galletta (Malhotra and Galletta, 2005). We can explain the commitment of resources to a group goal using the instrumentality theory of Briggs et al (Briggs et al., 2006a, Briggs et al., 2005). For participants to commit, and indeed spend resources to a goal they should expect some use of that goal, it should be instrumental to them; meaning that it should, with a certain likelihood offer them some individual utility (Briggs et al., 2006a). Expected utility of participation is caused by expectations of the usefulness of the result for the individual and the expected resource expense (time, effort, etc.) of participation. Mayer and Allen (Meyer and Allen, 1991) found that there are thee components of commitment; (1) affective or emotional commitment, (2) continuance commitment, a cost assessment and (3) normative commitment, a felt obligation to continue. Emotional commitment in collaboration can occur when the group has some bond, or when there are emotions attached to achieving the group goal. Continuance commitment resembles most to the utility assessment also found in expected utility theory (Schoemaker, 1982) which can be described as a cost benefit analysis of the utility of perusing the group goal for the individual. Last, continuance commitment resembles the concept of group think described by Janis (Janis, 1972), a group pressure to keep perusing the goal. Also the idea of sunk costs comes in mind here, the economical concept that explains that once some resources are spend, while there is not yet a result, the pressure to continue is larger. Besides the usefulness assessment, Briggs et al describe the assessment of likelihood. The likelihood assessment also found in expectancy theory (Schoemaker, 1982) is more complex. To assess the likelihood of goal achievement people will consider their own ability to achieve the goal and the expected resource expense from other group-members; since the group goal cannot be achieved by one individual alone, the likeliness of goal achievement is an assessment of both the individual and the group. Self efficacy (the perceived self-ability) is affected by the task complexity as perceived by the individual (Locke and Latham, 1990). Task complexity also can have a motivating effect; when something is perceived to be difficult, and thus likely to require a large resource expense, people are likely to spend more resources on it (Locke and Latham, 1990); like in continuance commitment, a half effort is a wasted effort. The expectation about the resource expense of others depends on the expected available resources of group members, and the expectation about whether that will be sufficient compared to the required resources. Bishop et al showed that perceived support from a team lead to increased commitment to this team (Bishop et al., 2005). In such case the individual determines whether he expects the other group members to have the skills, knowledge and time required, and whether they are willing to spend those resources on the group goal, an individual judgment of a group construct (Jung and Sosik, 2003).

Satisfaction as we discussed above can be determined for the process and the results. Satisfaction as described above is caused by an perceived shift in yield with respect to personal goals (Briggs et al., 2004). In case of goal congruence, when the group goal is instrumental to the individual participants, and this goal is achieved, then, it is likely that there is a positive yield shift with respect to individual goals which would give a satisfaction response according to the yield shift theory from Briggs (Briggs et al., 2004). Such yield shift is likely because in a collaborative effort the likelihood of goal achievement is likely to be smaller than 1; as goal achievement depends on the commitment of all group members. Satisfaction with the process depends on the instrumentality of participation; this does depend on the way the resources committed are used in the process, but causes of this instrumentality can vary from “I got to catch up with colleagues”, to “I learned a new way of working” and a myriad of other causes. When the resources are focused on goal achievement, through external support (process support or technology), and the condition of goal congruence is met, this focus will increase the likelihood assessment of individual goal achievement and therefore cause a satisfaction response, unless expectations about the effect of collaboration support were hyped.

The model in figure 3.5 describes how different individual factors contribute to the success of collaboration as a group effort. Quality of collaboration, defined in the previous section as effectiveness, efficiency, productivity, commitment and satisfaction can be achieved based on the following relations:

- Effectiveness; quality of the real result compared to the goal

Effectiveness for the individual is determined by comparing the expected result and its usefulness, with the result and usefulness of goal achievement, as perceived by the individual.

Effectiveness on a group level can be assessed when comparing the group goal, as presented or established at the start of the collaboration process (not in this model), with the goal achieved as a result of the group effort.

- Efficiency; the difference in real resource expense compared to the intended resource expense

Efficiency of the collaboration process can be assessed for the individual by comparing personal expected resource expense with the resources actually committed to the process, or by comparing the expected required resource expense of the entire group, with the actual commitment made by the entire group.

Efficiency on a group level can be determined comparing the resources that were planned to be available or allocated (not in this model), to the resources that were actually used (interactive use of resources).

- Productivity; the balance between results and the resource expense

Productivity of the collaboration process can be assessed by the individual by comparing the resource expense with the results obtained. Expected productivity is found by comparing expected resource expense with expected results.

Productivity on a group level can be assessed based on a norm, or by comparing the intended resource expense with the intended results. Productivity is partly described in the model with the expected resource expense and the expected utility of the result and by comparing resources committed with the actual result or goal achievement.

- Commitment of resources; a force that binds the individual to spend resources (time, effort, knowledge and physical resources) to achieve the group goal

Commitment in the model is the sum of the participant's real individual expense of resources, caused by perceived instrumentality of the group goal to the individuals involved. While some resources are committed fully at once, like physical resources that are allocated for the collaboration process, others are committed to a certain extent, like effort; people can make more or less effort in the man-hours they allocate to the collaboration process. From a participant perspective, willingness to commit resources can be assessed.

- Satisfaction; affective positive arousal towards the process and the results

Satisfaction responses will manifest with respect to any collaboration or outcome-of-collaboration that give rise to shifts in yield perceptions for the salient goals of their participants.

While this model gives a first understanding of how individual commitment supports achieving the group goal, it does not yet explain how collaboration support can affect this mechanism to help the group achieving the goal. In the next section we will explore the trade-offs in collaboration process design choices and derive from this exploration a theory on the quality of collaboration process design for Collaboration Engineering.

3.4 A theory of collaboration process design

From our theory on collaboration we can conclude that in order to achieve the group goal the stakes, resources, and task need to be aligned and trade-offs between effectiveness, efficiency, productivity and satisfaction need to be made while gaining sufficient commitment to avail the resources required to achieve the goal. To further understand the tradeoffs and relations between the quality dimensions in a collaboration process design we did depth interviews in which we asked facilitators to explain the design choices they made while designing a process based on a case description. (See textbox 4.5 in chapter 4 for method) We particularly asked facilitators about the trade-offs they made while choosing among facilitation techniques. The key tradeoffs we derived from this study are displayed in figure 3.7. They offer a theory on the quality of collaboration process design for Collaboration Engineering.

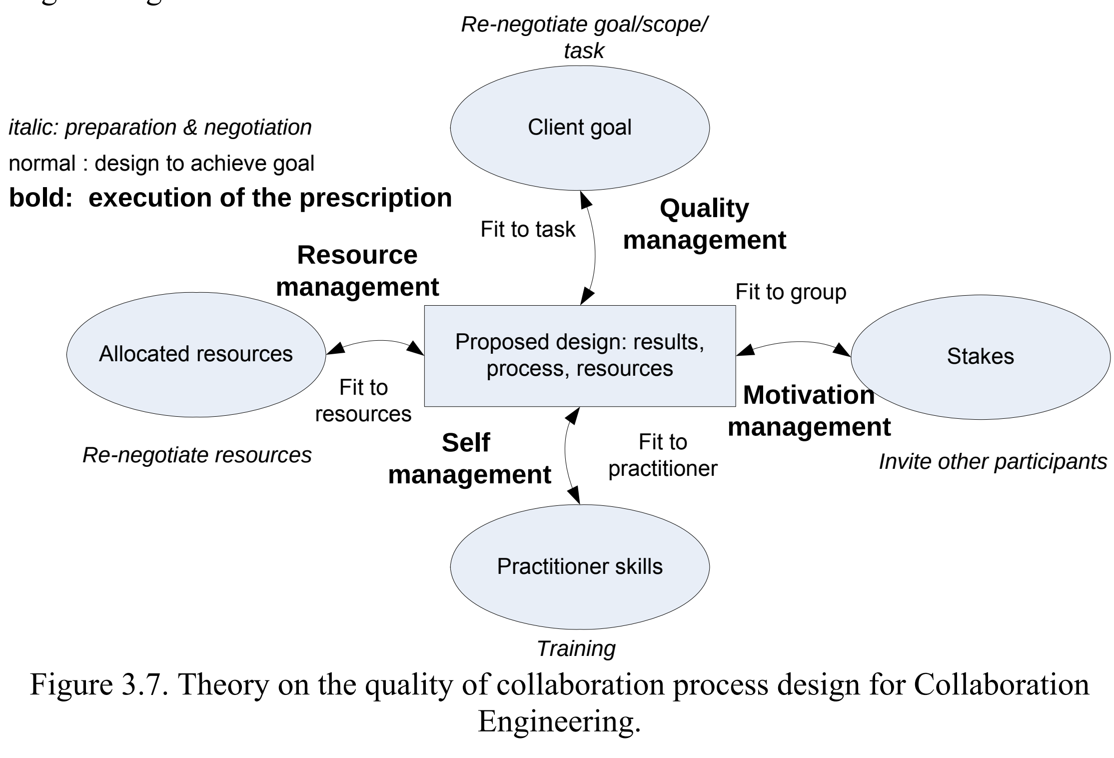


Figure 3.7. Theory on the quality of collaboration process design for Collaboration Engineering.

A collaboration process design describes the intended results, the process and the use of resources in this process. Resources in a collaboration process consist of effort over time, attention, knowledge and physical resources. There are four dimensions that describe the tradeoffs that emerge in the design of a collaboration process. First the goal that is established for the process, second, the individual stakes of the participants in the process, third, the resources allocated to the process; last the skills of the practitioner or facilitator.

When a collaboration engineer meets with the client the first time, most of the requirements and constraints for the process are still negotiable. In this phase the collaboration engineer will establish these requirements and constraints, starting with the goal and the allocated time. The collaboration engineer can ask for a larger timeframe for the meeting or reduce expectations with respect to goal achievement and quality of results to ensure the time allotted is sufficient to achieve the goal. Next, the physical resources are discussed and the stakeholders involved are analyzed or determined. Conflicting stakes or and the stakes in the outcome affect the amount of effort participants will make, and the knowledge they are prepared to contribute. If many stakes should be accommodated, the process should be designed in a way that enables this. Alternatively, the collaboration engineer can suggest involving more or less stakeholders to manipulate the need for stakeholder accommodation and the management of effort and knowledge sharing. A last possibility to alter the constraints to the design is to develop new thinkLets so there are more options in the design phase, and to adjust the intended amount of training for the practitioner.

Once these requirements and constraints are fully established, the process can be designed. In the design phase, the collaboration engineer has to identify techniques that enable the fit to the task, group, resources and the practitioner skills. In most cases the task fit is the most important fit, then the collaboration engineer checks if the group will accept the technique, if it will fit the timeframe and whether the practitioner will be able to use the technique successfully. Each of these 'fits' is established based on experience and predictions. If the collaboration engineer used a technique before, it will be easier to estimate: its results, whether the group will like it, the time and resources required, the ability of the group to perform it, and the ability of the practitioner to perform it. Once a process design is established it can be verified with stakeholders. Sometimes this requires re-negotiation of the resource budget.

During the execution of the resulting process prescription again these four dimensions play an important role. The practitioner needs to manage the resources available, the quality of the results the motivation and effort of the stakeholders and his own efforts. A key difficulty occurs if the requirements and constraints that were established in the negotiation and preparation phase turn out to be different or change during the process. Examples of this are people arriving late resulting in a shorter timeframe, no-shows or people walking away resulting in an alternation of the stakes represented, a different perception of the task and goal by the group or overestimation of practitioner skills (Vreede et al., 2003a).

If these are the four key tradeoffs in design then we can now establish the four quality dimensions related to this:

- **efficaciousness** (design fit to the goal)
- **acceptance** (design fit to the stakes)

- **transferability** (design fit to the ability of the practitioner and offers support for execution)
- **reusability** (design fit with available resources in each instance of the process)

To compensate for the lack of design skills from the practitioner, the difference between requirements and constraints at design time and during execution should be minimal. Therefore a last quality dimension of the design is its **predictability**.

We will more narrowly define the five quality dimensions of a collaboration process design below:

- *Efficaciousness = the extent to which the design, when used as prescribed will focus the expense of resources to achieve the group goal*
- *Acceptance = the extent to which the design when used as prescribed accommodates individual stakes sufficiently, to motivate stakeholders to commit the required resources for goal achievement.*
- *Reusability = the extent to which the design can be used successfully in different instances of the task*
- *Predictability = the extent to which the design, when used as prescribed, creates a process and results as intended by the collaboration engineer.*

- *Transferability = the 'ease of training' and the 'ease of execution' from the perspective of the practitioner*
- *The ease of training is determined by the training load:*
  Training load is the amount of cognitive effort required from the practitioner to sufficiently understand the process prescription.
- *The ease of execution is determined by the execution load:*
  Execution load is the amount of cognitive effort required from the practitioner to execute the process prescription.

In the next section we will determine what kind of support we can develop to enable the development of high quality collaboration process designs.

3.5 Design support for Collaboration Engineering

Now that we established the five quality dimensions of a collaboration process design, we will explore possibilities to support this design effort. In chapter 1.3 and in chapter 2.3.3 we already indicated that we will use thinkLets as the building blocks for the collaboration process design. However, further conceptualization of the thinkLets is required to enable the creation of a high quality collaboration process design according to the five quality dimensions. Additionally, a design and transfer approach and supporting artifacts are required to support the use of thinkLets in a Collaboration Engineering project.

The first two dimensions; efficaciousness and acceptance can only be assessed with respect to the specific situation. However to support the creation of an efficacious and acceptable design we can offer design support guidelines. For a set of thinkLets we can create classifications and guidelines on the possible combinations of thinkLets and their effects. These effects can be compared to the group goal to choose among thinkLets. Furthermore, we can offer guidelines on how to design a collaboration process. However, both for the entire design and for the thinkLets we cannot determine or support efficaciousness and acceptance independent from the context; we cannot assess the efficaciousness and acceptance of a process design in general, neither can we indicate a normative efficaciousness or acceptance of thinkLets without specifying specific requirements to the context.

For the transferability, reusability and predictability this is different. For instance, a thinkLet that can be executed on different platforms is by definition more reusable than a thinkLet that prescribes the use of a single platform; the chance that this platform is available in different instances of the task is smaller. To increase predictability of techniques we can document their effects based on experience and we can try to understand the causes of these effects to reduce the uncertainty of the effect. Last, to increase transferability we can decrease the cognitive load of understanding and using the process prescription, which will lower the need for practitioner skills and thus increase the likelihood of proper execution. In the next chapters we will further explore theoretical foundations to derive support for the design and transfer of high quality collaboration process designs for Collaboration Engineering.

Combining the insights in the quality of design we need to develop the following products to support the creation of high quality collaboration process designs for Collaboration Engineering:

- A transfer approach to teach the practitioner to use the collaboration process prescription and a template for a collaboration process prescription that is transferable to practitioners.
- A design approach with a set of guidelines and classifications of effects of collaboration support to enable the choice and combination of thinkLets into an efficacious and acceptable collaboration process design.
- The conceptualization of the thinkLet to increase its transferability, reusability and predictability based on cognitive load theory, the design pattern concept and rule-based interventions.

Since the thinkLet concept should support both the design and the transfer of the collaboration process we will first explore the requirements to the transfer approach and the process prescription template (chapter 5). Next, we will develop the design approach with the design guidelines and classifications to create such collaboration process design (chapter 6). Both will reveal requirements to the thinkLet concept. This will be further developed based on these requirements, the design pattern concept and rule conceptualization. (chapter 7). First, we will address the research approach in chapter 4.

Chapter 4. Research approach

In this chapter we will describe the research approach. We will first describe the research and application domain. Second, we will discuss the research philosophy. Third, we describe the strategy for the research. In the fourth section we will detail this strategy with the research instruments we used and the steps we took to derive the theory and to build and test the different design and transfer support concepts. Last, we will describe the outline of the research to give each step a place in the structure of this dissertation.

4.1 The research and application domain

The object of engineering in Collaboration Engineering is a collaboration process and collaboration support; rules and capabilities to support groups in establishing this process (Briggs et al., 2003a, Vreede and Briggs, 2005).

Collaboration Engineering has roots in several research domains. As described in the introduction in chapter 1, Collaboration Engineering was developed from collaboration support approaches such as facilitation (Bostrom et al., 1993, Griffith et al., 1998, Schwarz, 1994), Group (Decision) Support Systems (Fjermestad and Hiltz, 2001, DeSanctis and Gallupe, 1987, Nunamaker et al., 1997), and Computer Supported Cooperative Work (Grudin, 1994, Ellis et al., 1991). We will use the literature on collaboration support as our main resource to develop the design and transfer support for collaboration engineers.

The outcomes of interest in collaboration have been analyzed in the context of collaboration support but to gain some theoretical understanding of the phenomena found we used research on commitment (Meyer and Herscovitch, 2001), satisfaction (Briggs et al., 2004), and goal achievement (Locke and Latham, 1990, Veld, 1987).

Collaboration Engineering is an engineering approach and thus has links with systems engineering (Checkland, 1981, Jackson, 1983), software engineering (Boehm, 1988, Gamma et al., 1995), and Business Process Engineering (Davenport, 1993, Grover, 1999, Kettinger and Teng, 1997). These approaches can be used as a blueprint for the Collaboration Engineering approach and the design approach within this approach. For the transfer of the knowledge and skills required to execute the collaboration process we can use literature on knowledge transfer as addressed in cognitive theory (Sweller, 1988, Bjork- Ligon and Bjork, 1996) and knowledge management (Abell and Oxbrow, 2001, Nonaka and Takeuchi, 1995).

Last, an important source for the design support we will develop is the thinkLet concept (Briggs et al., 2003a, Briggs et al., 2001, Kolfschoten et al., 2006a, Vreede et al., 2006a) for this purpose we used design patterns (Alexander, 1979, Gamma et al., 1995), communication theory (Krone et al., 1987) and representation of natural language to instruct virtual humans (Badler et al., 1998, Badler et al., 1999).

The research fields related to Collaboration Engineering are displayed in figure 4.1. Collaboration support is the origin of the Collaboration Engineering approach, which is further developed in this study as an engineering approach with focus on the design and transfer of collaboration processes. For the thinkLet concept, support is developed based on several bodies of literature and finally the outcomes of interest in Collaboration Engineering are studied based on different theories in research.

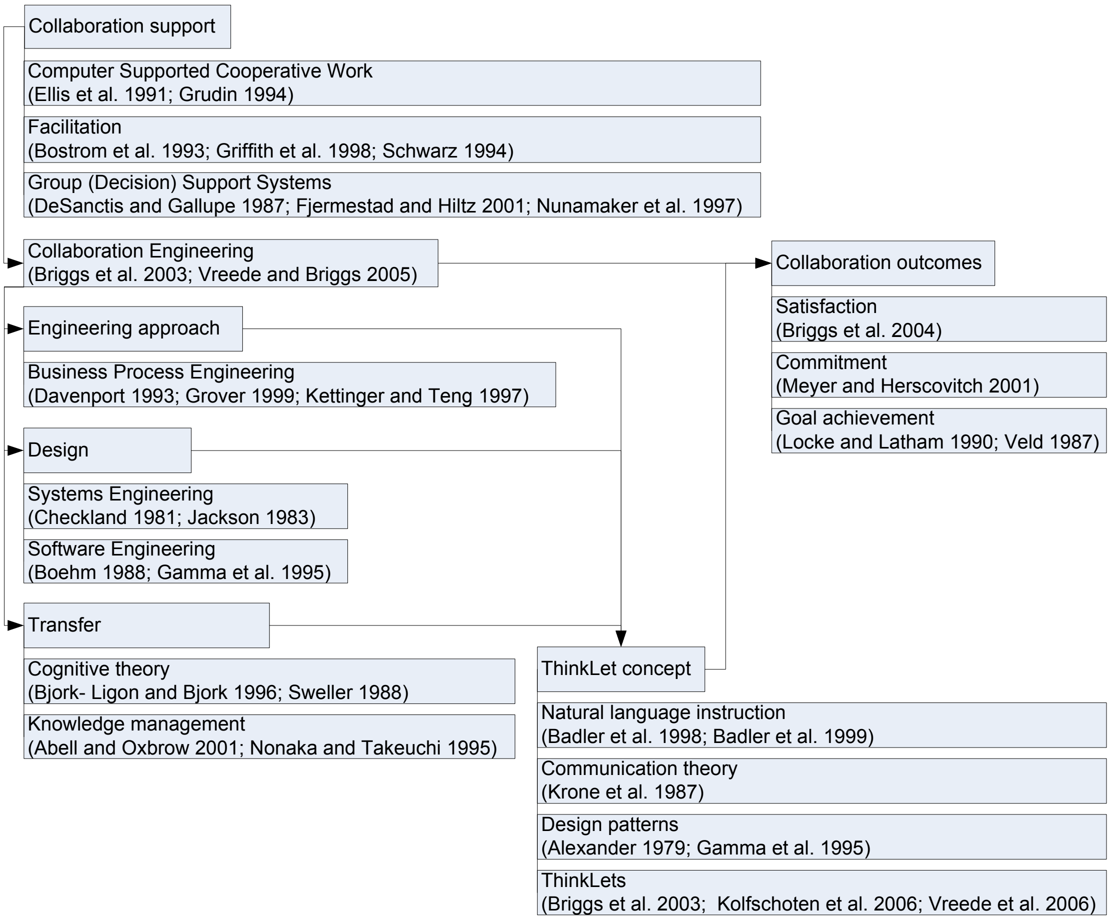


Figure 4.1. The research domains related to Collaboration Engineering.

Collaboration processes occur in a myriad of application domains. As described in the scope of Collaboration Engineering in chapter 2, this approach focuses on high value recurring tasks in organizations, that are knowledge intensive and for which a collaborative approach is required (joint effort of a group). With these requirements there is no apparent application domain demarcated. Logically, as collaboration is not an objective on its own, but rather it is instrumental to a goal. Supporting a collaboration process therefore might require domain knowledge of the field of application. However, practitioners are supposedly domain experts. Collaboration engineers on the other hand, will have to gain a basic understanding of an application domain in order to design the collaboration process.

4.2 The research philosophy

The research philosophy describes the worldview and the perception of the criteria that determine validity of the research. There are two main stream research philosophies; interpretivism and logical positivism. The key difference between these world views is that logical positivism provides a set of mental disciplines for addressing questions of cause and effect, while interpretivism provides a set of mental disciplines for addressing questions of the meanings that people ascribe to the words and actions of others.

Logical positivism assumes that there is an objective reality to the patterns of cause it investigates, that they exist independently of the human mind. The goal of logical positivist enquiry is to develop theoretical models to explain variation in the phenomenon of interest, to test these models by deriving hypotheses from them and conducting experiments, and if a theory survives such attempts of falsification, to apply the models in ways that increase the likelihood that people will survive and thrive.

Because the meanings that people ascribe to their experiences vary from individual to individual, interpretivism assumes that there is no objective reality with respect to meanings. Interpretivists believe that meanings can only be understood through social constructions by researchers (Klein and Myers, 1999). Research in the interpretivism tradition aims to derive knowledge from open interpretation of rich data collected as perceptions of the researcher. We will further explore these philosophies to derive a conclusion on the philosophy that will be used in this research.

Interpretivism

In interpretivism, understanding comes with experience of a researcher. The reality of meaning and observation cannot be separated; the researcher cannot be objective, because his background will color his experience and therefore his interpretations of new experiences. The researcher becomes himself the research instrument, he interprets his experiences. Typical research methods for interpretivism research are case studies and pattern analysis, but other instruments as survey's can also be used (Weber, 2004). To ensure quality and validity of interpretivism research, a number of guidelines can be found in literature (Trauth and Jessup, 2000, Klein and Myers, 1999):

- Hermeneutic Circle: interpretation is done through iteration between deriving meaning of parts, the whole that they form and their context
- Triangulation: using multiple methods and perspectives (researchers/approaches) to gain confidence in the results
- Contextualization: taking into account not only the phenomenon of interest but also the history and background of the research setting and elements involved
- Interaction: as the data obtained are an interpretation and perception the researcher should not stay as an outside observer but should have a close interaction with the subjects
- Abstraction: deriving generic understanding or theory that is observable in multiple cases
- Openness: being open for the story that the data tell; being sensitive to the possibility that data can be conflicting and that initial presumptions are wrong.
- Suspicion: Data and interpretation should be logic and truthful, without bias or distortion, which can be shown trough illustration with examples from the data obtained.

Logical Positivism

The logical positivist philosophy assumes that the patterns of cause-and-effect in the physical world exist independently of the researcher, and that they may be inferred or deduced by observing and describing relations among events. Such inferences are then formally documented as causal theories, whose causal propositions can be refuted by experience. Such theories are said to be falsifiable; it should be possible to prove by experience that the cause-and-effect proposition is false. Popper (Popper, 1959) argues that by testing whether logic or causal relations exist, we advance our knowledge. Positivists often use experiments to alter the causal factor and observe the effect, Next, they will use statistical analyses to determine if the cause-and effect pattern manifested as a significant effect compared to random chance. logical positivism does not, however, assume that the researcher is an objective observer. For this reason, replication of studies by multiple researchers is the criterion for reliability of results (Weber, 2004). Quality criteria for a logical positivist theory are (Bacharach, 1989):

- Internal consistency: the logic of the theory contains no tautology
- Falsifiablity: it would be possible to refute the logic of the propositions of the theory by experience.
- Predictability: the logic of the theory gives rise to predictions that are consistent with observed outcomes.
- Parsimony: The theory explains the variation in the phenomenon of interest with the fewest possible constructs, assumptions, and propositions.

Thus both interpretivists and positivists state that being an objective researcher is impossible and thus documenting objective truth is also impossible, but in order to test the theory the aim is to be as objective as possible (Weber, 2004). Naturally, the aim of the research is to show that the results of research are not accidental findings, but that they are reproducible, and can be assumed valid or corroborated in a more general perspective. In this research each of these worldviews is important; we want to understand the logic that explains the effect of the design on the quality of collaboration and we want to create predictable artifacts to create further understanding of the challenges in the creation of a collaboration process design.

Following Webber, Lin, and Trauth and Jessup (Weber, 2004, Trauth and Jessup, 2000, Lin, 1998) the research in this dissertation should not be labeled as logical positivism or interpretivism. Rather we feel these perspectives complement each other and can be used together to gain a richer understanding of the phenomenon of interest. Despite their different worldviews, the criteria to indicate proof of a concept, complement rather than exclude each other.

4.3 The research strategy

To gain understanding of our phenomena of interest, quality of design and quality of collaboration, and their relation we need to adopt an inquiry system; a strategy to derive understanding and to create knowledge. Churchman (Churchman, 1971) describes 4 historic inquiry systems that are combined in the Singerian inquiry system. The Singerian inquiry system uses a community with multiple perspectives from a variety of stakeholders, to agree on what can be defined improvement and to establish that such improvement is made. In order to measure this improvement, we need a measurement system that reflects multiple perspectives. In this research, the measurable improvement is defined in our description of high quality collaboration, based on an analysis of different roles involved in Collaboration Engineering, and their requirements to a high quality collaboration process design. With this measurement system, we can identify (measure) ‘conflict’ between the required qualities of a collaboration process design and quality as determined in various readings from various perspectives. To analyze the conflict the Singerian inquiry system prescribes the use of multiple perspectives as we intend by using different measurement instruments to measure results from different perspectives. Replicability is the validation method that the Singerian inquiry system prescribes, consistent with both research philosophies. We will use this Singerian inquiry system as a starting point for our research strategy.

Two research strategies are dominant in the research areas described above; inductive and deductive research strategy. Several authors suggest that inductive and deductive research strategies can be combined (Trochim, 2000, Creswell, 1994). Creswell explains that different combinations of these research approaches are possible. While qualitative methods are often used for inductive research and quantitative methods are often used for deductive research, both approaches can be combined phased, mixed or one dominant approach could be supplemented with small aspects of the other approach. In this research the approaches will be used phased on a higher abstraction level; first an inductive, mainly qualitative cycle has been used to build a theory and to create the design support, then a deductive, mainly quantitative cycle will be used to evaluate this design support. However, on a lower abstraction level the measurements will be supplementary; in the inductive phase, quantitative data are used to enrich the qualitative data and vice versa in the deductive phase, qualitative data are used to verify the quantitative evaluation data. Also, the different perspectives can be retrieved on a high level; the quality of collaboration process design will be assessed from four different perspectives; the practitioner, the collaboration engineer, the group goal, and the individual stakeholders in the process. Note in this respect that the ‘group goal’ is a difficult perspective, in some cases the group goal is proposed by the group leader in some cases it is determined by the group and there can be different levels of shared understanding about the group goal and different levels of goal congruence; the extent to which the group goal is conflicting with one or more individual goals of the group members and other stakeholders. However, a group process cannot be designed without a group goal, and so we will distinguish the group goal perspective and the individual stakeholder perspective as separate perspectives that might to a smaller or larger extent overlap.

Besides an inductive exploratory and a deductive evaluative phase, this research project requires the development of design support; models, guidelines and an approach to support the collaboration engineer in the effort of collaboration process design. Creating design support is a design effort by itself. Our design support will exist of both approaches (the design and transfer approach) and objects (design support models, the process prescription template and the pattern language). The design approach used to create design support will be an iterative design approach; both the processes and the objects will be created, tested and revised in several rounds to identify their weaknesses and to create improvements.

Since the different steps in this research have different purposes and require different quality and validity, different research instruments will be used. According to Strauss and Corbin (Strauss and Corbin 1998) who discuss the choice between quantitative and qualitative research, the research method is a tool to build theory. Their combination is often more valuable than using only one approach. This will provide the basis of our research strategy. We will use multiple methods and several perspectives to derive, operationalize and test our theory. We will further explain the implementation of this strategy for each step in the research project. The research strategy will involve 3 key phases. (See figure 4.2):

Inductive exploratory phase
The first phase is an inductive phase where exploratory interviews and literature reviews are done. These are used to derive patterns in the data that constitute the basis for the theory and the design support. In this phase the criteria for interpretivism research are taken into account such as triangulation, openness, interaction, etc. With these patterns tentative propositions are derived to develop a theory on the quality of design and its effect on the quality of collaboration, which will offer a theoretical foundation for the Collaboration Engineering approach.

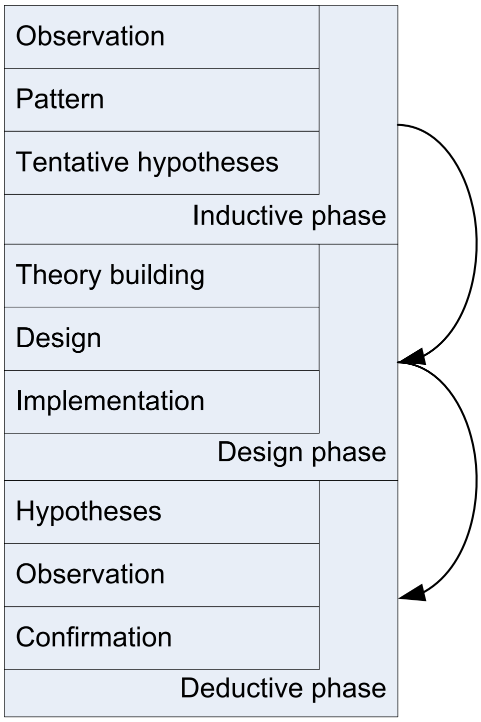


Figure 4.2. Research strategy based on (Trochim, 2000, Creswell, 1994).

Iterative design phase
Second, the theoretical basis of the collaboration support approach will be operationalized based on additional data and literature analysis and based on a large survey among facilitators. Based on this, support concepts are developed. To further refine and detail these support constructs, they are tried and evaluated by the users, researchers and other stakeholders involved. The evaluation results are used to identify ways to improve the different concepts. These steps are repeated until users

and stakeholders do not find improvements anymore, which can be implemented within the limits of the timeframe of this research. Based on this phase we will be able to reflect on the ‘engineerability of collaboration’.

Deductive evaluation phase
Last, the entire approach is evaluated in a final field experiment. Each of these phases is performed with multiple research instruments, which will be described in more detail below. The hypotheses that we will evaluate in this study is as follows:

A practitioner who executes a collaboration process prescription designed and transferred according to the Collaboration Engineering approach is not outperformed by a professional facilitator on:

a. satisfaction with the process
b. satisfaction with the results
c. commitment of resources to the group goal
d. efficiency of the process
e. effectiveness of the process
f. productivity of the process

This hypothesis will enable us to reflect on the transferability challenge in Collaboration Engineering.

For this purpose we will first detail the approaches for the design and transfer of collaboration processes and we will create design and transfer support to enable the collaboration engineer to design and transfer the process. This support will exist of a pattern language, classifications to support the choice of patterns and a collaboration process prescription template.

4.4 The research instruments

In this section we will discuss the research instruments, per phase in the research.

Phase 1 Exploratory
The first step in this research is to identify patterns that explain the relation between the quality of design and the quality of collaboration in the context of Collaboration Engineering. To this end we first analyzed the Collaboration Engineering approach. We defined the key concepts in Collaboration Engineering based on a literature review and we analyzed the roles involved in the Collaboration Engineering approach, and the organizational context of collaboration support, using in depth interviews. Furthermore, we analyzed the task of the facilitator from different sources in literature and split this task in tasks for the Collaboration Engineer and tasks for the practitioner. This combination of steps helped us to explore the approach from multiple perspectives; the theoretical perspective in literature, the practitioner perspective, and the facilitator perspective. We also used the hermeneutic circle guideline, in our interviews; we asked the interviewees to reflect not only on their own role and task but also about the organizational context and the success of collaboration support at large. Second, we performed a literature review to derive the factors to build our theory about the quality of design. To integrate the factors that were derived in this process we used a clustering method for abstraction. Last, we defined the resulting factors based on literature and the implications of the

Collaboration Engineering approach. In table 4.1 each step and the instruments used are displayed. For the interviews we will further elaborate on the method used.

Step	Instrument
Defining the key concepts in Collaboration Engineering	Literature review
Practitioner interviews	Interviews
Patterns in role separation organizational context of collaboration support	Interviews
Analysis of the facilitator's task and split in practitioner and collaboration engineer task	Literature review
Meta analysis outcomes of interest in collaboration	Literature review
Corroboration Quality dimensions	Interviews

Table 4.1. Inductive phase, theory building.

Patterns in role separation organizational context of collaboration support
Instrument: interviews
n: 18
Date: March 2006-September 2006

In this interview we compared the roles distinguished for group support in the organization with the way collaboration support is implemented in the organization and its successfulness. We interviewed 18 people that worked a significant part of their time to support groups. We found these people mostly through the search of users of GSS in our own networks, and furthermore we asked some of the respondents to suggest other respondents. For the interviews we used an interview protocol which was based on the role separation model described in chapter 2. The interview was tested with a colleague and revised based on insights from this try-out. The interview protocol developed in collaboration with Fred Niederman from Saint Louis University (Kolfschoten et al., 2006b) and was approved by the Saint Louis University Institutional Review Board and can be found in appendix 3.

We asked the respondents about their organization, the role of group support, the type of group support they offered and the task separation they used when supporting groups. Furthermore, we asked them about the skills, personality and knowledge required for group support, their own training and the successfulness of their efforts in supporting group, and the role of the organizational setting and the technology used in this success. The result had both numerical and textual results. For several questions we classified and counted the responses to generalize the results. For other questions we could calculate and average group result. Especially for the role separation we asked questions from different perspectives to corroborate answers. E.g. we asked "Who operates the technology?", and "Do you perform the chauffeur role?" We used the results in several aspects of the research, but mainly in section 2.2.

Textbox 4.1 Roles interviews

Practitioner interviews
Instrument: interviews
n: 4
Date: April 2004-June 2004

In this small study we interviewed 4 practitioners who were trained according to the Collaboration Engineering approach to support groups in a risk self assessment (Vreede and Briggs, 2005), to reflect on their experiences in supporting groups based on a collaboration process prescription transferred to them by a Collaboration Engineer. The interviews were semi structured. The interviews helped the researcher to get a first impression of the Collaboration Engineering approach in action, and it revealed some first challenges of the transfer and deployment of a collaboration process. Findings from these interviews are used in several situations in an illustrative way.

Textbox 4.2 Practitioner interviews

Corroboration quality dimensions
Instrument: interviews
n:15
Date: March 2005-May 2006

As part of the interviews on choice criteria (see textbox 4.5) we asked facilitators the question: "You support groups in their collaboration process, what do your think is high quality collaboration?"
We transcribed the answers from tape. We identified the different quality aspect mentioned in the answer and isolated them. We clustered the factors that were similar and we summarized those to derive an insightful set of quality dimensions. The results of this analysis are used in section 3.2.

Textbox 4.3 Quality of collaboration interviews

After using the first studies to get an impression of collaboration support and Collaboration Engineering in organizations, we focused on quality of collaboration. Based on the literature analysis and on the corroboration interviews (triangulation guideline), we derived six dimensions that describe quality of collaboration (efficiency, effectiveness, productivity, satisfaction with process and outcome and commitment). Furthermore, we used the research described below on choice criteria (textbox 4.5) to distill five quality factors of the design. These factors are: transferability, efficaciousness, acceptance, predictability, and reusability. To support the creation of a high quality collaboration process design for Collaboration Engineering we created several support concepts. For each dimension we will explain the exploratory steps and the iterative design steps that led to the construction and revision of these five support concepts to support the conscious design of a high quality collaboration process; a collaboration process prescription template, a transfer approach, a design approach, complementary design support and the thinkLet concept.

Phase 2: Iterative design

Design is a complex effort, which requires creativity, a good understanding of the requirements, and their effect on outcome quality. Requirements are seldom correct and complete the first time. Furthermore, in complex settings, multiple stakeholders can pose conflicting requirements which forces the designer to make design choices (Boehm and Ross, 1989). Consequently, design cannot be seen as a linear process, it requires iteration and revision, not only of the design but also of the requirements; after each iteration, the design will be evaluated and improved (Boehm, 1988). This has been done for the collaboration process prescription template, the Collaboration Engineering transfer approach, the Collaboration Engineering design approach, the design support and the thinkLet conceptualization. These iterative evaluations have only one objective; to find aspects for improvement of the design. The improvements can be implemented again, and re-evaluated to find new improvements until there are no more significant suggestions for improvement (taking into account the limitations of this research project). We will discuss the design and evaluation of Collaboration Engineering support for each (pair of) collaboration process design quality criteria.

Transferability

To support practitioners, a process prescription has to offer complete information and it should be transferred to practitioners in an efficient training. As a basis for this analysis we used the task description of the practitioner derived in chapter 2. To support transferability we used methods from literature on cognitive load to identify requirements to a collaboration process prescription template, training approach and the thinkLet concept. In three separate case studies, each repeated twice, the collaboration process prescription template was evaluated. In a separate case the training approach was evaluated. See table 4.2.

Step	Instrument
Identification of methods from cognitive load theory applicable to the process prescription template, training approach and thinkLet concept	Literature review
Collaboration process prescription template evaluation	Case study (method reported in chapter 5)
Training approach evaluation	Case study (method reported in chapter 5)

Table 4.2. Transferability research.

Efficaciousness and acceptance

To support the design of the collaboration process, design support has been developed, which is used in combination with a design approach to support the design of collaboration processes that meet the quality criteria. Specific focus in the design phase is on efficaciousness and acceptance. The design approach is based on the Collaboration Engineering approach and design approaches in related research fields as for instance software engineering. Based on these, a survey among facilitators with different experience levels was conducted to derive challenges in the design of collaboration processes.

One of the critical aspects of collaboration process design is the choice among and combination of facilitation techniques and the ability to combine different techniques. Additional research was therefore performed involving depth interviews and expert panels to derive choice criteria. In this study we also found the dimensions of quality of design and a classification of collaboration results. Additionally, a pattern analysis on combinations of thinkLets used in collaboration processes facilitated at the Delft University of Technology was performed. Last, the design approach was evaluated in two case studies. Again, these steps and the instruments used are described in table 4.3.

Step	Instrument
Challenges in the design of collaboration processes	Survey
Derive analogy with design approaches in other disciplines	Literature review
Derivation of choice criteria, quality of design dimensions and result classification	Survey Expert panels Interviews
Pattern analysis combinations and sequences in thinkLet use	Case study
Design approach evaluation	Case study (method reported in chapter 6)

Table 4.3. Instrumentality and acceptance research.

We will further discuss each of these studies.

Challenges in the design of collaboration processes
Instrument: survey
n: 89
Date: July 2004- April 2005

To find the key challenges in the design of collaboration processes we developed an exploratory survey. The survey consisted of six parts; a general set of questions to characterize the respondents, a section on the design activities a facilitator performs, a section on the information used in the design process, a section on the facilitation techniques used, a section on the people involved and a section on the importance of design and design documentation.

Using the International Association of Facilitators' (IAF) mail group and the website (International Association of Facilitators, 2004) and the Grp-Facl electronic discussion on group facilitation (Schuman, 2006) we solicited a response from approximately 200 facilitators who filled out the questionnaire. The respondents had a broad range of styles, methods, and work experience in many different environments, few used Group Support Systems; most facilitators used other tools and methods to support the group. We tried to build the questionnaire as generic as possible to accommodate all types of group facilitation. Despite this, it might have been difficult for some respondents to answer all questions. Some respondents explained their work situation, to provide the context for their answer. Questionnaires that contained multiple incomplete parts were excluded. As a result, 89 questionnaires were eventually taken into account; however, some questions were answered by fewer facilitators (as they were not applicable to some facilitators). The results of this study are used in various sections throughout the dissertation but mostly in section 6.1. The questionnaire can be found in appendix 4

Textbox 4.4 Challenges in design

Derivation of choice criteria, quality of design dimensions and result classification
Instrument: survey, expert panel and depth interviews
n survey: 89, n expert panel: 10, n interviews: 16
Date: July 2004-September 2006

In order to gather more information on the choice criteria used to select among facilitation techniques and the how these are used in a choice process, we followed an incremental, interpretive research approach using three complementary data sources. Data were gathered in three phases.

For the first phase of data gathering, we draw on part of a survey described in textbox 4.4 in which we asked respondent which criteria they considered when choosing among facilitation techniques. 58 respondents answered the question, most respondents indicated multiple criteria. Although this question revealed an interesting first result, we often got rather generic answers and wanted to get a more detailed insight in the choice criteria used.

For the second source of data gathering (Kolfschoten and Rouwette, 2006a, b) we held a group session with experienced facilitators at the 2004 IAF Europe conference. A total of ten facilitators participated in the 3.5 hour session. Participants each had several years of experience as a (self-) employed facilitator working in Eastern Europe or the United States. The participants shared facilitation techniques and for each technique they indicated when it could be used, and when not. Criteria to (not) use the technique were transferred to a whiteboard and discussed plenary. Although the group session resulted in rich information on session preparation and enabled participants to discuss choice criteria in their own wording, the question on the use of information addressed a general preparation process and did not focus on a particular session. We decided that in order to really elicit the choice criteria we would have to interview facilitators and ask them about the assumptions and reasoning behind their choices.

In the last phase of data gathering (Kolfschoten and Rouwette, 2006a, b) we therefore presented facilitators with a concrete and specific case description. The facilitators were asked to design a collaboration process for this case. They were then asked to choose techniques and verbalize their thinking process while doing so. This approach follows the guidelines of Verbal Protocol Analysis (Ericsson and Simon, 1993). VPA 'has been used extensively as an effective method for in-depth examination of cognitive behaviors' (Schenk et al., 1998). The verbal reports generated using this method are a valuable and reliable source of information about cognitive processes (Ericsson and Simon, 1993). The case concerned the development of a new ICT strategy for a university with a group of ten participants from different departments. Four hours were available to both analyze the problem and identify clear action points for the future. The case description was visible to respondents throughout the interviews which lasted from 0.5 to 2 hours each. A total of sixteen facilitators working privately or in Dutch universities and research institutes were interviewed. Each respondent had several years of experience in facilitating sessions using electronic meeting systems, paper and pencil methods, soft operational research, or modeling tools. Most interviewees combined experience in several areas. Each interview was transcribed into a written report. The interview protocol can be found in appendix 5

In the interviews we first explained exactly what we mean with the term facilitation technique. We then addressed the purpose of the interview and gave the respondent an opportunity to read the case. Respondents reacted very differently on the description. Some felt they had way too little information to design a session, while others immediately came up with a solution. We discussed the case until the respondent came up with an approach for the facilitation process. From this discussion we could distill different approaches to decompose the task and select techniques to create the required deliverables. To fully understand their approach we addressed each step and asked which facilitation technique the respondent would use. Next, we asked them why they chose this technique. To help the interviewees answer this question, we provided them with possible generic criteria. These were: the group need, the task, the facilitator's preference, a standard procedure and their perception of good collaboration. Note that we did not seek to confirm these aspects as criteria; we simply wanted to help facilitators to think of different reasons for their choice. When explaining why this was or was not the reason for choosing the facilitation technique, the respondent's choice criteria and the tradeoffs they encountered became apparent.

The session report and interview transcripts were then analyzed using a grounded theory approach (Glaser and Strauss, 1967). A central tenet of grounded theory is the close connection between empirical data and development of concepts to describe data. The analysis follows a four step procedure: exploration, specification, reduction and integration (Wester and Peters, 2004). The exploration phase aims to characterize the content of transcripts, by identifying as many relevant concepts or keywords as possible for each section of the text. In this phase the researcher's ideas about relevant codes and ideas from previous research play the role of 'sensitizing concepts'(Blumer, 1969). For this study the concepts identified in first phases of data collection, have the role of sensitizing concepts. In the specification phase, codes are compared and codes that are central are identified. The text segments that each central code refers to are compared to reveal differences and similarities, in order to clarify the dimensions of each central code. The reduction phase aims to elaborate the central concepts further, by describing and relating concepts. Finally, in the integration phase, the relations between the concepts are defined. Observation units are described in terms of the central concepts and related to literature, to finally combine them in the choice criteria overview (Wester and Peters, 2004). We coded both the reasons for choosing a technique and the outcomes that were intended to be produced by the technique. From the coding effort we derived the tradeoff dimensions that inform choice and quality of collaboration process design, and the classification of results. To test these we clustered the coded results among the constructs in the classification and tradeoff overview. If the coded results could not be clustered we revised the classification and tradeoffs to increase their completeness.
Limitations to this final data set are the following:

- The facilitators have similar backgrounds which could lead to more similar results than with a broader sample.
- The conclusions derived are not verified with an independent data set.
- The coding was only done by two researcher working on this project and not verified by independent parties

Textbox 4.5 Choice criteria, quality of design and result classification

Pattern analysis combinations and sequences in thinkLet use
Instrument: data analysis on GSS agenda
n sessions: 96
Date: January 2000- January 2003

To date, over 70 different thinkLets have been described (Briggs and Vreede, 2001). Many were captured from the experiences of expert facilitators around the globe. To determine if these patterns indeed emerge in collaboration sessions, and whether there are also higher level patterns that describe sequences and combinations of thinkLets, a set of sessions, conducted in 2000, 2001 and 2002 and were analyzed for pattern recognition and harvesting (Kolfschoten et al., 2004a). This set was reduced by only retaining 96 sessions that met the following criteria:

- The session was not used for educational or demonstration purposes. Sessions within (executive) educational programs mostly consist of small discussion exercises or survey's to evaluate sessions. They do not adequately represent real organizational use of GSS. Furthermore, demonstration sessions only serve to illustrate the GSS' functionality but do not address a real organizational goal.
- The session consisted of at least one sequence. A sequence consists of a series of activities that modify, extend or use the same data set in the GSS. When the group created a new dataset this meant a new sequence started. A sequence can be seen as a set of succeeding activities.
- The order of the activities in a sequence was clear. We excluded sessions where no sequence of activities could be recovered. The sequence was not always clear from the agenda. Sometimes the agenda changed during the session while this was not captured as a deviation from an earlier agenda. Informal interviews with the facilitators of the sessions were conducted where possible, to gather details about the thinkLet patterns and sequences used.
- The thinkLets were clear. The thinkLets could all be identified or constructed from the available information. Any uncertainties could be resolved through additional information gathering from the session's facilitator.

The session transcripts were collected and archived. Each transcript included an electronically generated agenda. The agenda provided a list of electronic tools used by the group and the configuration of the tools at the time the team finished working with it. The transcripts also included the complete set of electronic contributions made to each tool. Sometimes the transcripts included additional written instructions to the groups to orient them about what was expected from them during each activity. With these transcripts we studied the agenda and session results that revealed the tools, configurations, and instructions that were used during the sessions. This revealed regularities that could be codified as thinkLets. Next, the results were analyzed more carefully to determine the topic and the type of contributions. For instance we checked whether the items under evaluation in a voting tool were the same as the ones brainstormed in the previous activity. If not, we tried to identify relations between the data. For instance, if the data is a summary, or part of the previous list or if it contains completely new items. With this approach we identified sequences. Sequences logically exist of two or more thinkLets and thus automatically contain one or more combinations of thinkLets. Naturally, combinations of thinkLets are only appropriated when they occur in one sequence; that is when results from the previous step are in some form used in the next step.

There are a number of limitations that have to be taken into account when interpreting the results of this study. First, some of the sessions were executed before the thinkLet concept was introduced to the facilitators of these sessions. The reconstruction was therefore not always straightforward. We experienced some difficulty in the reconstruction of some scripts, which left some room for ambiguity. We tried to resolve any issues through focused interviews with the facilitators involved. Second, facilitators develop their own routines. Facilitators that were responsible for the sessions included in the study learned most of their routines in tool use through word-of-mouth and trial and error in sessions. Since most facilitators came from one department, this may also have caused a certain style to develop over time that is idiosyncratic to the department. These routines and style may not reflect facilitation practices around the world. Yet, the department involved is specialized in designing collaboration processes for problem solving and action formulation, so the identified sequences are grounded in a broad experience base in this area.

Textbox 4.6 Pattern analysis

Predictability and Reusability

The last dimensions of quality of collaboration process design can be supported through the use of thinkLets. ThinkLets are best practices of experts and as such are predictable to a certain extent, comparable with the predictability of design patterns. A literature review on the use and conceptualization of design patterns was performed as a basis for the re-conceptualization of the thinkLets. To derive an overview of the information that should be documented as the content of each thinkLet, we used part of the survey described in textbox 4.4 in which we asked what facilitators documented about facilitation techniques they used. Next, we analyzed literature to gain further understanding about the different patterns of collaboration. Furthermore, the pattern analysis of textbox 4.6 was used See table 4.4.

Step	*Instrument*
Literature analysis on pattern conceptualization and rule based interventions	Literature review
ThinkLet documentation textbox 4.4	Survey
Pattern of collaboration content	Literature review
ThinkLet pattern recognition textbox 4.6	Case study

Table 4.4. Predictability and reusability research.

Phase 3 Deductive evaluation

The last phase involves the complete testing of the Collaboration Engineering approach. In two comparable case studies, the researcher transferred a complete collaboration process prescription designed with the design approach and support, and according to the template with use of the thinkLets to practitioners in an organization using the training approach. The practitioners then executed the process prescription. We then compared the quality of collaboration as perceived by groups supported by practitioners with the quality of collaboration as perceived by groups supported by professional facilitators. In this phase we will be able to test our hypotheses about the effect of the Collaboration Engineering approach on quality of collaboration. The measurement framework and research instruments used in these final studies are explained in Chapter 8. The complete overview of the research phases and instruments are visualized in figure 4.3.

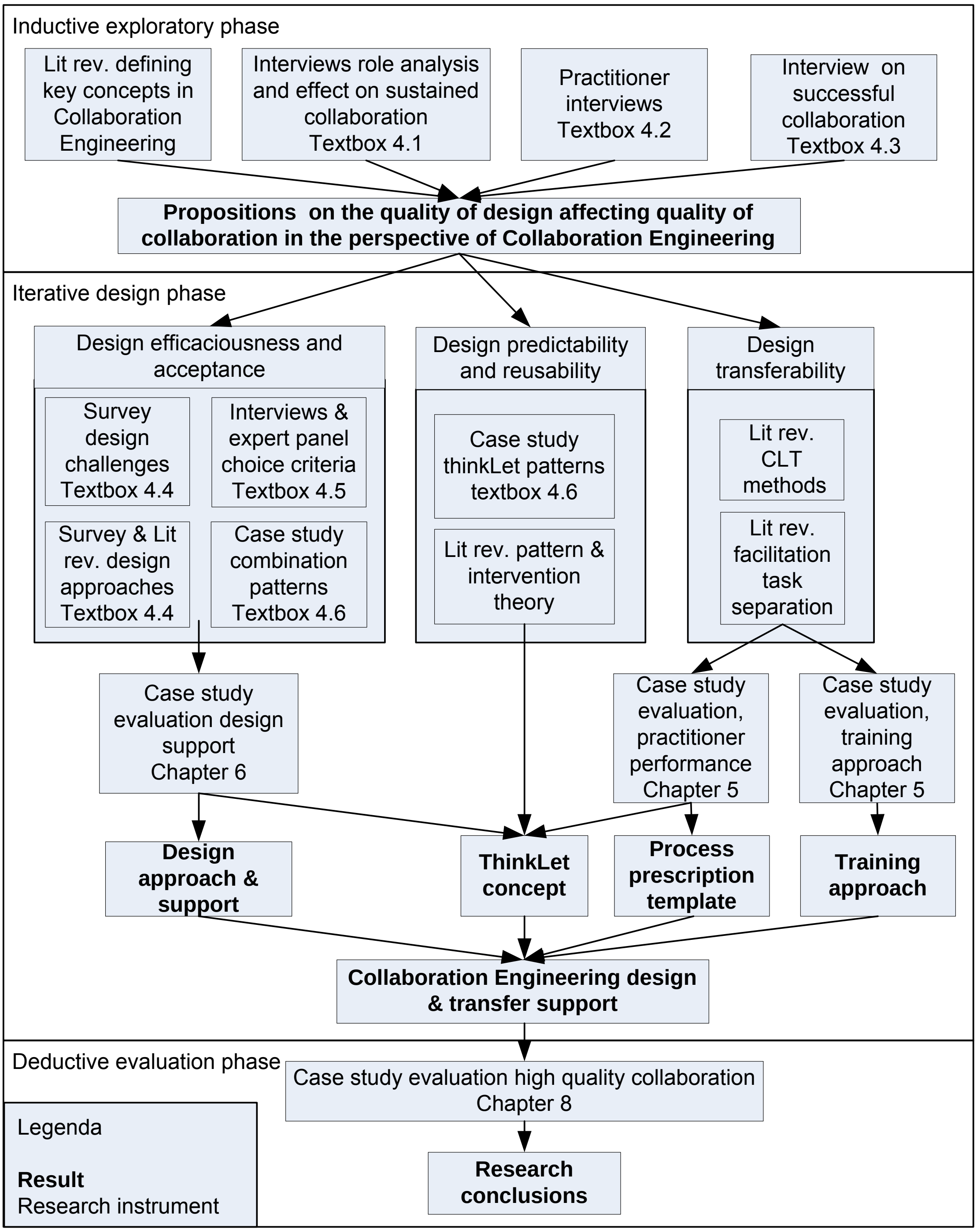


Figure 4.3. The research phases and instruments.

4.5 The research outline

In figure 4.4 the research outline is visualized. This research starts in chapter 1 with sketching the need for collaboration processes in organizations, and how groups need support in the shape of process guidance and tools to collaborate successfully. However, it is difficult to offer such support in a sustained way and one of the solutions to this is the Collaboration Engineering approach.

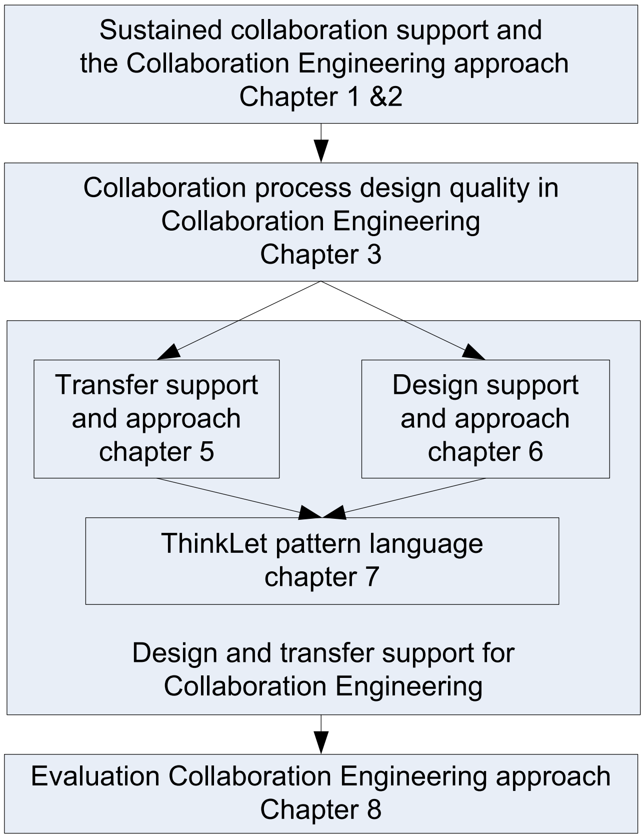


Figure 4.4. Research outline.

First we explain the Collaboration Engineering approach and the role of the design in chapter 2. In Collaboration Engineering a collaboration process and the required supporting methods are designed by an expert and transferred to a practitioner in the organization. This leads to the proposition that the design is the vehicle to operate the key success factors that lead to practitioner performance and high quality collaboration. This implies a key role for the collaboration process design. The research challenge therefore is: How can we identify, define, operationalize and test the factors that constitute high quality collaboration process designs that can be transferred as a process prescription to practitioners in an organization. The goal is to offer design and transfer support to enable successful implementation of the Collaboration Engineering approach.

In chapter 3 we derive based on a literature analysis, the factors that constitute high quality collaboration and high quality collaboration process design to build a theory about both of these phenomena. This led to the identification of five factors that describe the quality of design and lead to high quality collaboration. These are: efficaciousness, acceptance, transferability, predictability, and reusability of the design. Each will be operationalized in chapter 5-7.

In chapter 5 we explain the transfer approach, analyzing the task of the practitioner and the need for a transferable process prescription; one that constitutes low cognitive load for the practitioner and that can be transferred in an efficient and effective training. This will offer the basis for a collaboration process prescription template and a training approach that will support the transferability of the process prescription. Besides, requirements to the thinkLet concept to support its transferability are derived.

In chapter 6 we explain the approach to design a collaboration process prescription according to the template to support efficaciousness and acceptance of the collaboration process. With this approach comes a set of supporting models and classifications to select and combine thinkLets. This lead to a design approach with design support concepts. Furthermore, requirements to the thinkLet concept have been derived.

In chapter 7 we will explain the thinkLet pattern language which will be conceptualized based on requirements both from a design and a transfer perspective, and supports predictability and reusability, but also incorporates qualities that support efficaciousness, acceptance and transferability.

In chapter 8 we test the design and transfer support, developed in the previous chapters and test our hypotheses on transferability, comparing the performance of practitioners with the performance of professional facilitators, as seen from the participant perspective. For this purpose we developed several research instruments.

In chapter 9 we draw overall conclusions on transferability and engineerability, the key challenges in Collaboration Engineering, and we derive suggestions for further research.

Chapter 5. Transferability: reducing the cognitive load of the practitioner task

In this chapter we will explore the challenges of transfer to practitioners based on cognitive load theory. Next, we will use methods derived from this theory and literature on training approaches to identify ways to improve the transferability of the process prescription, and to derive a collaboration process prescription template and a training approach. The practitioner's performance is critical to achieve the intended effect of a collaboration process design on the quality of collaboration. Therefore, the Collaboration Engineering process prescription should offer the information required to support the practitioner in executing the collaboration process, and should offer this information in a way that facilitates efficient and effective learning. To support the creation of a transferable collaboration process prescription we should further understand the cognitive load of the practitioner task (execution load) and the practitioner's learning effort (training load).

5.1 Theoretical basis: cognitive load theory

To further understand transferability we will explore the techniques offered in cognitive load theory.

5.1.1 What is cognitive load?

Cognitive load can be defined as *the cognitive effort made by a person to understand and perform his task* (Sweller et al., 1998a) It has both a task-based dimension labeled mental load and a person-based dimension labeled mental effort (Sweller et al., 1998a). *Mental load is the aspect of cognitive load that originates from the interaction between task and subject. Mental effort is the aspect of cognitive load that refers to the cognitive capacity actually allocated to accommodate the demands imposed by the task* (Paas et al., 2003b). Cognitive load theory is based on the assumption that our short-term or working memory is limited to seven plus or minus two information elements (Miller, 1956). This is the information that we can process at a certain moment.

Besides working memory we have a long-term memory in which information is stored, in so called schema (Bjork- Ligon and Bjork, 1996). The long term memory is not limited in size. To learn we need to consciously combine individual elements of information to build schema. Schema in the long term memory are repositories of information that are interlinked. Consider for instance a schema of Italy. It can contain information about its geography, linked to information about Europe. It can contain information about Italian food, linked to information about (Italian) restaurants in the neighborhood. It can contain pictures of a holiday in Italy, or vocabulary of the Italian language. Each element of the schema can have links to other schema.

Schema can be handled by our working memory as an individual component. A schema is not just a storage frame; information in the schema is automatically accessed. This means that it is retrieved unconsciously. An example of this is reading; experienced readers do not process every letter they read anymore; they recognize entire words, or even parts of sentences (Sweller et al., 1998a, Buzan, 1974). Therefore, the larger the schema and the better they are automated, the more information we can process at the same time, the faster we can learn to find solutions or answers to problems.

The availability of schema determines the difference between experts and novices in several ways (Sweller, 1988): An expert, compared to a novice does not have more schema, but larger schema. A second difference is that an expert recognizes patterns of problems from previous experience, and combines these in his schema with solution-directions, while novices do not possess such schema and thus have to solve the problem from scratch. This lack of sophisticated schema causes another difference between novices and experts. Experts categorize their knowledge based on solution models, while novices do not yet see the direct relation between problems and solutions, they can only structure their schema based on similarities in the problem statement. Concluding, an expert, compared to a novice has better, larger schema that are more automated and therefore more accessible trough a better categorization and association with other schema. Furthermore, these expert schema are based on solution models instead of surface structures.

Cognitive load theory explains how we use our cognitive capacity to construct schema. There are 3 types of cognitive load (Paas et al., 2003a, Sweller et al., 1998a, Kirschner, 2002) (see figure 5.1):

- Intrinsic cognitive load is the cognitive load that is inherent to the task, and that is defined by the task complexity. Specifically, it is determined by the amount of information elements that need to be held in working memory at the same time in order to build the schema for storage in long term memory.
- Extraneous cognitive load is the cognitive load caused by the presentation and transfer method of the information. The extraneous cognitive load of a task should be as low as possible, because it does not directly contribute to learning and might even impair learning.
- Germane cognitive load is the cognitive load instrumental to or evoked by processes in which the schema in the long term memory are constructed and automated. When intrinsic and extraneous cognitive load together leave free memory space, it can be used to enhance construction and automation of schema. Researchers in cognitive load theory are in discussion to determine whether germane cognitive load is spend on schema building and automation itself, or to process instrumental to schema building and automation.

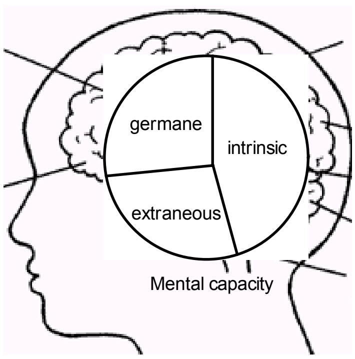


Figure 5.1. Three types of cognitive load

Consider for example learning from a school book (see figure 5.2). When a student learns from the book he spends cognitive effort to understand the information in the book. Depending on the complexity of this information the student will need to make more or less cognitive effort to understand this information, and thus the information in the book will pose an amount of intrinsic cognitive load depending on the complexity of the information in the book. To read the book and to look at the pictures in the book, the student also makes some cognitive effort. If the information is written in an elaborative way this will cost more cognitive effort than if it is written in a condense way. This effort is the extraneous cognitive load caused by the way the information is captured. As the mental capacity is limited, schemas are only built if there is sufficient mental capacity to understand the task and to build and automate schema, and if the student is motivated to learn. When the student already has (partial) schema of the information, he needs to spend less cognitive effort to store the remaining new information. Cognitive load thus depends on the learner.

Cognitive load novice student

Intrinsic cognitive load: understanding the information	Extraneous cognitive load: reading the book	Germane cognitive load: Instrumental to building & automating schema

Cognitive load experienced student

Intrinsic cognitive load: understanding the information	Extraneous cognitive load: reading the book	Germane cognitive load: Instrumental to building & automating schema	Spare capacity

Figure 5.2. Example of relative cognitive loads for a learning task.

5.1.2 Cognitive load and transferability

In Collaboration Engineering, the practitioner requires extensive schema of the collaboration process prescription and the collaboration process itself in order to apply it to the different instances of the collaboration process. Therefore, a practitioner needs to develop an expertise (high quality schema that are automated) in executing the collaboration process prescription. The cognitive load of the practitioner task is the mental effort required from the practitioner to perform his task as described in the process prescription and to learn and understand the skills required to execute his task. Therefore, a transferable Collaboration Engineering process prescription should pose a low intrinsic and extraneous load, to ease both training based on the process prescription and the execution of the process prescription.

The collaboration process prescription should support the transfer of facilitation skills and methods to the practitioner. Practitioners are not facilitators. However, the practitioner’s task is part of the facilitation task, and it does require expertise in some facilitation skills and responsibilities. Furthermore, the Collaboration Engineering approach prescribes that the collaboration process is re-used in different instance of the process. This can be considered a “far transfer task” meaning that the skills and information learned should be applied to different situations (Merrienboer et al., 2002). Far transfer is more difficult than near transfer, and requires more generalized and more abstract schema. To allow practitioners to perform as an expert on their task aspect they need to be able to make the schema that distinguish experts from novices for these specific task aspects. In other words, practitioners should become experts in executing the single recurring process prescription designed for them. The process prescription should therefore support the construction and automation of high quality schema of the process to build expertise. This expertise is required to reduce the cognitive effort required for the recurring elements of the task (giving prescribed instructions), so more cognitive capacity is available for the non-recurring aspects of the task (dealing with group dynamics), thus enabling successful far-transfer of the task (Merrienboer et al., 2002).

Further, because a collaboration process is a human process, not all aspects are predictable; it is likely that adjustments are required. When a practitioner has to adjust the process, to accommodate characteristics of the specific instantiation of the process a problem solving task emerges. This poses additional intrinsic cognitive load. Moreover, to add this experience to the memory and to develop expertise, additional cognitive load to build schema will be required. This is corroborated by the results of den Hengst et al, (Hengst et al., 2005) who studied demand rates of facilitation functions, and indicated that the demand rate of managing conflict and negative emotions, which often occur unexpectedly, is higher than the demand rate of other facilitation tasks, which are characterized as "slightly demanding".

Reducing intrinsic cognitive load will reduce the intrinsic complexity of the process prescription and thus of the execution task, which will make it easier to execute the process prescription (execution load reduction). Reducing the extraneous and intrinsic cognitive load, and thus increasing the capacity for germane cognitive load will, in case of sufficient motivation, make it easier to learn and understand the collaboration process, and will thus reduce the training load. This constitutes the process prescriptions transferability.

We define transferability as follows:
Transferability of the collaboration process design = the 'ease of training' and the 'ease of execution' from the perspective of the practitioner

The ease of training is determined by the training load:
Training load is the amount of cognitive effort required from the practitioner to sufficiently understand the process prescription.

The ease of execution is determined by the execution load:
Execution load is the amount of cognitive effort required from the practitioner to execute the process prescription.

Transferability has thus two aspects, the extent to which the process prescription and the activities involved in execution are easy to learn and understand, and the extent to which the process prescription offers the practitioner support for successful the execution of the task. Both lead to successful execution.

- Training load is the cognitive load that is experienced during the training phase, posed by the intrinsic complexity and extraneous load of the process prescription and the methods used in the training and process prescription template to free capacity for germane cognitive load.

- Execution load is the cognitive load that is experienced during the execution of the collaboration process, posed by retrieving the internalized process prescription, the use of memory aids, learning from experience and especially the group dynamics to which the practitioner is exposed.

The effect of transferability; training load and execution load on the performance of the practitioner is depicted in figure 5.3. The practitioner's performance is affected in a direct and in an indirect way. Directly it is caused by the execution load; the complexity of executing the process prescription. Indirectly the training load of the process prescription affects the training efficiency and effectiveness and therewith the competence of the practitioner as a result of the training. We will first explain the indirect effect. The expertise of the practitioner is established in the training and during the first experiences with the task. This expertise is affected by the talent of the practitioner and by the quality of the schema build. The talent of the practitioner is a multi dimensional concept. It can consist of previous experience with groups such as management or teaching, it can be character and charisma, and it can be something like 'feeling' for the task. Facilitators (see textbox 4.1 in chapter 4 for method) mentioned important skills required for group support such as being receptive, being self aware, being in control, being analytical and being focused. A theory like multiple intelligence theory (a person excels in a few of 7 (later discussion about adding several more has emerged) intellects such as logic intelligence, linguistic intelligence, interpersonal, spatial, etc.) would support this (Gardner, 1983). Next, competence is developed though the building of high quality schema in the long term memory that contain the process activities. Process activities contain interventions that should be made and outcomes and effects that should emerge as a result. The more competence, the easier it is to link new concepts, and the more information is recognized; therefore the required cognitive load reduces when competence increases; less schema have to be built. This implies that when the practitioner executes the process prescription on a regular basis it will become easier to prepare and execute the process, and the practitioner will learn more about the non-recurring aspects of the task.

To create a transferable collaboration process prescription we thus need to reduce the extraneous and intrinsic cognitive load of the process prescription to reduce its execution load, and to reduce training load and therewith free cognitive capacity for germane cognitive load which allows for the building and automation of schema. The more detailed schema are built, the larger the competence of the practitioner. Unexpected events increase intrinsic cognitive load of the execution task compared to the complexity described in the process prescription, but also require adjustments which constitute adjustment/ refinement of schema and thus additional cognitive load. Last, the thinkLet conceptualization will offer the information required for each activity in the collaboration process prescription in such a way that the schema built by the practitioner will be closer to the characteristics of expert schema instead of those of novice schema, therewith supporting the building of high quality schema. Design patterns, as discussed in chapter 7.1.1, describe problem-solution combinations and therefore the use of thinkLets, conceptualized based on the design pattern principle, to transfer a collaboration process prescription is likely to support the building of high quality expert schema in which solutions are linked to problems, instead of novice schema without such structure that would be build in case learning was achieved trough experience and without the structure of design patterns. This implies that we support the practitioner by increasing the quality of schema build. We will label this the thinkLet concept effect.

The direct effect of the process prescription on practitioner performance is determined by the difficulty of executing the practitioner task. This is affected by the complexity of the task as described in the process prescription but also by the complexity of generic tasks of the practitioner, such as listening to the participants. The difficulty of the practitioner's task increases when unexpected events happen (low predictability of the design and high group dynamics).

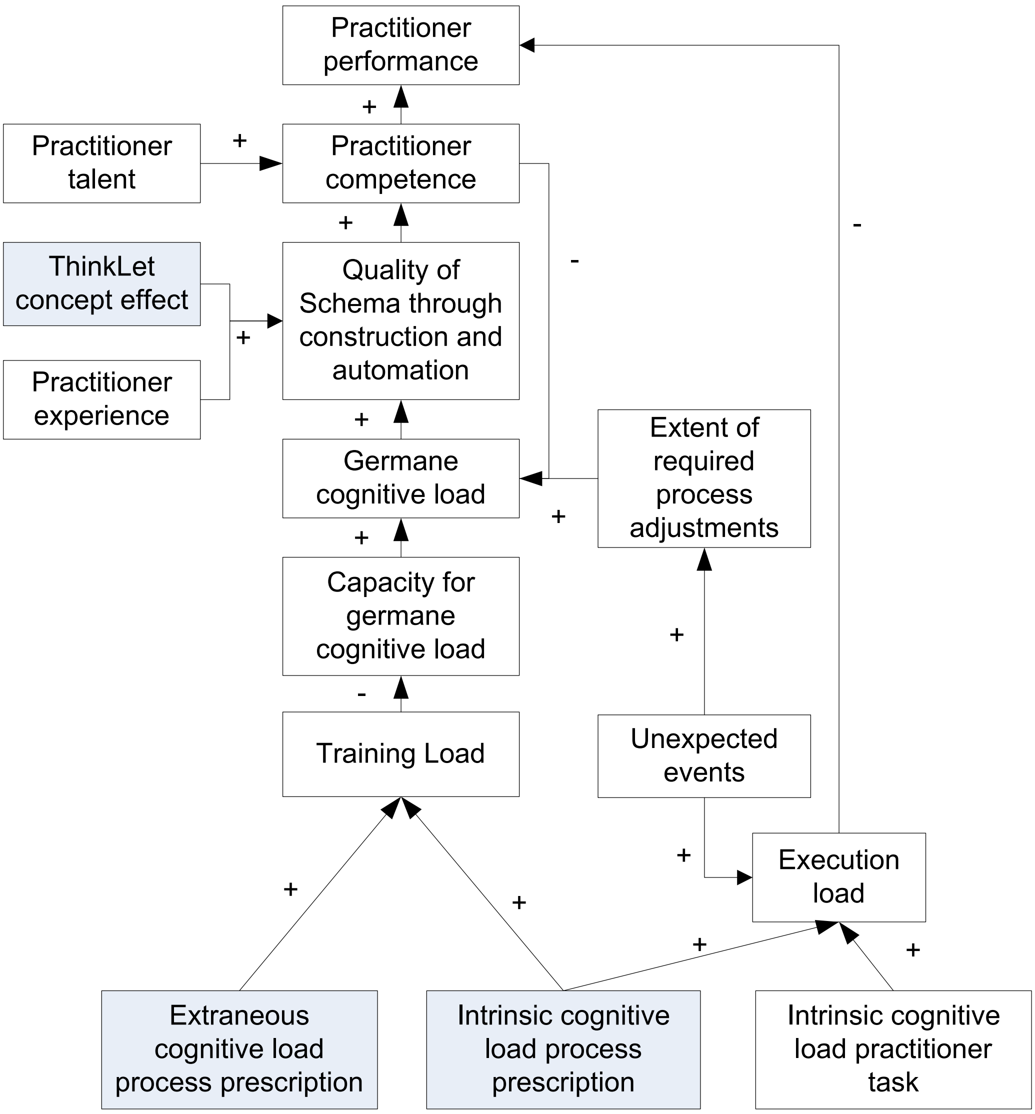


Figure 5.3. The effect of transferability on cognitive load and practitioner performance.

The dark blocks in figure 5.3 will be altered in this research to affect the transferability of the collaboration process prescription. To identify how we can reduce intrinsic and extraneous cognitive load of the process prescription we will first analyze the task of the practitioner and its intrinsic cognitive load. Next, we will analyze how we can reduce the cognitive load of the process prescription, and we will further elaborate on the effect of the thinkLet concept on cognitive load.

The learning process from a domain expert in an organization to become collaboration process practitioner for a recurring process in his domain has at least four stages; the training, the preparation of the first session, the first sessions in which there is still a learning curve visible (sometimes these can be trials), and the later iterations of the preparation and execution task, during which a stable level of performance should be achieved. The collaboration process prescription supports mainly the first 3 stages. Once the process is executed several times, the practitioner will internalize the process prescription and will not need supporting materials anymore.

We assume that a low cognitive load is of particular importance during the execution of the task; unpredictability, and the cognitive load (intrinsic and extraneous) of executing the process prescription will leave very limited cognitive capacity to learn and construct schema. As we also stated in the conclusions of chapter 2, learning by doing is not always an option. The practitioner has to be equipped to perform well the first time he executes the process. In the training and preparation of the process, the practitioner can take time to learn, understand and apply the instructions of the collaboration process prescription, and to memorize it, but during the execution his mental capacity is mostly occupied with the execution of the script, dealing with content (the practitioner is also a domain expert and will have to present information to the group), and with being receptive and responsive to unpredictable effects that require intervention. There will be limited capacity left to learn from the experience of the execution. Over time, the instructions task will be automated and more capacity will be available to learn from each experience. This is summarized in figure 5.4.

Intrinsic understanding the design	Extraneous reading design	Capacity for germane memorizing design

Training process

Intrinsic executing preparation instructions	Extraneous reading preparation instructions	Capacity for germane memorizing preparation

Preparation process

Intrinsic & extraneous executing process instructions	Intrinsic executing content	Intrinsic reception and response to group dynamics	Germane experience learning

First execution of the process

Intrinsic & extraneous executing process instructions	Intrinsic executing content	Intrinsic reception and response to group dynamics	Germane experience learning

Later executions of the process

Figure 5.4. Relative cognitive loads during the different stages of the practitioner task.

To further understand this division of cognitive load during execution we will analyze the cognitive load involved in the task of the practitioner.

5.2 Transferability of the practitioner task

The practitioner task is described in section 2.2.2 in chapter 2. In this section we will further elaborate on the cognitive load involved in the different aspects of this task.

Intrinsic cognitive load of the task

Practitioners are trained to execute a recurring process. However, each group, and each instance of the process can be different. Depending on the collaborative task, iterations can vary on content, participants, subject, etc. Development of practitioner competency is thus, not just executing the process prescription, but being able to apply the process to different instance of the situation (Kirschner, 2002). This requires expertise in the execution of the collaboration process prescription. The intrinsic cognitive load is determined by the complexity of the task. (Wood, 1986) and (Pollock et al., 2002) describe task complexity as the number of element and relations among those elements, and further, Wood adds the complexity dimensions “amount of uncertainty” and “dynamics of the task”. For intrinsic cognitive load it is important to assess how many elements need to be considered at a certain point in time. The complexity and cognitive load is different for each execution task, and for each collaboration process. However we can give an indication of the complexity of the execution task elements listed in chapter 2 (Vreede et al., 2002, Niederman et al., 1996, Dickson et al., 1996, Hayne, 1999, Clawson et al., 1993, Ackermann, 1996).

- Goal and outcome focus
  Goal and outcome focus requires that the practitioner constantly compares the progress and results with the goal and deliverables stated. When the task is large with complex deliverables this will require the processing and comparison of many information elements and thus a high intrinsic cognitive load. In a simple task with a clear single objective, focus might be easier to maintain.

- Structuring and focusing discussion
  Structuring and focusing discussion is more difficult and less predictable. While the goal should be determined in the process prescription, the focus of the discussion might be less clear. Some participants might need some reasoning and arguing to make their statement. The unpredictable factor in this task makes it more complex and results in a higher intrinsic cognitive load. This can be reduced by clear objectives and discussion rules or guidelines, but the unpredictable element remains also in simple tasks

- Presenting content information, asking questions, interpreting results and decisions, and giving feedback.
  Practitioners are expected to be domain experts and thus are supposed to have schema of knowledge about the content of the meeting. Despite the complexity of the task, we expect that the prior knowledge of the practitioner should reduce the cognitive load of this task aspect. However, it will remain a significant source of intrinsic cognitive load. In facilitation literature there is a discussion about the extent to which content and process guidance can be combined. In Collaboration Engineering we want to focus on domain experts because the collaboration process should be part of their task in the organization. However, when a practitioner is a domain expert he will be expected to give the group input and feedback on content as well as supporting the process. This can pose some additional challenges.

- Guiding rules for behavior
  To support the management of the group and to make group behavior more predictable, behavioral rules can be designed. However, after presenting these rules the practitioner will have to make sure that participants obey the behavioral rules, and will have to correct behavior that is not constructive and violates the rules. Again this requires comparison of the group behavior and the rules set and thus a high intrinsic cognitive load, especially when participants to not follow the behavioral rules.

- Ensuring participation of all stakeholders and inclusion of their interests
  While many collaboration techniques are based on the principle that all participants should be able to contribute, ensuring that they do requires attention from the practitioner. Stakes are not always open on the table. Stakeholders can have hidden agenda's or might even try to obstruct the process. Other stakes or interest might be less relevant to the goal, and thus interfere with the process. When stakes are hidden or unknown, the uncertainty element again increases the intrinsic cognitive load and complexity, especially for tasks in which the participation of many stakeholders is required.

- Be sensitive to the group and accommodate their needs
  Like the inclusion of stakeholders, the complexity and intrinsic cognitive load of this task depends on the amount of specific group needs and their clarity. A collaboration process prescription should be attuned to group characteristics, but this can be based on a generic group profile, rather than knowledge of the specific group, when the process will be used with a variety of groups.

- Support the use of tools and technology
  The process prescription will offer instruction for the use of tools and technology and a script for the tool instructions to the group. Depending on the complexity of the technology and the reliability of the technology, the complexity and intrinsic cognitive load of this task can vary.

- Manage the time
  Time management also can have a high intrinsic cognitive load; it requires comparison of progress, the current time, the end time, and, most difficult, the expected progress in the remaining time. The process prescription should be adjusted to the time required for the task, and the abilities of the practitioner. However, when unexpected events happen, or when things take longer than planned, time management becomes very difficult as it requires the practitioner to rush the group or shorten the process.

- Manage roles and responsibilities
  In simple processes there are two roles, practitioner and participant. However, more roles and responsibilities can be designed (e.g. a devils advocate (Janis, 1972)), in which case the actions of each participant should be compared with his role to attune and support this. Depending on the clarity of the roles and responsibilities, this task can have a varying intrinsic cognitive load.

While the cognitive load of each of each of these aspects can change in different contexts, the overview can help the collaboration engineer to get an impression of the cognitive load for execution of the task. When many of these task elements pose a high cognitive load, more time will be required for training and transfer.

The practitioner tasks that are not supported by the process prescription are mostly characterized by high uncertainty. An important judgment is when to intervene. Conflicts and discussions that are not designed might be essential in order to get shared understanding and to (re) gain goal congruence. Due to their unpredictable nature, these tasks can increase the intrinsic cognitive load unexpectedly, as discussed above.

The intrinsic cognitive load of the preparation phase might differ greatly both from process to process and especially from organization to organization. Firstly the complexity of this task depends on the amount of variables that are to be instantiated by the practitioner. For instance, a practitioner might be required to define the scope of the collaborative practice in each instance of the process. Other aspects such as arranging logistics and inviting participants might be more or less difficult depending on the organizational context in which the practitioner has to operate.

As we described above, the cognitive load of the practitioner task can be partly reduced by offering support during the process and by building and automating schema of the predictable aspects of the task. Automated schemas require less cognitive capacity to use and thus the more schemas we can build and automate during the training the better the practitioner will perform. In the next section we will discuss the cognitive load of the process prescription, and how we can reduce it to create an effective and efficient training approach.

Cognitive load caused by the process prescription

Besides the intrinsic cognitive load of the practitioner's task, there is the cognitive load associated with the process prescription. The cognitive load of a collaboration process prescription can be split in intrinsic cognitive load that is focused on the content and structure of the process designed, and the extraneous cognitive load that is focused on the representation of the process prescription. Further, it might be possible to offer support in the collaboration process prescription that can increase the efficiency of schema building and automation and the quality of those schema, affecting germane cognitive load. In the next section we will discuss ways to increase the transferability though the design of a template for the process prescription.

5.3 Increasing transferability: the process prescription template

With the insights from the cognitive load theory we can set requirements to the collaboration process prescription template. After presenting the process prescription template we will further analyze the transfer of the collaboration process to derive the training approach, and finally the transfer approach for Collaboration Engineering.

5.3.1 Requirements to the process prescription template

Techniques to reduce the cognitive load of the process prescription
Originally, the assumption in the research on cognitive load was that intrinsic cognitive load should be treated as a given and extraneous cognitive load could be reduced. This would then result in free capacity that could be used for germane cognitive activity and therewith increase learning. However, recent research has discovered that not only extraneous load can be reduced. There are techniques to decrease intrinsic cognitive load for complex tasks that even when offered with very low extraneous cognitive load, are too complex to understand at once, and thus very difficult to learn (Pollock et al., 2002). Given the amount of information that has to be transferred to the practitioner, the interrelatedness of that information, the amount of uncertainty and the short time frame available for transfer, we think that we should reduce intrinsic cognitive load as much as possible.

Techniques to reduce intrinsic cognitive load
The intrinsic cognitive load of the process prescription is determined by the amount of information offered in the process prescription and the interrelatedness of that information (Pollock et al., 2002). There are several approaches to minimize the complexity of the information that needs to be processed by a practitioner at a given moment when using the process prescription and during the training. However, it is also important to ensure that we offer the practitioner all information that is required to execute the task to minimize uncertainty.

Pollock et al (2002) suggest that in such situations information can best be taught in two steps. First the learner is offered a basic framework that can be schematized and in which interaction between information elements is mostly removed. This schema can then be used as a basis to learn the other material by offering the complete information with the interaction to the initial structure. In his example he offered the main procedure as a first task component, and the effect of each activity as successive component. This can be achieved by first introducing the practitioner with a high level overview of the general collaboration process and then explaining the specific function of each activity (Pollock et al., 2002). This is called the **isolated interacting elements approach** (see figure 5.5).

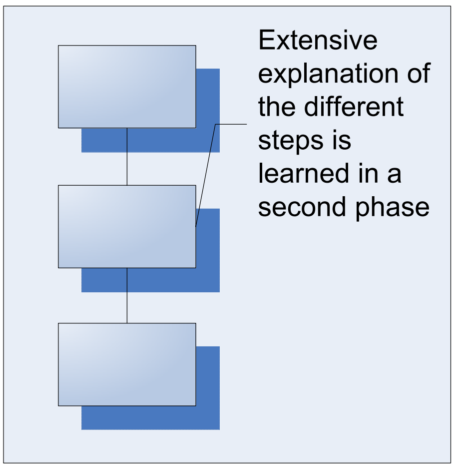


Figure 5.5. Isolated interacting elements approach.

A second approach to reduce intrinsic cognitive load is offering the task in smaller sub components (Pollock et al., 2002). A good way to distinguish process components is to use design patterns (see section 7.1.1). Patterns are especially useful for novices to a domain. We can offer practitioners design patterns that describe collaboration process components to reduce the intrinsic cognitive load. The component presentation has two effects; the **segmentation effect** (Mayer and Moreno, 2003); better transfer if material is presented in segments, that can be controlled by the practitioner, then as continuous whole process prescription, and the **pre-training effect;** better transfer when practitioners recognize the activities in the process because they learned their key characteristics in a previous step (Mayer and Moreno, 2003). The pre-training effect enables practitioners to build their memory schema faster, as they will first create basic schema of the design patters, and then add details to these schema, when they recognize the patterns in the extensive explanation of the interacting elements (See figure 5.6). This has important implications for the program of the training.

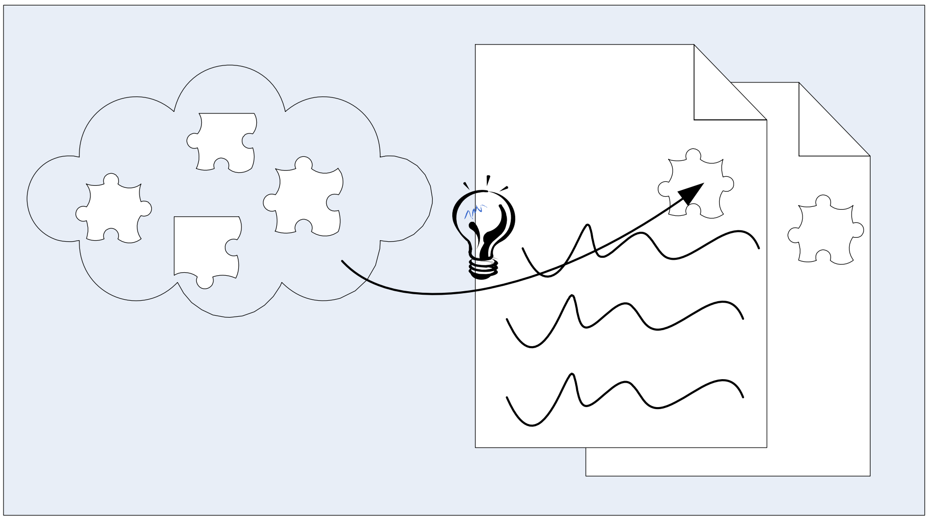


Figure 5.6. Segmentation and pre-training effect.

Naturally we cannot endlessly reduce intrinsic cognitive load. The practitioner should also get a collaboration process prescription that is complete and offers all information required to support the group in achieving their goal. Therefore the process prescription should offer a parsimonious description of the information required to recreate the pattern of collaboration. **Parsimoniousness** of the patterns implies that they contain all information required to recreate the pattern, yet no interesting but extraneous information (Mayer and Moreno, 2003). This is also called the **Coherence effect** (Mayer and Moreno, 2003). Redundant information increases cognitive load and has no added value to performance or learning (Sweller et al., 1998a, Mayer and Moreno, 2002). This effect is called the **redundancy effect** (Mayer and Moreno, 2003).

Concluding we can state that the intrinsic cognitive load of the process prescription is determined by the amount of information, the relatedness of that information and the predictability and dynamics of that information. The task of a practitioner is to execute the process prescription. The more predictable the process prescription, the lower the uncertainty and dynamics. A predictable process enables us to shift cognitive load from execution load to training load. The more we know about the process and what will happen the better we can transfer this information and train the practitioner to understand and execute the process. However, several tasks of the practitioner are unpredictable by nature and cannot be accommodated in the process prescription. Therefore the intrinsic cognitive load of the practitioner task; executing the process prescription can vary.

We can measure intrinsic cognitive load of the process prescription analyzing the amount of information, its interrelatedness, its completeness and the uncertainty that remains.

We can minimize the intrinsic cognitive load by:
1) Providing a parsimonious process prescription.
2) Offering relevant information as a reference framework of the practitioner such as the goal, objective, timeframe, deliverables, stakes, group needs, possible conflicts and possible challenges.
3) Offering an overview of the process and in successive phases the details of each activity.
4) Offering the detailed process activities in separate components that should be recognizable.

Techniques to reduce the extraneous cognitive load

Extraneous cognitive load is difficult to measure by itself. It is caused by offering the same information in multiple different forms and documents, from which the user has to distill the same lesson. Rather we can distill that lesson and offer it in one complete and consistent way. However, like intrinsic cognitive load, some extraneous cognitive load can be required, as offering information in different contexts will detail and improve the schema build. Furthermore, one representation (e.g. a picture) might better suite one practitioner while another representation supports another practitioner (e.g. a text), depending on their dominant intelligences (Gardner, 1983). Extraneous cognitive load is often assessed by inverse reasoning; if schema are constructed that represent the intrinsic complexity of the task then extraneous cognitive load must have been low enough to leave mental capacity for germane cognitive load (Merrienboer et al., 2002).

There are several techniques and methods to decrease extraneous cognitive load. A first technique is integrating information to reduce split attention. When information is offered in separate components (e.g. picture and separate text) and these components are not self-explanatory, both need to be held in working memory to process the information. Integrating text in a picture will therefore require less cognitive capacity (Sweller et al., 1998a). This effect is named **special contiguity effect** (Mayer and Moreno, 2003). Additional effect is achieved if cognitive load is spread over the visual and audio channel for instance by combining video with narration instead of video with text. This is called the **modality effect** (Mayer and Moreno, 2002) While animation would be very supportive in the transfer effort, the costs of creating animation or video instructions for each step of the process are likely to exceed the benefits unless a very large group of practitioners is trained. In a Collaboration Engineering project thinkLets need to be instantiated to the specific situation, and thus for each project a new video or animation needs to be created, which is costly and difficult. Instead, in the training approach we recommend practitioners, when possible to attend a collaboration process executed by a colleague practitioner.

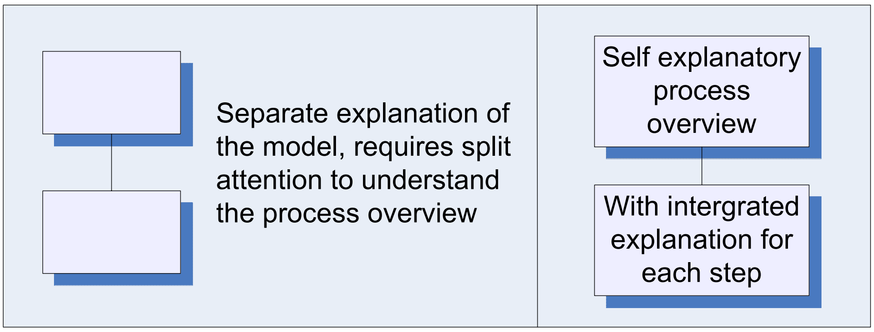


Figure 5.7. Special contiguity effect.

Concluding we can state that the extraneous cognitive load is decreased by integrating the textual explanation of the different components with the required visual aids (models and mnemonics). The process prescription should therefore contain the following feature to reduce extraneous cognitive load:

5) Both the process overview and components should be represented in a way that is self explanatory and sufficiently expressive.

Extraneous cognitive load is very difficult to measure directly, but since there are approaches to assess the intrinsic cognitive load and capacity for germane cognitive load we can reason that when intrinsic load remained the same and efficient learning occurred, the extraneous cognitive load must have reduced.

Techniques to affect germane cognitive load

In all approaches suggested above, germane cognitive load is not directly influenced. By reducing the other two types of cognitive load, capacity is made available to support schema building. Research has also indicated that learners do not automatically use this capacity; stimulation and guidance supports the use of this capacity and can increase germane cognitive load (Merrienboer et al., 2002). Also the variation of the application domain (context) of a task can increase germane cognitive load (Sweller et al., 1998a). The practitioner will have to apply the process to different situations and this will stimulate the building of schema that describe patterns in the execution of the process. An interesting question is whether the structure of components in the process prescription can serve as a template to constitute the basis of high quality schema that need to be constructed to store the required knowledge in the long term memory. This would require that we offer the information in a problem-solution combination (Sweller, 1988), the schema structure that experts use. Design patterns that will serve as a basis for the thinkLet conceptualization in chapter 7, offer problem-solution combinations. Therefore, the thinkLet concept is expected to increase the quality of schema build. Whether building such high quality schema based on problem-solution combinations will cost more, equal or less cognitive capacity requires further research; we expect that it will increase productivity;

creation of better schema in equal or less time. This renders our last insight to support transfer based on cognitive load theory:

6) The separate process activities should be presented as a problem-solution combination.

Germane cognitive load is difficult to measure, other than testing if the information is indeed captured in the long term memory and automated. This can be tested indirectly by measuring the practitioner's performance and their perception of mental effort. Another more direct approach is to measure if sufficient cognitive load is left, by testing the reflex of the learner during the transfer of information (Brunken et al., 2003). However, due to practical reasons we will not be able to use this approach for measurement.

Based on the insights in the cognitive load imposed by the process prescription during training and execution, and based on the methods to reduce this cognitive load, we can now derive requirements to the process prescription. With these insights we can reduce execution load, and manage training load to support efficient and effective learning. We will now describe how we will accommodate these insights in the process prescription template.

Requirements to the process prescription

Table 5.1 contains a summary of requirements to the process prescription and the cognitive load effects they should evoke.

Process prescription feature	*CLT effect*
Intrinsic cognitive load reduction	
1) Providing a parsimonious process prescription containing only important information	Decrease redundancy effect Coherence effect
2) Offering relevant information for the reference framework of the practitioner such as the goal, objective, timeframe, deliverables, stakes, group needs, possible conflicts and possible challenges.	Support far transfer
3) Offering an overview of the process and in successive phases the details of each activity.	Segmentation effect
4) Supporting the recognition of the components of the process prescription by pre-training.	Pre-training effect
Extraneous cognitive load reduction	
5) Both the process overview and components should be represented in a way that is self explanatory and sufficiently expressive.	Special contiguity effect
Germane cognitive load support	
6) The separate process activities should be presented as a problem-solution combination	Supporting the creation of high quality schema

Table 5.1. Requirements to the collaboration process prescription.

Summarizing the prescription of a collaboration process for transfer according to the Collaboration Engineering method, has four main components:

1. The description of relevant assumptions on which the process is designed, like the goal, the task, the group characteristics and the assumed resources that offer the reference framework for the practitioner. (requirement 2)
2. The overview of the process, the sequence of thinkLets, and other activities e.g. breaks and presentations. (requirement 1 and 3)
3. The thinkLet scripts, detailed prescriptions of the different activities in the process with explanatory information on the rationale behind the procedures. (requirement 6,3 and 4)
4. ThinkLet summaries on cue cards for use as memory aid during execution. (requirement 4)

5.3.2 Conceptual design of the process prescription template

We will explain each of the components of the process prescription template below. In appendix 6 we will offer an example of a collaboration process prescription according to the template.

Assumptions document

The process prescription is made for a specific recurring task with a specified goal. Several assumptions need to be made in order to design the collaboration process. The key assumptions should be known to the practitioner. They represent the reference framework for the practitioner to monitor progress, focus, conflict, etc. In chapter 2 we described the task of the practitioner during execution. For each task, specific information is required. Table 5.2 lists the different tasks and the information about the assumptions required to execute it. In the next chapter about the design approach we will explain how to analyze and specify this information. The assumption document is very specific for each process, but its completeness is very important. Without this information, critical tasks of the practitioner become unnecessary difficult. One important aspect to document here is the flexibility or extent of re-usability of the process prescription. For instance what is the minimum and maximum group size and what is the minimum timeframe required to execute the process.

Practitioner task	*Assumptions*
Maintaining goal focus	Goal, deliverables, scope
Structure discussion	Institutionalized methods in organization when used
Time management	Time frame, group size
Information retrieval, analysis and clarification	Content/Domain expertise required
Maintain guiding rules for behavior	Guiding rules for behavior,
Motivate participation and inclusion of stakeholders	Group background, Group experience, Group context, education level, org. culture
Support the use of technology	Available technology and tools, required knowledge of technology
Roles and responsibility management	Stakeholders/Actors involved and stakes
Assumptions about the practitioners skills and knowledge	The skills and knowledge required for the execution of the process besides those documented in the process prescription

Table 5.2. Assumptions that should be documented to support the different execution tasks of the practitioner.

Process overview
The second requirement to the collaboration process prescription is to offer an overview of the process. The overview describes a sequence of activities, and should contain sufficient information to offer a self explanatory overview of the process. For this purpose we extended the Facilitation Process Model (FPM) described in (Vreede and Briggs, 2005). The model should give an overview of all activities in the process and therefore needs to be extended with non-thinkLet activities.

The FPM focuses attention on the logic of the flow of the process from activity to activity. A FPM uses three symbols (see figure 5.8). This modeling convention represents each activity in a process as a rectangle with rounded corners that has been divided into five fields. In the left upper field, the sequence number is indicated and across the top the thinkLet name. Furthermore, a picture is included to enable the user to quickly link the activities in the facilitation process model to the scripts in the process prescription and the cue cards. The largest field contains a descriptive name for the activity that conveys what the group is supposed to do. The field on the left contains the primary pattern of collaboration to be instantiated in the activity, and in the right upper corner the starting time for the activity is indicated, while the duration is indicated along the arrow, together with the result. The time for a thinkLet is an indication. In many cases the practitioner might need to adjust the timeframe based on the task size of the specific instance of the collaboration process. When the practitioner prepares the specific instance of the task, the time for each activity should be indicated in “clock time” so the practitioner can directly check whether (s)he is on schedule. The process prescription should also offer support to the practitioner to determine the time for each activity in case this is flexible. We will further elaborate on this in chapter 6 and 7. Decision points are represented as circles and the decision with criteria is indicated along the arrows leading from the decision. The model can be used for training and as a reference for the practitioner during preparation and execution of the process.

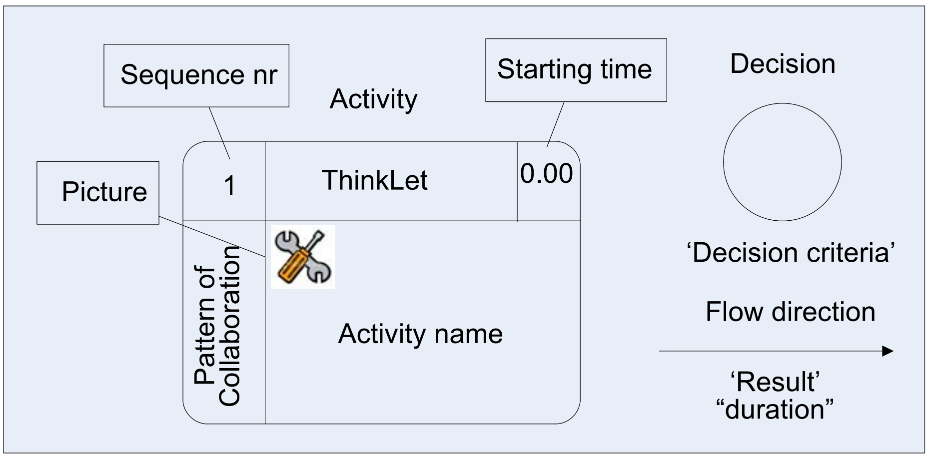


Figure 5.8. The symbols of a facilitation process model.

Scripts
Each activity of the collaboration process should be scripted for the practitioner. This script should be parsimonious; complete but without redundant information. This provides a trade-off. There are many known challenges and pitfalls that will not occur in each instance of a process, but are worth transferring. The coherence effect

prescribes that in order to get better transfer, interesting but extraneous information should be excluded (Mayer and Moreno, 2003, Pollock et al., 2002). However, the practitioner should be well-prepared in order to be able to run the process alone. Therefore we offer the practitioner 2 documents; one with the complete process prescription, and one with the information essential for basic execution of the process. This is accomplished by offering the script together with cue cards. The scripts can be made for thinkLets and non-thinkLet activities. For non thinkLet activities such as introductions, breaks, ice breakers, etc. a script with the key instructions and challenges should be offered.

Non-thinkLet scripts

The scripts for the non-thinkLet elements are situation dependent, but we will give some directions for the main non-thinkLet process elements, listed in table 5.3.

Aspect	*Scripting*
Introduction with scripted explanation of the goal, deliverable and overview of the process.	The goal and deliverables should be explained to the group, and the practitioner should check if they are accepted by all participants. When problems with respect to acceptance are anticipated, sufficient background should be available to convince participants. The process overview should be presented, but not with too much detail, to leave some freedom for flexible time-management. Also the process rules and behavior rules should be explained.
General preparations	A checklist with preparations can be made an should contain: - Roles involved in preparation - Invitation - Logistics (room, catering, materials, technology) - Preparing documents and hand-outs - Preparing thinkLet input when required.
Ice breaker, scripted	The script and the intended effect of the icebreaker should be clear. Some icebreakers can be documented as thinkLets; others can be described more simply.
Introduction to the technology used and small exercise (when complex technology is used)	A technology introduction contains an explanation of the tool, its benefits/advantages/characteristics, the function of the tool in the process and a short instruction/exercise to explain its use.
Presentations for content introduction, or to present results from previous activities	Content introductions should be as short as possible, and preferably, participants are briefed before the process.
Breaks	Breaks do not require scripting, however, it is important that practitioners use the breaks strategically. Breaks can be used when focus is lost, or for difficult decisions. When time is short, breaks can be shortened sometimes, but not eliminated, as this might reduce productivity.
Decisions in which the practitioner should choose a path in the process based on certain criteria for the output.	The decision instructions should contain clear criteria for the decision and should specify the margins in the decision and should describe how to explain the decision to the group.

Table 5.3. Non thinkLet scripts.

Aspect	Scripting
Wrap up, in which results are articulated, scripted	In the wrap-up the results are iterated, and compared with the objective. The script should indicate directions to explain follow-up activities and to identify action points and assign responsibilities when necessary. Furthermore, the wrap up should discuss what will happen with the results and when and how the participants will get feedback on the results.
Evaluation, scripted or other ways supported, for instance with questionnaire, with articulation of the purpose of the evaluation and instructions for processing the results	The organization can control the results of the process by evaluating it. Different approaches and evaluation criteria are possible. The evaluation process should be carefully scripted to explain the purpose of the evaluation and instructions to perform it; furthermore it should contain instructions for processing of the results.

Table 5.3. (continued) Non thinkLet scripts.

ThinkLet scripts

In chapter 7 we will describe the conceptualization of the thinkLet scripts, but we will briefly explain the instantiation of the thinkLets in this section. The script needs to contain all information to support the execution. Also, we need to document what will happen, so the practitioner can prepare him/herself for the challenges and dynamics that might occur. Last, we need to be able to link the thinkLet to the activities in the facilitation process model, and we should offer support for the memorization of the thinkLet. This can be documented in the identification information of the thinkLet.

Therefore, the ThinkLet script in a collaboration process prescription has three components.

1. The identification
2. The script
3. The "what-will-happen"

Identification

ThinkLets have a name, to identify them. The names are catchy, and contain a metaphoric element. Therefore they serve as a mnemonic to support the memorization of the name and the characteristics of the thinkLet (Buzan, 1974, Mayer and Moreno, 2003). This makes it easier to remember and transfer the thinkLet. To remember a thinkLet and to easily refer to it, the identification is strengthened with a picture and explanation of the metaphor (spatial contiguity effect). Once the metaphor is understood and remembered the name is a reference that can be used among practitioners to refer to a complex process (pre-training effect) (Mayer and Moreno, 2003). Besides these identification aspects, this component contains a brief overview of the technique. The process block from the facilitation process model is also printed in the right upper corner of the documentation of the activity to link the script to the Facilitation Process Model.

Script

The script contains the core of the thinkLet. It contains the rules that should be communicated to the group, and that should be maintained by the practitioner. In some collaboration processes, different roles can be distinguished. The rules are specified for each role, to instruct the specific tasks and responsibilities of the roles.

The script contains instructions on **what to do** and **what to instruct** to the group. Next, it offers a **checklist of the rules** that the practitioner needs to maintain.

What-will-happen

The what-will-happen, is a document that can be used as "background documentation" and learning material for the practitioner. The document explains the user what will happen when he uses the thinkLet, according to the script. It should indicate the pattern of collaboration that will emerge in the group, the result, the time it will take, and factors that might cause the thinkLet to take more or less time, challenges that can occur and the contribution of the thinkLet to achieve the group goal. Last, a successes story can be included to offer the practitioner an example of how and why the thinkLet works.

To decrease the cognitive load of this description we created a layout that will help the practitioner to quickly find the information needed. For this purpose each element of the thinkLet will have an icon to support recognition and navigation. The icons will be offered in appendix 6. Furthermore, the link between the Facilitation Process Model and the individual scripts should be clear, using the sequence numbers, names, FPM block and the pictures from the script in the model, the documentation and the cue cards.

Cue cards

The cue card is a small card that the practitioner can use during the session to execute the thinkLet. The cue cards should contain the thinkLet name, picture, pattern of collaboration, and the time allocated for the activity. The cue card should contain the script elements, the instruction for the set-up of the capabilities and the challenge-first aid and contribution from the "WhatWillHappen" document. Further, a cue card with the process overview (FPM) can be made. The cue card provides a summary of the essential information required to execute the collaboration process thus constitutes the coherence effect (Mayer and Moreno, 2003) and can increase learning efficiency (Kester et al., 2006). Practitioners should be encouraged to modify the cue cards to their needs, so they contain only the information that they expect to find difficult to remember. Besides the thinkLets, cue cards for non thinkLet activities can be included.

Besides the process prescription practitioners get a set of slides that they can use to present the introduction, and to introduce the different activities.

To evaluate and improve the process prescription we used it in several test case studies. In these case studies we will use the process prescription template to transfer a collaboration process prescription to (novice) facilitators and practitioners. In the next section we will discuss the findings from this evaluation.

5.3.3 Case study: transferability of the process prescription template

To evaluate and improve the transferability of the collaboration process design according to the process prescription template and the thinkLet scripts, we studied a set of cases in which we trained practitioners and under graduate students to facilitate a collaboration process. To evaluate the transferability and the effect of the collaboration process prescription template we did both an evaluation interview with the practitioners and we ran a questionnaire among the participants of the collaboration processes to evaluate the results. The questionnaire and the interview were preliminary versions of the instruments used in chapter 8. Due to time constraints and because the pre-knowledge and experience of the practitioners we did not yet use the training approach in these cases. Based on the initial cases we improved the process prescription template, and this improved version was evaluated in a second iteration of the same case. We will describe for each case study the results with respect to the transfer of process prescription template. The lessons learned for each case that informed improvements of the process prescription and the training approach described in the next section are listed in appendix 7. As insight in the template increased we also improved our measurement framework. This introduced some limitations concerning cross case comparisons which will be addressed in the result descriptions. The case numbers indicate the sequential order in which they were performed, and in which improvements were made to the process prescription. The pairs of case studies described below discuss a preliminary and more mature version of the process prescription used for the same case. See figure 5.9:

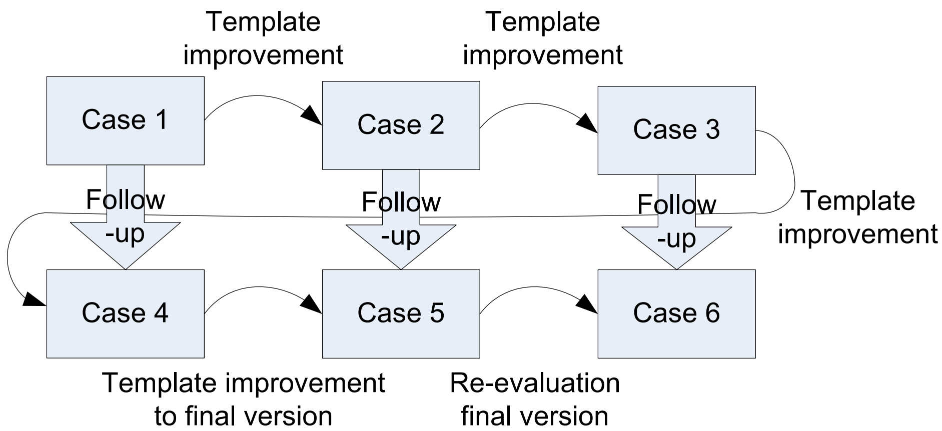


Figure 5.9. Order and relation of cases.

The results on the questionnaire to measure quality of collaboration are listed in table 5.4 below. “Case” refers to the case described in this section “n” refers to the amount of practitioners; “gs” refers to the (range of) group size. Delta (Δ) is the difference in the results between the initial case and the follow up case in which the same session was facilitated but a more advanced version of the process prescription was used. The number of the case describes its sequential order, but the case is discussed in combination with its follow-up case. In the table the average of each quality dimension of collaboration is measured on a scale from 1 to 7, 1 indicating a low score, 7 a high score. In case 1 and 2 efficiency, effectiveness and productivity were evaluated based on 3 constructs, in later cases all quality dimensions were evaluated based on 5 constructs. (See also section 8.2 and appendix 10a for the final version of the instrument, which was under development during these case studies)

We see that most cases are evaluated neutral to slightly positive, where the business cases were evaluated better than the student cases. Especially with respect to the results (satisfaction with the results and effectiveness) the results are slightly negative. Furthermore, we see that commitment ratings vary. Especially in the student cases the stakes involved in participation were relatively low. This requires the facilitator to spend significant effort on motivating the students. In the business cases commitment was higher. We will now briefly discuss each case, the results and the follow up case.

Construct	Case 1 n 2 gs9	Case 4 n1 gs14	Delta	Case 2 n3 gs 27-30	Case 5 n4 gs 9-19	Delta	Case 3 n6 gs 8-11	Case 6 n6 gs 12-17	Delta
Satisfaction process (5)	4.3	5.1	**0.8**	4.0	4.5	**0.5**	4.7	4.8	**0.1**
Satisfaction result (5)	4.2	4.6	**0.4**	3.5	3.9	**0.4**	4.0	3.9	**-0.1**
Commitment (5)	5.8	5.6	**-0.2**	4.3	4.6	**0.3**	4.8	4.8	**-0.1**
Efficiency (3-5)	5.0	5.4	**0.4**	4.3	4.8	**0.5**	4.7	4.7	**0.0**
Effectiveness (3-5)	4.0	4.4	**0.4**	3.7	3.8	**0.0**	3.9	3.8	**-0.1**
Productivity (3-5)	4.5	4.9	**0.4**	3.7	4.6	**0.8**	4.3	4.2	**-0.1**

Table 5.4. Results from 6 initial case studies on the transferability of the collaboration process prescription according to the template.

Case 1 and 4: Practitioners case consultants

In this case study two consultants from a medium-size consultancy firm focused on construction projects were trained to run a risk assessment session with the use of GSS. The purpose of this project was to add this skill (GSS based risk assessment) to their portfolio, and to use it for different groups and different organizations. The consultants both had experience in running risk assessment sessions with pen and paper, but did not have GSS experience. To run the process with GSS the activities in the process did not change, but the facilitation techniques used were altered to take full benefit from the GSS functionalities.

The process prescription and thinkLet selection was done in collaboration with the practitioners, as they also functioned as project managers. After this participative design effort, the process was documented according to a preliminary version of the process prescription template and the thinkLet script in which the following elements were included:

- The facilitation process model
- The thinkLet scripts containing identification, a script with slightly different elements but conveying rules, do this, say this and more detailed explanation of actions, parameters and capabilities (see chapter 7), and the what-will-happen document
- A script for the introduction and wrap up
- Cue cards that they modified themselves, adding memory aids that where important to them.
- Sheets to present the introduction and the different instructions to the group

There was thus no assumption document and the icons for the thinkLet script were not used in this preliminary version of the process prescription.

In the transfer first the general process was discussed based on the facilitation process model and the process slides. Next, the script for each thinkLet was discussed, with focus on the essential script elements and challenges. Furthermore, the GSS tool for each activity was demonstrated and collaboration with the chauffeur (experienced technology operator) was briefly addressed when needed. The practitioners re-read the script and altered the cue cards customizing them as their personal memory aids to further prepare themselves. In total the practitioners spend 12 hours each to prepare for the facilitation task.

Results

The resulting collaboration process went mainly as designed. While the practitioners thought they executed the script as instructed, the collaboration engineer observed several instances in which important instructions or script elements were forgotten. Most important were the purpose of different activities and the explanation of the rationale behind the process prescription. The process prescription and the transfer were considered complete and information was offered in a useful order.

Follow up

In the follow-up case study the practitioner was a more experienced facilitator/ consultant. This practitioner worked normally with pen and paper tools to support groups, so the techniques and GSS where mostly new. The transfer had to be fast due to time constraints, and thus was fit in 1,5-2 hours. In the transfer first the process model was discussed. Then we discussed the characteristics of GSS and each activity of the script in detail. The importance of activities and the emphasis on rationale discussed above were discussed during the transfer. Furthermore, the icons in the template were included for the first time, and the assumptions document was included. However, several aspects of the lay-out were different than the prescription in appendix 6.

The practitioner did not prepare in addition to the transfer described above, and did not use the cue cards. Also the icons in the script were not used in any way. During execution of the process, some reminders were offered to the practitioner by the researcher, and the researcher introduced the GSS. There were only two large deviations from the script, the instructions for voting were too limited and the interventions to support the discussion about the voting results were not as scripted.

Results follow up

The practitioner found especially the voting activity different than expected and expected more from the brainstorm. The convergence activity that he anticipated as difficult ran smooth.

The evaluations from the participants where rather positive, which is partly because of the experience of the practitioner, however, the template seems to become more complete, and cognitive load seems to be moved more to the transfer than to the execution. Despite the experience of the facilitator the metal effort is rather low, also in comparison to his colleagues in case 1. As an experienced facilitator the practitioner should have been able to identify information that was missing in the process prescription and transfer. As this was not the case we feel that we have achieved some level of completeness.

Case 2 and 5: Practitioners case undergraduate students

In this case study we let undergraduate students, who participated in a course on facilitation of group meetings, execute a session for peers in a course on system design. The session concerned a process for requirements negotiation according to the Easy Win Win approach (Briggs and Gruenbacher, 2002). In the course on facilitation the background on the task of a facilitator and the function of collaboration support in general were discussed and tested in an exam.

The student practitioners got a similar process prescription as the practitioners in case 1. Again, in this preliminary version of the process prescription template, we did not yet offer them an assumption document, and the icons were not used. Furthermore, there was no improvement made to the lay-out of the script.

The practitioners got a full day workshop as part of the facilitation course in which they experienced and tried to facilitate several thinkLets. There were a few months between this workshop and the process transfer. One of the two practitioners had considerable experience as a technical assistant, had observed a large number of facilitated GSS sessions, and had a one time experience in facilitation. Based on the course experience we needed only one hour to transfer the process prescription. In this hour we explained the topic of the meeting and the assignment the participating students got. Next, we discussed the process in general. Last, we discussed the thinkLet scripts. Due to sickness of one practitioner, only two practitioners ran the process, one ran the process twice in a row.

The time for transfer was very short and especially the content of the session was discussed only marginally. In the Collaboration Engineering approach it is assumed that the practitioners are domain experts, but in this case study the practitioners had no more expertise on the topic than the participants. Furthermore, the practitioners did not study the documentation very extensively after the transfer (1-1,5 hours) as they already knew the thinkLets.

Results

The process went less well than the first case study. Student participants had to role-play to simulate stakes in the session, but they had difficulty playing these roles and thus the ‘simulation’ did not become very realistic. This resulted in low response rates in discussions and a slow process in which practitioners needed to spend much effort on motivating the participants. Second, the practitioners had difficulty to run the session in the time allocated, which increased the pressure to motivate and encourage participants. Therefore this became a large drain on the attention of the practitioner. As a result the practitioners started to run the session more as a demonstration. For this purpose they made significant alteration to the script. Furthermore, the added value for the participants, with respect to the content became considerably less. A last challenge was that the chauffeurs were relatively un-experienced this time and therefore the teamwork between chauffeur and practitioner was challenging. In the results we see that commitment of the participants was rather low, and consequently, execution of the process becomes more difficult as participants are less willing to make effort and share knowledge.

Follow-up case 5
The practitioners in the follow-up course had either followed a course on facilitation and group processes in which they facilitated the session from case 3, or they were experienced technical assistants for the GSS system, and observed many sessions in that role. The process prescription in this case study was the full version as described above and in appendix 6. The last improvements involved the link between the facilitation process model and the thinkLet descriptions by adding the FPM block in the left upper corner of the script pages that concerned the specific block.

The transfer again consisted of a one hour briefing, and the practitioners got the process prescription, slides to present the introduction and assignments in the session and background about the session content, and read this before the briefing. In the briefing we specifically addressed the challenges for each thinkLet and the issues with respect to the 'demonstration-character' of the session.

The practitioners prepared themselves by reading the materials, on which they spent 1-3 hours. One practitioner practiced in explaining things to the group. Some practitioners indicated that they would have read the prescription better if they could do it again, and some indicated that they would prepare more about the content of the meeting.

Results follow up
The researcher observed the session and added input in the brainstorm phase of the process to ensure that the content of the session was elaborate and meaningful. The process for all practitioners went as intended, and surprises were limited. Most surprising was the need to explain instructions in much more detail than expected and the results (how it appeared on the main screen in the GSS) and feedback from the group. Deviations from the script were limited, and most were caused by lack of time. Some practitioners shortened a step and altered the labels of categories to fit the input from the group. These deviations are good as they show the practitioner's ability to be flexible based on the input of the group and the progress in the process. One deviation was not anticipated, a practitioner re-did a sorting task with the group because the purpose was not clear and therefore the results were not useful. Again motivating the participating students in the demonstration setting was difficult, combined with the relative large group sizes, and they still felt insufficiently prepared with respect to the content of the session. One practitioner also indicated that (s)he was much focused on running the process prescription, while during the session and the reflection (s)he indicated that next time (s)he would focus more on the participants. All practitioners indicated that they would be able to train other student practitioners to run the same process, based on the process prescription.

Case 3 and 6: Practitioners case undergraduate students
The set up of this third case study was similar to case 2, but in this situation the practitioner students where master students while the participating students were bachelor students. Furthermore, the participating students needed the results of the session for a project that was part of the bachelor curriculum. The session supported the bachelor students in finding solutions for their project and comparing these solutions with a multi criteria analysis. In this case study the assumption document was introduced to discuss the background of the collaboration process with the

practitioners. In this case six practitioners participated, and the groups they supported were considerably smaller. A challenge was that the practitioners had to offer technical support to their peers and could not use experienced chauffeurs. For this purpose we added a GSS walk-through to the transfer and we offered an overview of the tricky settings and data manipulations with screenshots and instructions. Again the practitioners learned a generic set of thinkLets during a one day workshop as part of a course on facilitation, in which techniques were discussed, experienced and tried by the practitioners. In this case, the time between the workshop of the facilitation course and the process transfer was shorter. The practitioners had no facilitation or chauffeur experience, and only participated in a few GSS sessions as part of their study. Besides the workshop, the facilitation course offered some background on the task of a facilitator and the function of collaboration support in general were discussed, and tested in an exam.

The transfer lasted 1 hour in which the process was first discussed in general based on the assumptions, the process model and the slides and next we ran through the thinkLet scripts to discuss critical instructions and challenges. Practitioners prepared very differently. Some practitioners used the cue cards, some the script and some developed their own memory aid for execution. Others just re-read the documentation, and some of the practitioner's rehearsed in teams especially focusing on the use of the GSS.

Results

The process ran as prescribed, but different practitioners made different variations to the script. Some of these variations had a positive effect, but many involved forgetting script elements or less precise instructions. One of the thinkLets (FastFocus) was considered particularly difficult, as it required them to both lead a discussion and to capture and clarify the key issues rising from that discussion. The practitioners indicated they would have liked to exercise with this activity. In several activities (voting, final brainstorm) it was difficult for the practitioner to present the results of an activity to the group, and the script did not offer much support in this task. The rooms in which they had to execute the process were small, which caused inconvenience.

Follow-up case 6

The sixth case was identical to the third case but we used the final version of the process prescription template as in case 5. The practitioners had all followed a course on facilitation and group processes similar to the one described above, but with more self-study hours and less lectures/training. A key problem in this case study was that 4 of 6 practitioners did not speak the mother tongue of the participating students. This made it very difficult to support the process. Some sessions were (partly) held in English, others were executed in English while content was generated in Dutch. This made it very challenging to support the session.

Again, the transfer was very limited. The practitioners did the workshop in which they leaned and used a set of thinkLets, after which a walk through of the process prescription was the only transfer. Practitioners spend an average 2.75 hours on additional preparation after the walk through.

Results follow up

The setting of this case was a bit chaotic. Due to limited resources the practitioners had to execute the process in parallel with 2 other groups and practitioners in one very tight room. This made observation relative difficult. Practitioners mostly indicated that the session went as expected. One practitioner found the session more stressful than expected, one found the session more easy than expected. Practitioners were satisfied with the results, but most indicated that the results could have been better. However, this is also due to some problems in the projects of the participating students. Changes in the process were made, but these were smart deliberate changes, mostly to reduce the amount of alternatives in two rounds to increase efficiency. One practitioner indicated that (s)he forgot to mention some rules, mostly this involved insufficiently explanation of the rationale of steps.

5.3.4 Conclusions about the process prescription template

As described, each case has been repeated with a more advanced version of the process prescription. In this section we will look at trends in the results from the initial cases and their follow up cases. We measured the mental effort of facilitation as reported by each practitioner and listed the average mental effort per case in table 5.5. For the final two student cases the mental effort of facilitation was 5.3 on average on a 7pnt scale, more than medium but not 'high.' This indicates that some mental capacity was left to deal with group dynamics and to be flexible in the execution. This was confirmed in the observations. The practitioners in the final cases made more deliberate changes to the process, and forgot fewer instructions. Note that a limitation is that we changed the scale of measurement, as the instrument was further developed over the cause of the cases. In the table below, a correction for this is indicated. In the first two pairs of cases, the average mental effort of facilitation has decreased with the more advance versions of the process prescription. In case 3 and 6 the difference in mental effort was low but the average mental effort increased slightly. This can be explained by the language problems indicated. The results in the business setting were better than in the education setting. We think that one explanation could be that in the business setting commitment of the participants, and their stake in the results was higher, which makes it easier for the practitioner to get the group moving. Limitations with respect to this is result are that 'n' is limited, and that previous facilitation and GSS experience varies among subjects.

Case & follow-up	*First case n*	*Follow up case n*	*Average mental effort first case*	*Average mental effort follow up case*
Case 1 & 4	2	1	6.5 (1-9)	2.5 (1-9)
Case 2 & 5	2	4	8.3 (1-9)	5.3 (1-7) (≈ 6.8 for 1-9)
Case 3 & 6	4	6	6.5 (1-9)	5.3 (1-7) (≈ 6.8 for 1-9)

Table 5.5. Decrease of the mental effort of execution based on average.

In the final two cases the practitioners indicated that the manual was complete. Practitioners indicated that its structure was useful and that there was no real superfluous information although some elements were more useful than others, none should be removed. Most of the suggestions with respect to the final versions of the process prescription template concern additional explanation about the structure of the prescription, explanation of the use of GSS, and a demo session or try-out.

Additional explanation of the process prescription can be offered in a transfer training. In the cases above the subjects either did not have time for extensive transfer or did have sufficient background to study the process prescription without extensive transfer. In the next section we will discuss the transfer training of the process prescription to practitioners in a business setting in with practitioners had less or no experience or knowledge about facilitation and collaboration support.

5.4 Increasing transferability: the training approach

After discussing the requirements to the process prescription, we will now discuss the transfer approach for this process prescription. While we use the process prescription to transfer a collaboration process to practitioners and facilitators we did not yet use a deliberate training approach for the transfer of the process prescription. Further, we did not always work with completely novice practitioners. Based on additional lessons learned from cognitive load theory and on training approaches in literature we will derive a training approach to better transfer the process prescription. We will first describe the constraints on the training approach and next discuss methods to reduce the cognitive load of the training.

5.4.1 Requirements to the training approach

Constraints to the training approach

As described in the conclusion of chapter 2 the practitioner in many cases needs to execute the process right the first time, and even the initial sessions might be critical both for the self efficacy of the practitioner and for the reputation of the new approach in the organization. Therefore, the transfer is a critical step in the Collaboration Engineering approach. In the transfer the information in the process prescription is explained to the practitioner and the skills described are practiced. The key objective of the transfer is to equip the practitioner to execute the collaboration process prescription. To derive a transfer approach we need to consider the following constraints.

There are, according to Paas and van Merrienboer (Paas and Merrienboer, 1994) two approaches to transfer expertise; the product approach that focuses on productive task performance and the process approach that focuses on mimicking expert methods. As described before, practitioners need to become experts in the execution of the process prescription and they need to mimic the expert behavior of facilitators. The performance of the practitioners depends on the intrinsic quality of the design and on the successful transfer of the information captured in the design. In the next chapter we will offer support to increase the intrinsic quality of the collaboration process design; its efficaciousness and the acceptance of the design. This should enable high practitioner performance when the process prescription is executed as intended. Therefore we will focus our transfer approach on the explanation of expert methods and guidance in the appropriate execution of those methods.

Practitioners should be domain experts and consequently we limit the pool of employees that are eligible for the role of practitioner. This implies that we cannot choose practitioners solely based on their talent in dealing with groups. An assumption for training is that the practitioner has no facilitation skills or experience. It is therefore likely that there are no or only very limited schema of the collaboration process available and that the intrinsic cognitive load of the learning task will be high. Note that due to the expertise reversal effect (learning support for novices often causes redundant additional load for experts), a transfer approach based on this assumption will not be useful for more experienced practitioners (Kalyuga et al., 2003).

Clearly, just handing over the process prescription is not enough to transfer the knowledge and skills required for the practitioner to execute the process prescription. There is a need for additional transfer of tacit knowledge and skills. Practitioners need to perform part of the task of a facilitator. Facilitation is a skill, and thus practice and copying from the expert is important (Post, 1993, Ackermann, 1996). This requires a transfer approach in which the collaboration engineer can demonstrate the methods and in which the practitioners can try these methods and get feedback from the collaboration engineer. Therefore the transfer requires that the practitioner and the collaboration engineer meet face-to-face. Furthermore, it is important that the practitioner develops self efficacy about his task, which is difficult to transfer through a manual for instance. The collaboration engineer needs confirmation that the practitioner build sufficient self efficacy and feels equipped to successfully execute the task the first time. This requires face to face contact between the collaboration engineer and the practitioner.

There are several approaches possible for this transfer as described by McConnell (McConnell, 1993):

1. Group training – involves three or more persons who participate in a common learning activity along with a facilitator.
2. Coaching – uses one-on-one training which includes demonstration, lecture, and observation.
3. Mentoring – assigns an experienced employee to a new employee.
4. Self-Paced Learning – allows the learner to determine the rate of speed for mastering the concepts of instruction.
5. E-learning – used to describe learning activities conducted from the user's desktop via the internet.
6. Computer-Assisted Instruction- learning based on computer instructions in which the learners progress is tracked and used to direct the learner to appropriate learning material
7. Distance Learning Training – describes instruction in which the trainer is geographically separated from the learner.
8. Self-Study – refers to learning activities initiated and participated in by an individual.
9. Simulations – are controlled and standardized representations on a job, activity, or situation.

10. Lectures – include structured oral presentations delivered for information transfer.
11. Job Assignments – place an individual into the actual position, limited to a period of time, for which the goal is to learn part of the process.
12. Job Rotation – includes several assignments in a preplanned order or exchange of jobs with another person.

For a first introduction of the collaboration process in the organization, mentoring, job assignment and job rotation cannot be used as there is no present experience in the organization. The process prescription has to be transferred at a focused moment in time. In order to make the Collaboration Engineering approach more efficient than training or hiring traditional facilitators, transfer to the practitioners should be effective and efficient. Since the transfer of collaboration processes is information intensive, the cost of getting up to speed for individuals is comparatively high. They may not have the stamina to 'self-study'. Therefore, a training or workshop setting appears to be the most suitable basis of this transfer. Further, gaming elements and a hand-out or manual can be used.

We need to transfer skills and experience besides the documented information in the process prescription, which requires presence of a trainer. Lectures and self-study could be used as an augmentation to the training, but not as the exclusive means to transfer the skills and experience. Computer supported learning, e-learning, computer assisted instruction and distance learning might be possible but would require high training development costs. Since the process prescription is process specific, video or animation of the process needs to be customized to the specific situation. Such approach would only be efficient if a very large amount of practitioners is trained with the same process. In such case videos of pilots of the process should be made to show the practitioners different skills and pitfalls. However, in many cases privacy and security will limit the possibility to make a video report about the session. The training approaches left are group training, coaching and simulation. The training approach we derived will have aspects of each of these approaches.

Techniques to reduce the cognitive load of the training

As described in 5.3.1 we can use the isolated interacting elements approach (Pollock et al., 2002) to spread the intrinsic cognitive load over the different activities in the training, thus reducing the intrinsic cognitive load at separate moments during the training. This means that first isolated elements are explained, that is explaining elements of the process without addressing their implication on the group process, practitioner task or results. In a next step the details of each activity and their effects are explained. This phased introduction of the information will allow the practitioner to first build an initial schema of the process and then capture the more detailed information by adding it to the schema. For the process overview we can add an interactive training method: When students construct the process overview themselves, and get feedback on it (after completion), they will gain deeper understanding of the process (Moreno and Valdez, 2005).

A second method for managing intrinsic cognitive load is by dividing it in recurrent and non recurrent constituent skills. Non-recurring skills require interpretation of cognitive schema to adapt the execution of the skill to each variance in the instantiations of the collaboration process. Recurrent skills, however, require applying rules that are captured in automated schema. In order to master non-recurring skills, the practitioner needs to understand the theory and explanation behind each of the process activities (Merrienboer et al., 2003, Kester et al., 2006). In Collaboration Engineering this is the mechanism that gives rise to the effect of the thinkLet and its contribution to the overall objective of the collaboration process. The information for recurrent skills is procedural information, e.g. the script and the process overview. The cognitive load of the script is by nature significantly lower than the cognitive load of understanding its effect. Therefore several authors suggests to focus the training on explaining the theory and the effects of the collaboration process, while a just-in-time support method can be used for procedural information (Merrienboer et al., 2003, Kester et al., 2006) that can be offered as a script for use during the first exercises with the process. It is important that these scripts do not evoke temporal or special split attention. We expect that in many Collaboration Engineering cases there is very limited or no possibility to practice and we cannot offer the procedural information for the first time when the process is executed, as practitioners need to understand this information, and be flexible in case small modifications are required. However, we can prevent that practitioners have to learn the script by heart. For this purpose we will support practitioners with a summary of the procedural information on cue cards and (partly) in the session introduction slides, which they can use during the execution of the collaboration process. The cue cards will become redundant after several iteration of the task and should then be removed (Sweller, 2004).

To successfully execute procedures in various instances of the collaboration process, conceptual understanding is required. Conceptual understanding means, according to van Gog et al (Gog et al., 2004) that the practitioner should understand why the activities in the process are relevant to goal attainment and why they are performed in the particular order of the process prescription. Furthermore, understanding is required about specific challenges that can occur and associated solutions. In order to learn this information and to build schema of it, practitioners should self-explain this information (Renkl and Atkinson, 2003) and thus should be stimulated not only to study this information but also to experience or imagine what they would do in different circumstances, and whether this corresponds with the solutions offered in the process prescription. Self-explanation and imagination require some initial experience. While practitioners are novices to facilitation, it is likely that they have experience with group work from a participant perspective. Therefore we think that this type of training can be used with respect to the challenges of the different activities. This requires us to make two additions.

Once the thinkLets are introduced, the practitioners should do an exercise in which they make a first attempt to self-construct the process. In this way, a discussion can be held, to gain understanding on the order of activities prescribed. This information is required to enable the practitioner to explain the process and the added value of each activity to the group. The practitioner should be able to make this argument other ways he can fall in the trap of “putting the agenda on the agenda (Vreede et al., 2003a)”

The second addition is the imagination of challenges that can occur through simulation of the critical process activities. Challenges, known pitfalls, are documented for each thinkLet. They can be discussed during the practice by providing mini simulation cases/ scenario's in which the practitioner is asked "what would you do if…" Once high quality schema of the process and its individual activities are built, we can support schema automation though the stimulation of imagining the process. This will increase the automation of schema. A condition is that first, sufficient high quality schemas are built. This would therefore be one of the last steps in the training (Cooper et al., 2001). One way of stimulating this can be to ask practitioners to imagine things that could go wrong. This will stimulate them to rehearse the process and identify the elements they feel uncertain about.

Practice of the practitioner task is not always possible. When there is an opportunity to exercise the collaboration process in a pilot it should be arranged in a way that it allows the practitioner to make the critical non-recurring errors that he should be able to solve during the process (Rikers et al., 2004). This also allows the collaboration engineer to give explanatory feedback, which will further support the construction of the schema that contain the theory behind the procedural information (Moreno, 2004). Therefore, in simulation, the key challenges should occur. This can be done in a simulation setting, where roles are played and scripted to let the challenges occur. A basis for this simulation and practice of each step in the process can be found in the micro teaching approach, in which small modules are explained and then rehearsed by the trainees in small groups where peers and the trainer offer feedback on their performance (Education Encyclopedia, 2002).

Concluding the training should:

- Focus on offering a basis for the schema construction by first explaining isolated elements of the process and the process overview (thinkLet blocks), followed by the explanatory interacting elements (thinkLet scripts)
- Introduce the procedural information and let practitioner self-construct to enhance deeper understanding
- Offer procedural support during execution
- Enable schema building and self-explanation of the theoretical basis of each activity
- After sufficient schema construction, the training should support rehearsal and the imagination of what will happen when it is executed to automate schema.
- Offer practice with feedback, especially in the non-recurring tasks.

5.4.2 Conceptual design of the training approach

Based on the requirements above, we propose the following structure for a training program to transfer a collaborative work practice. We will illustrate each element of the training program in an illustrative case study below:

1. Lecture: short introduction on collaboration, the process, the role of the practitioner, when applicable the technology used, and the purpose of the Collaboration Engineering approach
2. Group training: self-construction of the process in an exercise.

3. Lecture: introduction of the process overview with reference to the process activities that are filled in with the interacting components
4. Lecture: explanation of the component conceptualization (thinkLet concept)
5. A sequence for each component existing of:
    - Lecture: (explanation): further explanation of the interacting components (techniques to accomplish each of the process activities and elaboration on the rationale behind each activity)
    - Simulation and coaching: to practice the challenges with feedback
    - Group training: discussion to invoke imagination of challenges
6. Self study: re- processing the information from the training
7. Execution support with cue cards and process overview, when possible with coaching

5.4.3 Case study: transferability effect of the training approach

To test the training approach we evaluated the different steps in the training and their effect on cognitive load during an existing case study at ING Group (Vreede and Briggs, 2005). ING Group is a large international financial services organization headquartered in the Netherlands. Following requirements from regulatory bodies, such as the Basle committee, ING was faced with the challenge to perform hundreds of operational risk management (ORM) workshops. They commissioned the development of a repeatable collaborative ORM process that operational risk managers could execute themselves. Based on the experiences and the requirements from the ORM domain experts, collaboration engineers developed a first prototype of a repeatable collaborative ORM process, the Risk and Control Self Assessment (R&CSA) process. This process was evaluated in a pilot project within a particular business unit, leading to a number of modifications to the definition of the overall process in terms of collaborative activities, their interdependencies, and the thinkLets used. The resulting collaborative ORM process was shown to a group of 12 ORM experts. During a half day discussion, the wording and order of activities was modified and the proposed collaborative activities were tested with a number of chosen facilitation techniques. In the period that followed, over 300 ORM practitioners have been trained to execute this process. To date, these ORM practitioners have moderated hundreds of workshops where business participants identify, assess, and mitigate operational risks. Below we detail the ING R&CSA training program following the structure presented in the previous section. The order of steps is not entirely similar as the ING training was developed before this thesis, however, the content of the steps does resemble the steps of our training approach, and its evaluation will therefore give insight in the training approach.

1. Lecture: short introduction on collaboration, the process, the role of the practitioner, when applicable the technology used and the purpose of the Collaboration Engineering approach

A practitioner should be able to explain to the participants in the collaboration process why a collaborative approach is used. For this a basic understanding of ‘what is collaboration’ and ‘why collaboration support is useful’ is required. This information can be presented by the trainer in a short lecture, and can be re-addressed in self-study.

At ING, the introductory part of the training includes a discussion part with the trainees about the value of collaborative approaches to risk and control assessment compared to more traditional approaches, such as questionnaires or one-on-one interviews. Further, the collaborative R&CSA approach is presented as one of the elements of the organization's portfolio of operational risk management tools and techniques.

2. Group training: self-construction of the process in an exercise.
3. Lecture: introduction of the process overview with reference to the process activities that are filled in with the interacting components

In this step the practitioners are encouraged to think about the sequence of activities and the thinkLets used for each activity. This will increase the understanding of the role of each thinkLet, the contribution and the order of thinkLets. In an interactive group setting the sequence can be built and adjusted to gain understanding in the process and the rationale behind the sequence of activities. Next, the final process overview is presented

At ING, this step was implemented by forming subgroups of 2-4 trainees. Each subgroup would get a set of 'cut' activities printed on foil that they could move around on a table to organize them in what they felt was the correct sequence. Each subgroup would present their results in a plenary session during which time discussions would be held about differences between the subgroups. After each subgroup had presented, the actual 'correct' sequence would be shown and discussed.

4. Lecture: explanation of the component conceptualization (thinkLet concept)

The collaboration process prescription consists of thinkLets. Before these thinkLets are explained, a brief introduction to the thinkLet concept and the elements and representations of the thinkLet is required in a short lecture.

At ING, the thinkLet concept is introduced in two steps. First the key patterns of collaboration in group work are introduced and illustrated: divergence, convergence, organization, evaluation, and consensus building (Vreede and Briggs, 2005). Second, thinkLets are introduced as facilitation techniques to consciously create a particular pattern of collaboration. The thinkLet concept is explained and anecdotes are provided to illustrate. This step was performed before the process sequence was created.

5. A sequence for each component existing of:

- Lecture: (explanation): further explanation of the interacting components (techniques to accomplish each of the process activities and elaboration on the rationale behind each activity)
- Simulation and coaching: to practice the challenges with feedback
- Group training: discussion to invoke imagination of challenges

Once the full process is understood, the details of the thinkLets can be transferred. The script is discussed and the purpose of each rule and instruction is explained. To practice the key challenges and pitfalls, the process can be simulated during the training. For this simulation mini simulation cases/ scenario's can be presented in which the practitioner is asked "what would you do if…" Once the thinkLets are discussed, it is important that the practitioner can imagine himself executing the thinkLet. To establish this, we can ask the practitioner to try and think about what can go wrong when the thinkLet is executed and what seems difficult. Known challenges and questions can then be discussed and suggestions and guidelines for solutions can be offered. For this step we use a group training setting.

At ING, the introduction and explanation of the thinkLets was chunked: A maximum of three thinkLets was introduced at a time before trainees would practice them. This part of the training made use of a practice case based on a fictitious insurance company. The explanation of each thinkLet was done in two phases. First, the actual execution of the thinkLet was explained. During this phase, most attention was paid to explaining the flow of the script. The trainer would normally act the thinkLet out in front of the group. Second, the rationale behind the thinkLet would be discussed. To this end, the trainer would ask the trainees to argue about the reasons behind certain execution aspects of the thinkLet. For example, in the DirectedBrainstorm thinkLet participants in a workshop can see and work on only one of several available brainstorm pages at a time. To practice each step, each trainee would in turn facilitate a part of the R&CSA process while his or her colleagues would act as participants. The case was designed to be general enough for all trainees to be executable, yet specific enough to provide some real basis for discussion and realistic group behavior. The case also included a number of specifically designed 'worst practices' so that the trainees would have a chance to experience the importance of certain parts of a thinkLet script, or the importance of making the right decisions during the preparation of an R&CSA workshop. The trainer would then ask the trainee who executed the thinkLet to first reflect him or herself. Then the other trainees, who acted as participants for the trainee executing the thinkLet, would be asked to share their thoughts and experiences. Finally, the trainer would address any relevant issues and best practices that would not have been discussed up to that point.

6. Self study: re- processing the information from the training.

The training still has to address many issues and include a lot of information, despite using techniques to reduce cognitive load. Re-reading the documentation might help the practitioner to further construct his memory.

At ING, the self study step was not a formal part of the training program. However, each trainee received various background materials before and after the training for further reading. These materials included articles on facilitation and operational risk management practices.

7. Execution support with cue cards and process overview, when possible with coaching.

When it is possible to practice the process in a pilot this can be useful. However, in most organizations, there is only one opportunity to make a first impression, especially with the introduction of a new role and a new approach. When practice is possible, it would be good to practice the non-recurring elements of the task, especially how to deal with challenges and questions. An alternative is to observe a peer-practitioner and to discuss among practitioners how to solve difficulties. To support the execution, the procedural information and the challenges and solutions should be available on the cue cards. This removes the need of spending precious training time on rote-learning the script. It is not likely that after one training event and short practice (if any) all challenges and solutions are captured in schema. Since this information is very important, it should also be available during the execution and thus on the cue cards. When possible, the practitioner should be observed and evaluated during the first execution by a coach, being the collaboration engineer or a more experienced practitioner.

At ING, the execution was not part of the formal training program. However, the execution step was supported in two ways. First, after taking the R&CSA facilitation training, many trainees sought out experienced R&CSA practitioners to observe a real R&CSA workshop and/or run such a workshop as a team. Second, some participants attended an advanced R&CSA facilitation training that focused especially on knowledge sharing among the trainees concerning ‘best R&CSA practices’. The advanced training also introduced the trainees to a number of new thinkLets to execute parts of the R&CSA process and paid in-depth attention to ‘soft facilitation skills’, e.g. how to deal with ‘difficult participants’ in a workshop.

Method and measurement framework

In order to test the training approach we need to create a set of evaluation criteria. In cognitive load literature the training or learning efficiency is assessed by comparing the mental effort required with the quality of the results in a test (Paas et al., 2003b). Unfortunately we do not have access to data that indicate the quality of the practitioner’s effort in this case study. Therefore we will measure only the mental effort and add to this a set of metrics to get a subjective indication of our ability to minimize the intrinsic and extraneous cognitive load and to support the building of schema.

First of all we want to address if we succeeded in minimizing the cognitive load of the training. As we defined above, the cognitive load is minimized if the training offers all information required and no superfluous information, and if the different training steps and information elements were all useful. This will give an indication to whether we succeeded to reduce the intrinsic and extraneous cognitive load. If elements of the training or documentation are not useful or superfluous, the training efficiency can be increased.

Next, to find if sufficient quality schemas are made and if they are automated we can ask practitioners if they feel equipped for the task (if they did not yet run a session) or if they were equipped for the task if they did run a session. Another indicator of this can be the need for additional preparation for the task (besides logistics and content preparation). Furthermore, the actual use of the process is an indicator that the training supports the building of schema. Finally, satisfaction and meeting the expectations of the practitioners are indicators that the training achieved the intended effect.

To measure the effective cognitive load of the training we asked the participants about the metal effort and difficulty of the training and whether the training was tiring. In Table 5.6 an overview of the constructs and metrics is offered.

Results
To measure the constructs described above we ran a questionnaire among practitioners after their training. The final version of the questionnaire can be found in appendix 10c, for this case some questions were removed. The instrument contains questions to evaluate each of the factors described in the measurement framework above. We ran the questionnaire in five training sessions among a total of 63 respondents. The results are presented below.

Construct	*Metrics*
Cognitive load reduction	
Usefulness	Training steps (per step)
	Process prescription elements (per element)
	Order of training elements
Completeness	Need for additional training
	Need for additional information
	Superfluous information
	Usefulness training
	Usefulness compared to other trainings
Quality Training	Feeling equipped
	Additional preparation activities
	Use training
	Satisfaction training
	Satisfaction compared to other trainings
	Training meets expectations
Cognitive load training	
	Mental effort
	Difficulty
	Difficulty compared to other trainings
	Tiring

Table 5.6. Metrics for the evaluation of the training approach.

Cognitive load reduction:

Usefulness
We examined the self-construction of the process (step 2), the usefulness of the process overview (step3), the thinkLet concept introduction (step 4), the thinkLet explanation and simulation (step 6 and 7), and the discussion of challenges (step 8). We measured Usefulness on a 7 point scale, 1 not at all useful, 4 neutral, 7 very useful. The results are listed in table 5.7.

Aspect	Usefulness
Usefulness self-construction of the process (step 2)	6.19
Usefulness process overview (step3)	5.61
Usefulness thinkLet concept (step 4)	5.94
Usefulness thinkLet explanation and simulation (step 6 and 7)	6.22
Usefulness discussion of challenges (step 8)	6.03

Table 5.7. Usefulness of the training steps.

Besides the usefulness of the training steps we asked participants about the usefulness of the training materials (see table 5.8); the background material (slides) the process overview, and the thinkLet cue cards.

Aspect	Usefulness
Background material	5.84
Process overview	6.16
ThinkLet cards	6.17

Table 5.8. Usefulness of the training materials.

Last, we asked if the material was presented in a logical order. 1= very much disagree, 4 neutral, 7 very much agree.

Aspect	Agreement
Logic order of presentation training material	6.05

Table 5.9. Order of presentation training material.

Completeness

We asked the trainees whether they wanted any additional training materials in advance, whether they wanted any additional training, whether the training materials were complete and whether there was no superfluous material.

Aspect	Percentage "yes"
Received sufficient material before the training	87.30%
Needed no additional training	31.80%
Training materials were complete	84.10%
No training material was superfluous	92.10%

Table 5.10. Completeness of the training and material.

Additional training requests focused on further background on being a devil's advocate, the use of a more realistic case, or a demonstration/video to gain more realistic experience and background on a 'high level version,' i.e. a shorter version of the process focused on top management. Finally, practice with the tools used and an advanced course or community of practice were indicated several times. All these training requests involve additional skills and knowledge that is not primarily required to execute the process. Only two practitioners wanted to get more training on dealing with emotion and conflict, one wanted coaching. The other training requests were for a different training than facilitating the specific process. Most trainees indicated that the training materials were complete and that there were no superfluous training materials.

Aspect	Agreement
Usefulness facilitation techniques	6.16
Usefulness training compared to other trainings	5.08

Table 5.11. Usefulness of training and facilitation techniques.

Training quality

To assess the training quality we asked participants whether they felt equipped to facilitate the process, what they would do for additional preparation, especially if they wanted to observe a real session, whether they intended to use the techniques learned, whether they were satisfied with the training, in comparison with other trainings, and whether the training met the expectations of the participants.

Aspect	Agreement
Feeling equipped	5.48
Intention of use	5.83
Satisfied with the training	6.13
Satisfied in comparison with other professional training	5.26
Training met expectations	5.93

Table 5.12. Training quality.

To prepare for their first session most practitioners indicated they would read though the materials, and many would make a customized process agenda. A few would ask support from an experienced practitioner and several wanted to practice in a safe environment. On the question whether they would attend a workshop before running it themselves, most answered "I would like to, but I'm not sure I can", several answered that they can not, some answered they would, while some answered they did not need to because the training was sufficient.

Mental effort of the training

To assess the cognitive load of the training we asked participants to indicate if they found the training required a lot of mental effort, was difficult and was tiring, and whether it was more or less difficult than other professional training. 1=very much disagree, 4= neutral, 7 = very much agree.

Aspect	Agreement
I found that the training required a lot of mental effort	5.15
I found the training difficult	3.65
I found the training difficult in comparison with other professional training	4.10
I found the training tiring	3.11

Table 5.13. Mental effort.

Mental effort was not neutral, but not high either, difficulty was about neutral and the training was not very tiring.

5.4.4 Conclusions about the training approach

The results from the questionnaire are very encouraging. It seems that the cognitive load of the training is not exceptionally high and the material and training seem to be rather complete and useful, practitioners feel equipped to execute the process and were satisfied with the training. Their intentions for preparation are to read the training material and only a few want more practice or support from more experienced practitioners. As we expected many of the practitioners were not sure they would have a chance to observe or try a session with another practitioner, and several indicated that they were sure that this was not possible. It appears that through the training we were able to sufficiently equip the practitioners to run a process as complex as 'risk self-assessment' by themselves.

5.5 Transferability support in Collaboration Engineering

Based on the training approach and our analysis of the task of the practitioner we can now derive the transfer approach for Collaboration Engineering. The approach, as discussed in chapter 2, has 3 phases, the transfer training, the preparation by the practitioner and his first execution. In the training, the approach is introduced, the process is introduced, the separate thinkLet components are introduced and then explained in detail, and the process is practiced or simulated to let the practitioners experience it. Last, challenges for each thinkLet are discussed. After the training, the practitioner should prepare himself for the first session. During this preparation he should verify the goal and scope for the specific iteration of the collaboration process, select participants, arrange tools and logistics and instantiate the variable content of the process such as categories, voting criteria and other session specific parameters. Last, he should revise the training material. Once everything is instantiated for the specific version of the collaboration process, the first session can be conducted. Of course, each session should be prepared separately. After the first sessions in which there is still a learning curve, the full implementation in the organization should be established (see chapter 2). The process is visualized in figure 5.10.

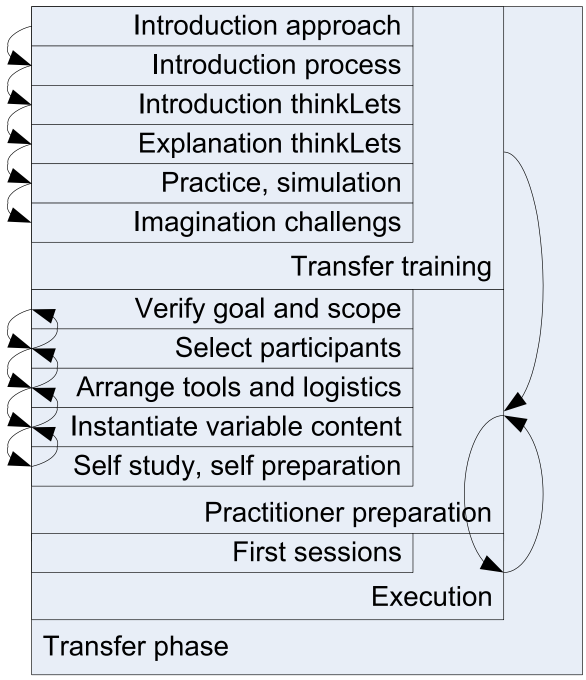


Figure 5.10. The Collaboration Engineering transfer process.

To support the training, practitioner preparation and execution we offered a template for the collaboration process prescription. Experience suggests that the template and the training approach will reduce the cognitive load of training and execution. They offer:

- Parsimonious instructions; script and supporting layout
- Identification and memory support
- What-will-happen description; enabling anticipation for far transfer
- Just in time execution support to reduce the need for rote learning

Furthermore, it can be argued that the thinkLet concept and the recurring facilitation process model blocks will increase the quality of the schema build. Combined, the transfer approach and template will increase the transferability of the collaboration process prescription. We found that practitioners feel equipped to execute the process and both the mental effort of the training and of the execution have been lowered to an acceptable level.

With the transfer process and its support established we can now turn to the design approach. We will offer design support to create a collaboration process prescription according to the template and hence transferable. Furthermore, we will offer design support to increase the efficaciousness and acceptance of the collaboration process to further support the practitioner in supporting the group to achieve their goal. In the next chapter we will therefore derive the design approach and design support. Based on the design and the transfer insights we will then further develop the thinkLet conceptualization in chapter 7.

Chapter 6. Efficaciousness and acceptance: design support

Besides transfer, a second challenge in Collaboration Engineering is the design of a collaboration process prescription. While experienced facilitators can adapt their process prescription on the fly to achieve efficaciousness and acceptance of the process and the interventions they make, practitioners need a process prescription that will reliably move the group to its goal and that will be accepted by participants as it is documented, without need for modifications on the fly. Therefore we need to better understand how we can support the design of a collaboration process prescription that will be efficacious, acceptable, transferable, reusable and predictable. To create a transferable prescription we can create a collaboration process prescription according to the template described in the previous chapter. However, for a prescription to be both efficacious and acceptable it must be composed of a sequence of activities that supports the group in achieving their goal while accommodating the interests of the individual participants. For this purpose, we will develop a structured approach and supporting models to aid the collaboration engineer in designing an efficacious and acceptable process prescription.

First we will present a theoretical basis for the Collaboration Engineering design approach based on the (process) engineering literature. Next, we will derive the requirements to the design approach and modify the process from 6.1 to accommodate the specific requirements of the Collaboration Engineering approach. Third, we will look at the challenges facilitators encounter when designing collaboration processes. In this analysis we will also derive the requirements for the pattern language (presented in chapter 7) from a design perspective. The pattern language, consisting of thinkLets, will be discussed in the next chapter and will support all five quality dimensions of collaboration process prescription design, but particularly predictability and reusability. We will then further explore these design challenges and derive a set of guidelines and tools to support the design of collaboration process prescription. These will be integrated in the Collaboration Engineering design approach that we will present. Finally, we will discuss a case study that we used to evaluate the value of the design approach, supporting guidelines and tools.

Note that, as also explained in chapter 2, when we are talking about 'design' we mean the effort of designing a collaboration process prescription. When we talk about 'process prescription' we mean the artifacts (such as the script and cue cards) used by the practitioner to learn and execute a collaborative work practice, i.e. the deliverable for the design effort of the collaboration engineer. When we talk about 'steps' we mean the steps in the design approach (analysis, decomposition, etc). When we talk about 'activities' we mean the activities that comprise the collaboration process prescription, most of which will be thinkLets.

6.1 Theoretical basis: design and engineering

The verb 'to design' means to plan and fashion the form and structure of an object (Dictionary.com, 2007). Engineering is the application of a systematic, disciplined, quantifiable approach to structures, machines, products, systems, or processes (IEEE, 1990). Both 'structure an object', yet the difference between 'engineering' and 'designing' is the use of a systematic approach while structuring the object. In this paper, the object of design or engineering is collaboration; which can be seen as a process or system. System design or system engineering is an interdisciplinary approach and means to enable the realization of successful systems (INCOSE, 2007). Process design or process engineering is discussed in several disciplines, of which the most closely related discipline is business process (re)engineering. One of the founders of this domain, Thomas Davenport defines it as "the envisioning of new work strategies, the actual process design activity, and the implementation of the change in all its complex technological, human, and organizational dimensions"(Davenport, 1993).

Designing a collaboration process prescription is a creative task. Regardless of domain, the act of designing involves creating something new: a solution to a problem, a new functionality for a system, or perhaps a new work of art. Therefore the act of designing a collaboration process prescription is similar, in some ways to the more general concepts of creativity or problem solving. Several similar general approaches,, labeled variously as "design," "decision-making," "creativity," or "problem solving," are represented as a sequence of steps which include (Simon, 1960, Mitroff et al., 1974, Drucker, 1967, Brady, 1967, Ackoff, 1978, Couger, 1995, Checkland, 1981, Simon, 1973):

- Identification of the issue, where the problem or challenge is identified and the scope determined
- Analysis, in which the situation, context, different aspects and processes involved are rigorously examined and modeled or other ways captured and simplified to gain insight in the problem and to determine constraints to the process
- Finding (and evaluating) alternatives, where different solutions and ideas are derived through a creative process and where these solutions are further analyzed to enable precise comparison
- Choice, where based on some set of criteria the different solutions are compared and the one best solving the problem is identified
- Implementation, where the chosen solution is realized and embedded in its context

We will use this sequence of steps as a basis for the Collaboration Engineering design approach. We tailor the approach to make it useful for the design of a transferable collaboration processes prescriptions that can be instantiated across many different instances of a task. We will explore each of the steps of the design process and discuss their challenges and the need for refinement and support in these steps

6.2 Design in Collaboration Engineering

In order to find the challenges of designing a collaboration process prescription for Collaboration Engineering, we first need to identify the basic steps for the design approach. The steps that we identified as a basis for this process must be refined to fit the constraints and requirements of the Collaboration Engineering approach. We will discuss these refinements first.

Collaboration Engineering design focuses on high value recurring collaborative tasks. The assessment of whether a task is a Collaboration Engineering task is extensively discussed in Chapter 2 section 2.3.1. In chapter 5 we described the transfer, which is the first step in implementation, and we will not discuss deployment in this research. In this chapter we will focus on the analysis and design phases (see figure 2.3 in Chapter 2), assuming that the investment decision is positive. Therefore the identification of the issue will not be discussed as part of the design approach, and we will start with the analysis phase, in which stakeholders are interviewed to elicit the requirements and constraints to the process.

The choice for a sequence of thinkLets will be the key step in the design approach. Just based on the requirements and constraints derived from the analysis step, we cannot yet choose among the set of thinkLets. We first need to identify the sequence of activities that is required for the group to create its deliverable and accomplish its goals. Therefore we will add a sub-step in the design process named activity decomposition in which we will derive the process prescription blue-print.

One of the main challenges of the design of a collaboration process prescription is the choice of appropriate tools and techniques (Antunes et al., 1999, Dennis et al., 2001, Hayne, 1999, Wheeler and Valacich, 1996, Zigurs and Buckland, 1998, Nunamaker et al., 1997). Since the same tools can be used for different facilitation techniques, Collaboration Engineering research focuses on the use of thinkLets, which describe actions and capabilities required to execute an activity, rather than the use of tools alone (Kolfschoten et al., 2006a, Vreede et al., 2006a). Since thinkLets create a predictable pattern of collaboration, and capture best practices of expert facilitators, the choice of a thinkLet simplifies several steps of a common design approach. Rather than finding alternative solutions, evaluating them, and choosing the best solution, a set of thinkLets is available to choose among. In chapter 3 we suggested the use of design patterns or thinkLets to increase predictability, which will be further explained in chapter 7. We will thus assume that after a blueprint of the process is created, the next step will be to choose among thinkLets from the library.

The resulting design approach would thus contain the following steps: analysis to set requirements, determining the process steps and choosing different facilitation techniques to match the steps in the process. The resulting sequence of activities should be documented in a prescription to lay out the agenda for the collaborative process. We asked 39 expert facilitators (We defined ‘expert’ as facilitated more than 100 sessions) (see textbox 4.4 in chapter 4 for method) whether they executed these tasks in their design effort. The results are displayed in table 6.1

Workshop design activity		Experts (n=39)
1	Analysis of task/ problem	90%
2	Analysis of group and context	97%
3	Define tasks/steps	87%
4	Choose techniques	95%
5	Create agenda	100%

Table 6.1. Percentage of expert facilitators that performs the indicated design task.

We see that each of these steps is performed by expert facilitators. Since we need to offer a process prescription that is highly predictable we need to add one additional step. ThinkLets increase the predictability of the different facilitation interventions used, but their use does not ensure the predictability of the collaboration process prescription as a whole. For this purpose we need to validate and adjust the design as a whole. This is done before we start the documentation of the process prescription for transfer and enter the transfer phase. While this approach and the validation step will support the consideration of the quality dimensions we identified, transfer and its evaluation might still reveal new problems that require adjustments in the design. However, we think it is worthwhile to validate the design both before and after transfer because of the stakes in getting it right the first time.

Based on this we will set the following requirements for the design approach:

- To enable transfer, the design approach will result in a collaboration process prescription according to the template in chapter 5. The design approach should thus offer guidelines for the documentation of the process prescription.
- The design approach will have an extra sub-step to decompose the goal and deliverable into a sequence of activities that offers a blue print for the collaboration process prescription.
- The design approach is based on the use of thinkLets to increase reusability and predictability as explained in chapter 7 and will therefore not address the step "finding alternative solutions" but instead assume the choice for a design pattern (thinkLet) from the pattern language (thinkLet library).
- The design approach will have an additional validation step before the transfer phase starts to validate the effect of the sequence of activities rather than the effect of each activity alone.
- For reasons of efficiency, the full documentation of the design (creating the process prescription) is done after the validation step. However some aspects need to be captured to enable validation. We will name this step agenda building.

The Collaboration Engineering design approach will thus contain the following steps:

- Task diagnosis
- Activity decomposition
- Task thinkLet choice
- Agenda building

- Design validation
- Design documentation (for transfer)

It is important to note that the design process as described appears to represent a "waterfall" approach. However, as in software engineering, it is clear that these steps may not be executed sequentially, but may be iterative and incremental in nature (Boehm, 1981). Insights and choices in every step can affect past and future steps and choices (Lehman, 1989). For instance, choices of thinkLets affect the choices made in the decomposition and validation might lead to revision of the requirement, and thus changes in the sequence of activities. Furthermore, the documentation of the collaboration process prescription will not take place at the end of the design process but rather on a continuous basis during the design process; for instance the assumptions can be documented after the analysis. We will not discuss the process prescription documentation in this section as this was discussed extensively in chapter 5.

6.3 Increasing efficaciousness and acceptance: design support

With this approach as a blueprint for the design process we can identify further challenges in the design of a collaboration process prescription for Collaboration Engineering to identify guidelines and support that we can use to further detail our design approach.

6.3.1 Requirements to the design support

In this section we will derive requirements to the design support based on challenges in the design effort for facilitators, and requirements from a practitioner perspective. Some of these requirements can be implemented with design support; others pose requirements to the thinkLet conceptualization as presented in chapter 7.

Step 1: Analysis challenges
In the analysis step, the collaboration engineer should interview stakeholders in the organization(s) that will be involved in executing the collaborative task. This would include the project team responsible for the deployment of the collaborative task, and may also include some of the domain experts that will be trained as practitioners. Based on the process prescription template in chapter 5 we can offer a checklist of information elements that should be explored in the analysis phase to set requirements and constraints for the process. However, some of these requirements and constraints might vary in different instances of the task which might pose a challenge.

The Collaboration Engineering process prescription will be reused in several instances of the task. This requires that the process prescription can be instantiated for many different occurrences of the task. Looking at our quality dimensions for design (figure 3.7 in chapter 3) this can imply different groups and stakes, different topics, different resources and different practitioners. In order to understand the setting and the required flexibility, like in a problem solving process we start our design effort with an analysis step, in which the desired goal, requirements, constraints, and the deliverables are identified.

The requirements and constraints that must be taken into account can partly be derived from the descriptive model of a GSS session, described by Nunamaker (Nunamaker et al., 1991). The components of this model are the group, the task, the technology, the organizational context and the process and outcomes. The group, organizational context and task aspects will inform efficaciousness and acceptance, while the technology represents part of the resources and thus informs constraints to reusability.

Using a questionnaire we examined the information that expert facilitators use during the design of collaboration process prescriptions (see textbox 4.4 in chapter 4 for method). We verified the importance and availability of information aspects regarding the group, its context, and aspects of the task. The aspects considered are listed in table 6.2.

	Importance Mean	Expert n=39 Stdv	Availability Mean	Expert n=39 Stdv
task goal	6.5	1.1	5.8	1.5
task complexity	5.7	1.6	4.8	1.7
task deliverables	6.0	1.3	5.6	1.3
task size	5.2	1.6	4.9	1.7
task time frame	5.6	1.6	5.4	1.6
group size	5.8	1.3	5.9	1.3
group # stakeholders	5.8	1.3	5.1	1.5
group education level	4.0	1.8	4.1	1.8
group organization culture	5.2	1.4	4.1	1.7
group institutionalized methods	4.5	1.6	4.0	1.8

Table 6.2.Indication of the importance and availability of task and group aspects on a scale from 1-7. 1= very unimportant/available 7= very important/available.

If we look at the availability of information, some scores are lower than others, but most information is generally available. Thus this step does not pose a great challenge. It seems that contextual group aspects such as education level and institutionalized methods are less important and less available; therefore we think that the design is less sensitive to variations on contextual group aspects than on task aspects. However both should be considered, and a scope of possible variations should be determined and accommodated as design requirements. Task analysis seems more complex, the goal, complexity and deliverables are important information for the design and thus these should be similar for the recurring task. Also the timeframe available for the task is an important constraint to the collaboration process design, and interviews with practitioners (see chapter 4 textbox 4.2 for method) indicated that a large change in the available timeframe resulted in considerable problems for the practitioner, as they were not able to finish the sequence of activities within the timeframe, or were not able to achieve the required quality of results due to lack of time for rigor in the various activities. A change of the timeframe available for the task could therefore reduce reusability and predictability of the process prescription. The task size is considered less important but still important. A large difference in task size may affect the time required to process information and thus can pose problems.

Besides the factors discussed in the survey we found some additional challenges in the interviews with practitioners (see chapter 4 textbox 4.2 for method). Resources such as a room, whiteboards and materials will not usually pose significant problems. Although lightning, room temperature, use of colors etc. can affect productivity, these factors will not be prescribed in this approach. We will however take into account the use of technology. Technology such as GSS but also more simple tools as MS EXEL or MS Messenger can offer data processing capacity and can offer specific features such as increased anonymity, faster parallel work, electronic minutes and data structuring features (Bostrom and Anson, 1992). Some of these functionalities can also be offered with pen and paper tools, however, especially data processing capacity and electronic minutes are more difficult to implement without information technology.

A last source of design requirements and constraints is formed by the practitioners themselves. Depending on the organizational setting it might be possible to select practitioners. Practitioners should first of all be domain experts but some other aspects might also help in selecting practitioners. We asked facilitators the key skills, personality characteristics and knowledge they required for their role (see chapter 4 textbox 4.1 for method). Results are listed in table 6.3

Skills	**#**	***Personality***	**#**	***Knowledge***	**#**
group dynamics	14	in control	7	groups	8
analytical	9	receptive	6	techniques & methods	6
result minded/goal focused	9	self-aware disciplined	6	terminology, domain	5
listening skills	8	analytical	5	process management	3
leadership skills	7	focused	5	GSS	3
social skills	6	social	4	psychology	2
process management skills	5	ego-less	4	general background	2
master facilitation techniques	5	open & flexible	4	presence	1
communication skills	4	nice humoristic	3	communication	1
create atmosphere	4	professional	3	culture	1
impartial	4	integer	3		
acceptable/likable	3	understand hierarchy	2		
presentation skills	3	understand techniques	2		
patient	2	communicative	2		
knowledgeable on topic	2	positive	2		
self confident	1	tidy, neat	2		
inspire	1	patient	1		
be critical	1	creative	1		
enable learning	1				

Table 6.3. Skills, personality and knowledge required for group support.

The most important skills for a practitioner to have are group dynamic skills, analytical skills, result-mindedness or goal focus, listening skills, leadership skills and social skills. Several personality aspects might support these skills such as being analytical, focused, receptive, in control, and social. Knowledge required besides domain knowledge would be knowledge about techniques, methods and technology, but for practitioners this knowledge captured in the process prescription to the extent they need it. Knowledge about groups was also frequently mentioned. While most of this knowledge is required for the design of collaboration processes and thus not for the practitioner, practitioners with experience in supporting groups such as classroom or management experience might have some advantage.

Concluding, requirements to support the analysis phase are the following: To support this design step we will need a checklist for the interviews and offer guidelines on how to negotiate on fixed requirements and constraints and how to deal with flexible requirements in the design. The analysis will also set some requirements to the selection of thinkLets. Several characteristics of the group, task, resources and practitioners limit the choice of thinkLets. Therefore it is useful to be able to indicate for each thinkLet which constraints to its application are fixed and which are flexible.

Step 2: Activity decomposition challenges

A collaboration process prescription describes a sequence of activities that leads to accomplishment of a task through group effort. To define this sequence of activities the collaboration engineer decomposes the task into a sequence of activities. This decomposition of the process is a critical and difficult step.

We did a detailed analysis of the design efforts of facilitators using interviews. We asked facilitators to reason out loud how they would support a group in creating input for a long-term strategy. This interview was used to understand the design task. (See chapter 4 textbox 4.5 for the method). All facilitators asked the interviewer questions to gain more information about the task and the group and then at some point created a first draft of a sequence of activities that would lead the group to the goal they had in mind. There were two key approaches to determine this first sequence; either they found a match in a standard approach or they decomposed the final deliverable into smaller sub deliverables and identified the activities to create these deliverables.

In the first approach, some characteristics of the deliverable matched or resembled a known approach. Some standard approaches that the collaboration engineers mentioned only created part of the deliverable defined in the group goal, and thus the selected standard approach was used as a basis, and the decomposition approach was used to further elicit the sequence of activities.

Facilitators that used the decomposition approach and did not match the task to a known standard approach, seemed to struggle more on the task; sometimes facilitators wanted to start over again, and often they went back a few steps and altered them. Many facilitators first wanted to know the details of the deliverable and the goal and then wanted to know what exactly the starting point or status quo in the case was. The facilitators then identified activities to create each of the requirements to the deliverable, reasoning back to the starting point, or reasoning from the starting point in either a sort of “trial-and-error” fashion in which the output of the first activity was compared to the deliverable, and an activity was found to resolve the difference, or by identifying high level patterns that are required to move the group from generating input through different modifications of this input towards the deliverable.

The logic of the sequence of activities is very important. The sequence of activities should produce intermediate deliverables that contribute to the group goal and that can serve as input for the next activity or that can meaningfully alter the input from the previous activity. This makes the decomposition a difficult challenge that is interconnected with the choice of facilitation techniques. The choice of one technique limits the possibilities for the next technique and sets requirements to the output of the previous technique. Furthermore, the decomposition can result in activities with different complexity levels. For instance by summarizing the results from a brainstorm the group can reduce and clarify the results in one activity. An alternative approach would be to first vote on the most important items, and then discuss the ranking. This will separate the reduction and the clarification activity, resulting in lower cognitive load of the activities both for the group and for the practitioner, but it will also affect the level of commitment and the level of shared understanding with respect to the results.

To support the decomposition of a collaboration process into activities that will be easy to match with thinkLets we need to offer collaboration engineers a set of classifications to identify outcomes and high level patterns of collaboration that can be linked to the process activities and sub deliverables. Such classification will help to create a sequence of activities. However, this phase will be highly iterative with the choice phase.

When we offer these classifications of high level patterns and outcomes, we also need to classify the thinkLets accordingly. This will offer a first choice method.

Step 3: Choice challenges

If we consider all facilitators (novices, experienced and expert facilitators) that filled out the questionnaire (see textbox 4.4 in chapter 4 for method), 77% (n70) develop their own facilitation techniques and almost all, 97% (n71), adjusted their techniques to the specific situation. Practitioners do not have the experience and skills to adjust or create facilitation techniques. Furthermore, they will not have the knowledge and experience to choose facilitation techniques. On the other hand, facilitators rely on this skill. Only 7% (n71) of the facilitators indicates that they do ***not*** choose facilitation techniques during the session, but only during the preparation. 93% (n71) chooses facilitation techniques both during the preparation and the session. Since facilitators heavily rely on this skill, we asked them how they choose among facilitation techniques. A few typical example choice approaches they mentioned are listed below:

- "I consider the group, task timeframe and goal and choose"
- "Based on experience or intuition"
- "I use the techniques I know"
- "In discussion with the client, I propose, revise, propose etc."
- "I use books as guidelines"
- "I use what worked before"
- "I use a standard and adjust it"
- "I consider alternatives"

While these approaches occurred frequently, some of them are not transferable and should be made explicit. We will therefore further explore some of these approaches to identify challenges and offer support methods.

As indicated above, a first approach to the choice among thinkLets is to use classifications of results or patterns of collaboration. This will help to establish the fit between task and thinkLet. Some of the facilitators we interviewed classified the results of the sub activities and based on that classification they choose a thinkLet that created that outcome or pattern. For this purpose the thinkLets should be classified to the patterns and outcomes they create. A challenge in this approach is that some thinkLets can be used to create several results or several outcomes at once. To give an example; a thinkLet that can be used to generate ideas can also be used to perform a qualitative evaluation, and a technique to select ideas can also be used as a way to rank results. To resolve this problem we have two options; we can classify the thinkLets among multiple outcomes and patterns or we need to ascribe the effects (outcomes and patterns) to the different interventions within a thinkLet, and enable collaboration engineers to design thinkLets for a specific activity. This challenge will be further addressed in the conceptualization of the pattern language in chapter 7. When designing collaboration process prescriptions to transfer them to practitioners, existing, known thinkLets will have preference over new, designed thinkLets, since new thinkLets are less predictable than tried and tested thinkLets.

Some of the facilitators we interviewed only had a very small library of techniques and simply scanned each to see whether it fit. In this case they often ruled out several techniques right away and then compared the effect of the remaining techniques to make a choice. A choice among a few techniques is of course simpler than a choice among many techniques, but ‘fit’ between task and technique is less likely with a limited set.

Many combinations of thinkLets or facilitation techniques are simply not possible; the input required for one thinkLet is not created by the previous thinkLet. To support the choice among a larger set of thinkLets, we could offer a choice map: A matrix that indicates whether a combination of thinkLets is possible when the output of the first thinkLet is used as input for the next thinkLet. Once a first thinkLet is chosen, it will be much easier to select the next thinkLet when only the possible combinations are considered. Another way to work with the choice map would be to select a thinkLet to create the final result, and work back to select thinkLets that create the input required for the selected thinkLet.

Sometimes facilitators choose a known combination or sequence of thinkLets. We did a search on high level patterns of thinkLet combinations used by facilitators (see textbox 4.6 in chapter 4 for the method used). The pattern language should offer not only a set of thinkLets but also an overview of sequences of thinkLets that can be used. The resulting patterns and the number of times they recurred are presented in table 6.4. (Note that LeafHopper 1 level means that people contributed based on an existing list of ideas that was created in a previous step or that was created in preparation for the session, while leafhopper level 2 means that participants added ideas in different topics and also commented or elaborated on each other’s ideas).

#	1st ThinkLet	2nd ThinkLet	3rd ThinkLet	4th ThinkLet
20	LeafHopper 1 level	CheckMark		
15	LeafHopper 1 level	StrawPoll		
6	LeafHopper 2 levels	BucketWalk		
10	LeafHopper 2 levels	CheckMark		
10	LeafHopper 2 levels	StrawPoll		
7	LeafHopper 1 level	BucketWalk		
5	Leafhopper 1 level	BucketWalk	Checkmark	LeafHopper 1 level
32	PlusMinusInteresting	FastFocus		
8	PlusMinusInteresting	FastFocus	StrawPoll	CheckMark
5	PlusMinusInteresting	LeafHopper 1 level		
32	OnePage	LeafHopper 1 level		
10	OnePage	LeafHopper 1 level	CheckMark	
5	OnePage	LeafHopper 1 level	StrawPoll	
10	OnePage	PopcornSort		

Table 6.4. Patterns in thinkLet use.

Each combination has a distinct effect that supports a specific type of group process and creates a specific deliverable. From experience and from this analysis we know that some combinations of thinkLets work better than others. We can use this information to suggest preferred combinations in our choice map. For this purpose thinkLets should therefore be documented with:

- An indication of the possible combinations and their effect
- A description of alternatives and when to use them
- An indication of required input
- A classification according to output and pattern of collaboration
- If a thinkLet has multiple effects, the effect of each intervention and variations on interventions should be explained.

In a few cases, the facilitators that were interviewed did not make a choice but created a flexible activity. They kept 2 options and mentioned the criterion on which they would choose during the session. A flexible activity might be difficult for a practitioner as it increases the amount of thinkLets that need to be mastered and it increases cognitive load as more choices have to be made during execution. Yet, it also enables a collaboration engineer to create a process prescription that is more flexible towards the different instances of the recurring task, especially if the outcome can vary, or when there is a variation in the available resources such as time and technology. In such cases the collaboration engineer can build in an activity that is flexible towards such constraints. The choice for a flexible activity is thus a tradeoff between the need for flexibility and the cognitive load imposed on the practitioner, and can be used to increase reusability of the process prescription or to fit a variation in outcome. For this purpose an overview of sequences and an overview of alternative thinkLets will also help. Furthermore, the requirements to the physical workspace should be documented for each thinkLet as a set of capabilities that can be instantiated in different instances of the task.

Questionnaires among 80 facilitators (see textbox 4.4 in chapter 4 for method) indicated that 78% has a set of facilitation techniques that they regularly use.

Although facilitators have often access to databases with facilitation techniques such as (Jenkins, 2005, Briggs and Vreede, 2001), they tend to fall back on their favorite facilitation techniques. A key explanation for this is that a technique becomes more predictable when a facilitator has tried it in more settings and when the facilitator gains experience with the technique and thus better masters it. Collaboration engineers are expected to be master facilitators who can choose techniques based on extensive experience. However, this experience will also help them to master the thinkLet better. Collaboration engineers might thus underestimate the challenges of a thinkLet for novices. Therefore, challenges of a thinkLet as experienced by novices, offer a valuable source of information not only for practitioners, but also for collaboration engineers who can use this information to assess the fit with the practitioner skills. For each thinkLet we should therefore document:

- A detailed predictable effect of the component
- Challenges that can occur when a novice uses the thinkLet

A last fit is a fit with stakeholders. The number of different stakeholders often is an indicator to the likelihood of conflict and the need to consider different perspectives and to create consensus. Some thinkLets are particularly designed for these purposes; others can be modified to support such activities. Another aspect that can be considered to optimize the fit between stakeholders and the process is to enable stakeholder to contribute from their own perspective. If stakeholders cannot share their views and concerns with respect to the topic, they are less likely to accept the process. This ability is mostly offered or constrained by the specification of the questions, assignment and parameters used for each thinkLet. A process without generation steps with limited constraints and without discussion during the other collaborative steps is more likely to cause resistance. A last consideration in the choice among thinkLets to fit to stakeholders is the cognitive load of the task. The more complex the activity and the thinkLet, the more effort will be required from the participants. We learned from our theory on collaboration (Chapter 3 section 3.3) that when stakes are low, participants will make less effort and commit less other resources such as knowledge and time. Thus depending on the stakes and of course the cognitive abilities of the participants, a more or less complex process can be designed. To support this fit we can indicate some information for each thinkLet such as;

- A time frame indicator
- A complexity index
- A discussion index
- A description of the added value and advantages of the thinkLet

Based on these choice guidelines to optimize the fit to the task, stakes, resources and practitioners we can distill the following requirements to the design pattern conceptualization and pattern language (thinkLet concept):

Task fit:

- An indication of the possible combinations and their effect
- A description of alternatives and when to use them
- An indication of required input
- A classification according to result and pattern of collaboration
- If a thinkLet has multiple effects, the effect of each intervention and variations on interventions should be explained
- An overview of sequences of thinkLets that can be used

Resource fit:

- The requirements to the physical workspace for each thinkLet as a set of capabilities
- An indication of fixed and flexible constraints

Stakeholder fit:

- A time frame indicator
- A complexity index
- A discussion index
- A description of the added value and advantages of the thinkLet

Practitioner fit:

- A detailed predictable effect of the component
- Challenges that can occur when a novice uses the thinkLet

Step 4: Agenda documentation challenges

When a first process sequence is decomposed and thinkLet choices are made, the agenda can be built. In the agenda, not only the thinkLets and the time, or range of time with guidelines for planning for each thinkLet should be documented, but also the additional process activities such as introduction and wrap up, presentations, breaks and other activities should be planned. Furthermore, the questions, assignments and parameters for each activity should be formulated where possible, or instructions for the practitioner to formulate these should be made. Posing the right questions and assignments has a large impact on the cognitive load of the process for the participants, and affects the use of effort and knowledge. In the design approach we therefore need to offer guidelines for the formulation of questions and assignments and we need to support the documentation of the agenda.

Step 5: Validation challenges

After the thinkLets are chosen and arranged in a sequence of activities, the process can be validated. There are different approaches to validation that can be used, but it is important to offer a checklist to evaluate each of the quality criteria described in chapter 3. Based on the choice insights above we can offer this checklist:

Efficaciousness (task fit)

- The intended deliverables of each activity contribute to achieving the group goal
- Execution of the design is likely to produce the required deliverables
- Activities are combined in a logic sequence, where output of one activity fits a input of the next

A high quality process prescription describes a process that is likely to produce the results that are required to reach the group goal. A group goal is not necessarily an explicit group result such as a decision or strategy; it can also be a more tacit result such as awareness or mutual understanding. This means that one needs to specify the goal in high detail, including the deliverables required and the quality dimensions related to those deliverables such as precision, completeness, etc. Furthermore, it is important to be able to argue how each step in the collaboration process contributes in creating those deliverables and achieving the goal. It is important that each activity is necessary and important to achieve the goal; if there are activities that do not directly contribute to the goal, either the goal is not sufficiently specific or the activity is redundant. The structure of the process prescription is also important for its effectiveness. Each activity should lead to the goal, and the activities should also be combined in a logic sequence in which the output of one activity is suitable as input for a next activity.

Reusability (resource fit)

- The time allocated for the process is used optimally and the timeframe or planning guidelines are feasible
- Other resources such as GSS and other available tools are used optimally
- The process prescription has enough variation to keep the participants attention
- The process motivates the participants to make the required effort and commit other resources to the process
- Questions and assignments are clear and unambiguous and pose low cognitive load
- There are enough breaks
- The knowledge available in the group is used optimally; it is ensured that participants understand and consider each others input

Resources in a collaboration process are effort, time, knowledge and physical resources. Effort is a special resource and is available in relation to the acceptability of the design, as addressed below. Resources are almost always limited. Sometimes it might be possible to negotiate about the resources available, and some resources are fixed while others will vary over the different instance of the task. Therefore, the use of resources should be optimized in order to create a process that fits the resources available and uses them efficiently. One of the resources that must fit is the timeframe. It is important that the timeframe of the meeting is feasible. Furthermore, the allocated resources such as GSS and other collaboration support tools should be used optimally when available. When available technology and tools can increase the efficiency of the process they should be applied.

Effort and knowledge are critical and scarce resources in a collaboration process. While time and other resources often can be allocated to the process by the project manager, effort and particularly cognitive effort is not as easily allocated and coordinated as the use of other resources. Effort and knowledge are committed by participants willingly based on individual stakes and stimulating an increase in effort and knowledge sharing requires motivation and accommodation of stakes. For this purpose the process prescription should have enough variation to keep attention and focus, and it should motivate participants to make the required effort. To manage the physical effort of the process there should be enough breaks, and breaks should be planned strategically. Furthermore, it is important that the questions are focused and clear so the cognitive effort of the task is as low as possible and no effort is wasted on understanding the task rather than performing it. Last, to create high quality results, the stakeholders should be encouraged to share their relevant knowledge and shared understanding should be facilitated.

Acceptance (stakeholder fit)

- The stakeholders in the session will be able to have their say; there is room for concerns, different opinions and perceptions
- The group goal is likely to be supported by all stakeholders
- There is time to identify, address and resolve critical conflicts
- The design enables verification of the support for the outcome
- The design fits the group's characteristics

Participation of stakeholders is a commitment of time and thus a first indication of acceptance. However, often goals are interpreted differently or participation is driven by individual goals that are not necessarily related and can even be conflicting with the group goal, and thus the acceptance of the process can change during the process. To collaborate effectively and efficiently, the group members need to commit to the group goal, and conflicting individual goals need to be resolved. A second indicator of acceptance is whether the outcome of the session is likely to be supported by the participants. Acceptance of the results is required in order to consider them a valuable group result. It is unlikely that stakeholders will accept a group result when they do not get the opportunity to contribute to it and to criticize or question it. When this gives rise to different perspectives and concerns, there should be time to discuss this and to resolve the critical conflicts that impair acceptance of the results. Groups can be very different, depending on characteristics such as history, culture, education level and size. These factors affect the available resources (effort, skills and knowledge), and they might affect the amount and variety of stakes. Furthermore, the fit with the task can be positively influenced, for instance through the use of institutionalized methods.

Transferability (Practitioner fit)

- The practitioner that will execute the design is capable to do so

The last aspect that influences the quality of design is simple but important. A very complex and sophisticated process prescription might be efficacious and acceptable, but if the practitioner does not have the skills or competences to execute it, it will not support a successful collaboration process, therefore the process prescription should fit the practitioners' capabilities. As we discussed in chapter 5, any reduction in cognitive load for the practitioner that does not reduce efficaciousness and acceptability should be applied. The remaining intrinsic complexity of the process prescription can only be reduced at the cost of these qualities. In some cases this might still lead to a higher quality collaboration process. A try-out by practitioners might reveal insights in this choice.

Predictability

- The process design has as little uncertainty as possible, and remaining uncertainty is addressed during the training.

Based on this validation, the analysis of requirements and constraints and the selected thinkLets a collaboration engineer can assess the amount of uncertainty that is left in the collaboration process prescription. Some uncertainties can be resolved with the use of flexible steps (see above), others might be resolved with training, and yet others can be resolved with additional validation through testing or a pilot. It is an illusion to create a fully predictable process prescription, but with this design approach and the use of thinkLets and additional validation methods, a process prescription can be made that is as predictable as possible.

These validation aspects can be considered using a variety of different validation methods ranging from re-reading the process prescription, through a validation with colleagues, to a role-played walk through, to a full try out. The different methods will be explained in the design approach. Depending on the amount of uncertainty left and the size of the project more precise validation methods can be used.

6.3.2 Conceptual design: design support for Collaboration Engineering

To design a collaboration process prescription that meets the quality criteria listed above we will now describe the approach to each of the steps, the supporting tools and guidelines and the outcomes. An example of the resulting process prescription can be found in appendix 6

Figure 6.1 gives an overview of the design process. The central blocks represent the different steps in the design effort. Left the external input for each step is listed. Between the steps, results that serve as input for the next step are described. Design documentation, displayed in the background is a continuous activity that is done during the other phases but addressed separately below. The black arrows indicate the iterative nature of the design effort

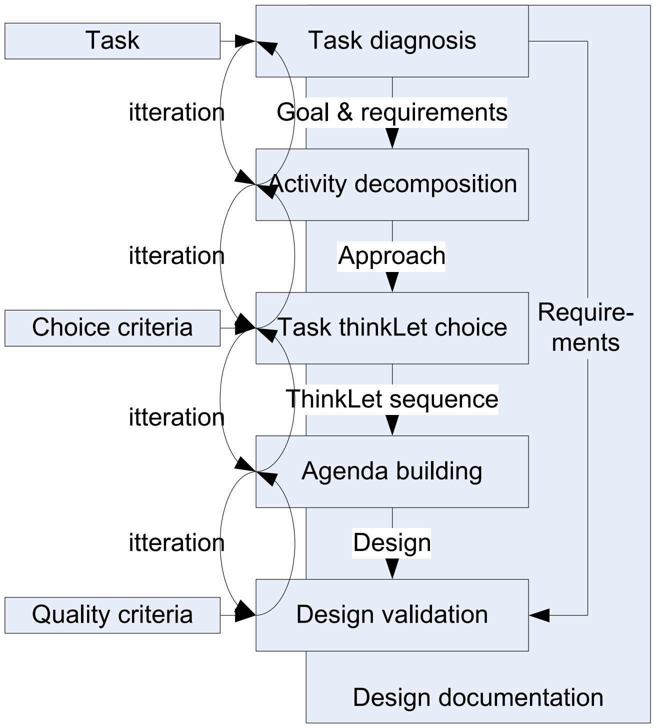


Figure 6.1. Approach to create a collaboration process prescription or Collaboration Engineering.

Step 1: Task diagnosis

Design is an evolving process that starts with the initial conversation with stakeholders involved in the Collaboration Engineering process in which the collaboration engineer determines, adjusts and negotiates about the requirements and constraints on the collaboration process with respect to the task, the stakeholders involved the resources available and the practitioners. The following information should be analyzed:

- **Task analysis:** Goal, deliverables, and objectives

The goal, deliverables and objectives of the session are the cornerstone of the design. If the process creates the wrong deliverables or achieves a different goal or has a different objective than the stakeholders had in mind, the session will not be successful. The practitioner will have to gain commitment from the group with respect to the goal, objectives and deliverables that are determined by the collaboration engineer, and thus these should be established very carefully.

The goal and deliverable are often a solution, decision, or analysis. However, the collaboration process can also have “experience goals”. An example of an experience goal is a collaboration process that has the objective to create awareness of a problem among the participants. Goal setting theory describes that a goal should be specific and challenging enough for the participants in order to evoke productivity (Locke and Latham, 1990). Specificity is also important for the practitioner in order to assess the progress made by the group in achieving the goal and creating the deliverables. If the goal and task are very straightforward, then defining the deliverables is simple. But if the goal is to create awareness for instance, the deliverables are less clear. A deliverable is the tangible output of the process, for instance a detailed solution, a ranking of preferences, a list of options, etc. Also important is to establish what will be done with the deliverables and how they will be used; will the participants get them, should they be worked out in more detail, will the decision be implemented, etc. Depending on how a result will be used, some characteristics of the deliverable can be established. E.g. a proposal that needs to be judged by the management of a large organization should not be a 100-page document, but a concise management summary, while an evaluation report might require more length and detail. A good way to get an understanding of the deliverables is to ask the stakeholders to give an example of a deliverable. Additionally, the collaboration engineer can ask questions to further specify the deliverable. Below is a list of adjectives that could be used to indicate the characteristics of a deliverable that we derived from analysis of the designs made by facilitators based on the case description (see textbox 4.5 in chapter 4 for method):

Amount of concepts	Consistency	Level of understanding
Balanced	Creativeness	Representative
Breadth	Deliberateness	Richness
Certainty	Depth	Scrutiny
Clarity	Feasibility	Shared/Joint
Coherence	Importance	Supported
Completeness	Level of abstraction	Variety
Consent/Agreement	Level of detail	Preciseness

These adjectives can be used to specify the deliverable, but also as a basis for interviewing the client to elicit more specific characteristics of the deliverable.

- **Stakeholder analysis:** group, stakes, roles and needs

Many collaboration processes can be used in a large variety of different groups. However, aspects as group size, age, sex, culture, educational background, and organization level, can help to customize the process prescription, for instance by adjusting the tone of the script (formal/ informal) or to help creating sub-groups. A separate issue is the amount of stakeholders involved; it will be difficult to interpret the signals from the group and to manage discussions, misunderstanding and conflict if it comes as a surprise. A good approach to the stakeholder analysis is to consider the team history, and for each stakeholder involved:

- Their individual stakes in the process and results
- Their reason for participation, what do they expect
- What they can contribute
- Whether they will commit to the group goal
- Whether they will accept the results
- Whether they will accept the process
- Whether they will commit the required resources
- Whether they will be motivated to make effort and share knowledge

A process design should accommodate stakes as much as possible to increase commitment of resources (knowledge and effort) and acceptance of the process and results. It is possible to give different stakeholders different roles in the process. Roles in the process can be incorporated in the thinkLets, but a problem owner for instance can follow the instructions for all participants, but might have a separate role during the introduction. These roles should be documented.

- **Resource analysis:** time, knowledge, effort and physical resources

The design must be adjusted to make optimal use of the available resources. In discussion with the organization, a time frame, resources, technology and budget need to be determined. The timeframe of a collaboration process design is not very flexible, but it might be possible to create a "light version" of the process in which some steps are removed to fit a shorter time-frame. In some cases the time for the process will vary, in that case the process prescription should specify a range for the timeframe and the time for each activity and it should specify guidelines for the planning of the process. The resources available are often more dynamic and in any case technology is not fully reliable (think of a power-black-out). Depending on the reliable availability of resources a process can be designed in which specific resources are indicated or in which requirements to the resources are specified as capabilities and practitioners are instructed on how they can implement the design with different resources. Knowledge and effort is committed to the process by the participants. To make sure these resources are availed to the process, the individual stakes of the participants should be accommodated.

- **Practitioner analysis:** skills, experience, personality, domain expertise

The practitioner analysis can be done based on two scenarios; the selection of practitioners is already determined and the design should be adjusted to the practitioner or the practitioner profile, or the collaboration engineer is asked to create a profile that is used to select practitioners. A key requirement to the practitioners is that they get time allocated for the group support task. Furthermore, they should be, or become experts on the domain of which the collaboration process is part. When practitioners have some experience in working with groups such as being a teacher or manager, the role of practitioner might be easier for them, but such experience is not required. Furthermore, some social skills, listening skills and analytical/focus skills would be useful. Also, leadership or authority as a personality characteristic or as experience could help a person to feel more comfortable in the role of group leader. No knowledge other than domain knowledge is required in advance, but if group support technology is used, some affinity with technology is useful.

These factors can be used as a checklist to analyze or negotiate the requirements and constraints to the collaboration process design. In this analysis it is important to determine whether constraints and requirements are fixed or dynamic. For example, in an evaluation process, practitioners might have to specify the concepts for evaluation, and therefore the number of concepts and their level of detail might vary, posing a dynamic requirement. A collaboration engineer could in such case establish a set of criteria for the concepts, for instance, there should be at least 3 and no more than 10 concepts, they should be described according to several guidelines and if there are more than 5 concepts, the process time will be 30 minutes longer. Once the analysis is complete and all requirements and constraints are set, this analysis can be used to fill in the assumptions document (see section 5.3.2 in chapter 5)

Step 2: Activity decomposition

When the goal and requirements are clear, the basic process needs to be determined. To do this we need to further analyze and decompose the task into activities. A first step is to determine if the organization has already a pre-defined way of executing the task. If the traditional practice is functional and results can be improved by making it collaborative then it can be used as a starting point. If no process is followed in the organization, then standards in the literature might provide a starting point for the activity decomposition. If the process is first of its kind, then a new process for the task should be defined.

The following steps are required to define a process from scratch.
-Elicit deliverables of the task
A task always has a deliverable, and a deliverable always serves a goal. There might be requirements to the deliverable that describe how it is captured and used in a next phase. The goal of the process might also set requirements to the deliverable, such as for instance the level of detail or consensus about an outcome.
-Define activities for the deliverables
Each deliverable requires a number of collaborative activities to be achieved. These should be defined as an activity in the collaboration process.
-Name and sequence the activities in the process
Finally, once the activities are defined, they must be named and sequenced, e.g. the output of activity 1 can be the input for activity 2.

With the resulting rudimental process and deliverable description, we can further decompose the process in smaller steps. For this purpose we offer two approaches; process decomposition and result decomposition. Both can be used in combination, but will be explained separately.

Process decomposition
In process decomposition the patterns of collaboration are used. Patterns of collaboration characterize a group activity as the members move from an initial state to a next state (Vreede and Briggs, 2005). Each pattern has a number of sub patterns that can be matched to activities in the generic process description. The patterns of collaboration are (Briggs et al., 2005, Briggs et al., 2006b, Kolfschoten et al., working paper):

Generate

- Creativity
  Move from having fewer to having more new concepts in the pool of concepts shared by the group
- Gathering
  Move from having fewer to having more complete and relevant information shared by the group
- Reflecting (idem evaluate)
  More from less to more understanding of the relative value or quality of a property or characteristic of a concept shared by the group

Reduce

- Filtering

Move from having many concepts to fewer concepts that meet a specific criteria according to the group members

- Summarizing
  Move from having many concepts to having a focus on fewer concepts that represent the knowledge shared by group members
- Abstracting
  Move from having many detailed concepts to fewer concepts that reduce complexity of the overall concept

Clarify

- Sense making
  Move from having less to having more shared meaning of context, and possible actions in order to support principled, informed action
- Building shared understanding
  Move from having less to more shared understanding of the concepts shared by the group and the words and phrases used to express them.

Organize

- Categorizing
  Move from less to more understanding of the categorical relationships among concepts the group is considering
- Sequencing
  Move from less to more understanding of the sequential relationships among concepts the group is considering
- Causal decomposition
  Move from less to more understanding of the causal relationships among concepts the group is considering

Evaluate

- Choice: social/rational
  Move from less to more understanding of the concept(s) most preferred by the group
- Communication of preference
  Move from less to more understanding of the perspective of participants with respect to the preference of concepts the group is considering
- Reflecting (idem generate)
  More from less to more understanding of the relative value or quality of a property or characteristic of a concept shared by the group

Consensus Building

- Building agreement
  Move from less to more understanding of the difference in preference among participants with respect to concepts the group is considering
- Building commitment
  Move from less to more understanding of the willingness to commit of participants with respect to proposals the group is considering

Result decomposition
Decomposition based on results is based on a further analysis of the deliverables and requirements to come up with the elementary activities to create the results. Decomposition should lead to a level of activities where deliverables of each activity cannot be decomposed any more. Decomposition depends on the requirements defined in the first phase such as:

- **Time**: If little time is available for the task a choice might be to use activities with less detail and less discussion.
- **Project embedding:** It might be possible to assign participants to do preparation tasks before and "homework" after the collaboration session.
- **Cognitive load**: depending on the cognitive capacities of the group members, further decomposition of activities might be required to reduce task complexity.
- **Technology**: A GSS allows for more efficient data processing than a manually supported process
- **Practitioner skills**: A skilled practitioner can handle more complex activities
- **Task requirements**: The detailed requirements to the deliverable help to decompose based on results.

The following classification of the outcomes of collaboration can be used (table 6.5):

input	*structure*	*focus*	*shared understanding*	*commitment*	*empathy*
creative	clusters	selection	shared knowledge	decision	respect
informative	ranking	summary	shared meaning	support	shared stakes
visionary	model	scope	mutual learning	agreement	consideration
reflective	sequence	direction	mutual differences	consensus	team bond

Table 6.5. Result classification.

Input

- There are four types of input that we could distinguish; *creative* input such as ideas and solutions, *informative* input such as facts and experiences, *visionary* input such are future requirements, visions, scenario's and trends and *reflective* input such as comments, preferences and opinions. Each of these input types can be specified further with a number of quality criteria such as the level of detail, amount, precision, uniqueness etc. Besides generating this input it is important that people share the input and modify it to create a group result. This can be done trough structuring and focusing or trough the creation of shared understanding. Input can be textual but it can also be numeric or video or audio based.

Structure

- Once there is a collection of input elements, the individual input elements can be structured. We distinguish several types of structure: a *cluster* of related concepts, a *ranking* of concepts based on some criterion, a *model* in which more complex relations can be indicated and a *sequence* in which the timely relationship of concepts is indicated. Structure quality factors can be, for instance, consistence, abstraction level, complexity, and completeness.

Focus

- When a group needs to work towards a decision or a small set of alternatives, it needs to focus the contributions by integrating them and selecting among them. Results in this category include a *selection* where only a few concepts are chosen by the group, a *summary* in which concepts with similar meaning are integrated without removing unique input, a *scope* in which the boundaries for a collection of constructs are formulated, and a *direction* in which concepts that fit a specific cause of action are taken into account. Quality factors for focus include the amount of reduction, integration, and parsimoniousness of results.

Shared understanding

- Generating input is in essence a one-directional effort. Once input it created and shared, a second requirement for useful integration towards a group result is that the input is received and understood. Creating shared understanding is a key result in a collaboration process, and may be difficult. Yet, if the content of the process is simple or if the group already has a shared language and shared understanding this activity might not be necessary. We distinguish several types of shared understanding. First, *shared knowledge*, followed by, *shared meaning* about the knowledge in the group. Next, is *mutual learning*: people might learn from each other and advance both their own knowledge and the group knowledge. Last, *mutual differences* and disagreements can be revealed to gain understanding on different types of conflicts. The level of shared understanding or its quality is difficult to determine and improve as people can be unaware of differences in meaning based on asymmetry of information, assumptions and perceptions.

Commitment

- Once a result as discussed above is identified or created by the group, participants and stakeholders in the group will have different levels of commitment towards this result. Groups might want to achieve a specific amount of commitment from the critical stakeholders to be able to align future efforts and goals. One type of commitment is a *decision*, which can be made based on majority or on more sophisticated and inclusive decision making rules. Another option is to simply get *support* for a plan or proposal. Yet another type of commitment is an *agreement*, for instance to spend an amount of resources. A last type of commitment is a *consensus*, in which all critical stakeholders commit to the proposal (Briggs et al., 2006a).

Empathy

- A last outcome of interest to facilitators is empathy. This entails creating mutual understanding of stakes in the process and stimulating group members to take mutual stakes into consideration in order to align effort and goals. Results that we classify as empathy are: *respect* for other stakeholders, *shared stakes* when people accommodate the stakes of others among their own, *consideration* – taking those stakes into account, and a *team bond* in which mutual goals are pursued. Often some level of empathy is required to achieve commitment and shared understanding.

ThinkLet choice

After the decomposition the activities can be matched with thinkLets. This match is again made based on criteria and guidelines. The choice of a thinkLet is a complex task. Many factors influence the fit of the thinkLet to each of the dimensions. Taking al these aspects into account and comparing them to the characteristics of the thinkLets is difficult. There are several things that reduce the complexity of this choice: the memorization of thinkLets, using the categorizations, considering the previous and next thinkLet, reusing known combinations of thinkLets, and using the thinkLet documentation on the effect and contribution of the thinkLet, and its input requirements.

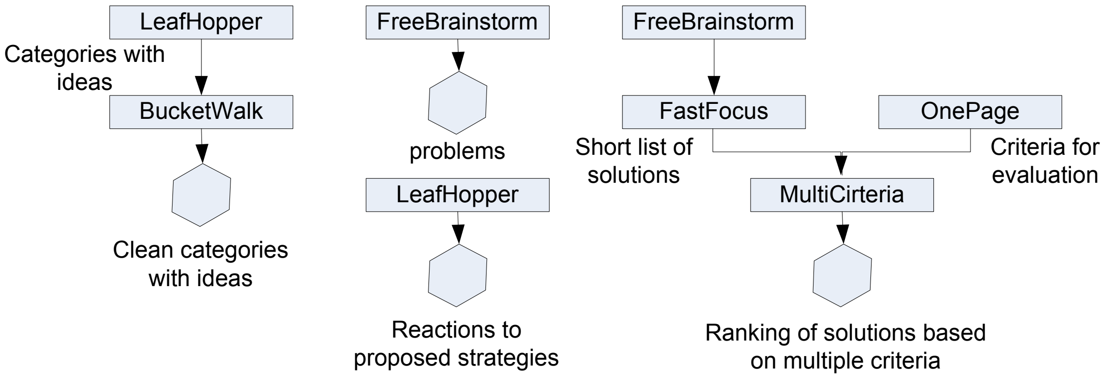


Figure 6.2. Different sequences of thinkLets; hexagon is end-result, text on arrows indicate in-between results.

Classification	
Pattern: Generate	***Result: Input***
FreeBrainstorm	FreeBrainstorm
OnePage	OnePage
Comparative	Comparative
LeafHopper	LeafHopper
DealersChoice	DealersChoice
PlusMinus Interesting	PlusMinus Interesting
TopFive	TopFive
BranchBuilder	BranchBuilder
TheLobbyist	TheLobbyist
DimSum	DimSum
PointCounterPoint	PointCounterPoint
	StrawPoll
	MultiCriteria
	CheckMark
	StakeHolderPoll
	BucketVote

Choice map ***ThinkLet combination***	***FreeBrainstorm***	***OnePage***	***Comparative***	***LeafHopper***	***DealersChoice***	***PlusMinus Interesting***	***TopFive***	***BranchBuilder***	***TheLobbyist***
Starting point?									
FreeBrainstorm		x							
OnePage									
Comparative									
LeafHopper									
DealersChoice									
PlusMinus									
TopFive									
BranchBuilder									
TheLobbyist									

Figure 6.3. Classification and choice map example.

Choice Map

In the choice map (see figure 6.3), introduced above, the thinkLets displayed vertical follow the thinkLets displayed horizontal. e.g. the square that is marked with x indicates the combination FreeBrainstorm followed by OnePage. The color of each square indicates whether the combination is excellent, (white) possible but tricky (gray) or impossible (dark). With combining thinkLets we mean that the output of the thinkLet is used as input for the following thinkLet (see example left figure 6.2). The first row indicates if the thinkLet can be used to start a session with no input from a previous collaborative activity. If it is not intended to use the result from the previous thinkLet, or previously created input, use this section (see example middle figure 6.2). If two deliverables are created and these should be combined in a next activity, this map does not offer support (see example right figure 6.2).

Step 4: Agenda building

A sequence of thinkLets is not yet a complete collaboration process prescription. Additional steps are required. The most important steps besides the thinkLets are discussed in chapter 5, table 5.3.The agenda should specify all information for each thinkLet, relevant for validation. A format for the agenda is displayed below (see. figure 6.4).

Activity	Task	Question/ Assignment	Deliverable	ThinkLet and Pattern	Time
1					
2					
Etc…					

Figure 6.4. Agenda format.

The first column is to identify and number the activities. Note that breaks, presentations and other activities should also be included in the process prescription. In the second column the task is described. An example of a task is “categorize ideas” or “brainstorm requirements”. The next column is reserved for the specific question or assignment to the group. Posing the right questions or assignments is one of the most vital steps in the design. The assignment or question should be:

- Required for achievement of the goal
- Not too complex, e.g. do not ask two questions at the same time
- Clear and unambiguous
- Specific about the characteristics of the results intended (e.g. “Write a detailed description of possible solutions”, or “state in one sentence, a possible solution”)

A last important aspect is the scope of the question which is specified in the question or separate in an introduction presentation. For instance, if the question is to specify success factors for “the project”, the introduction of the session should contain a brief explanation of the project scope if this is not yet a shared understanding of the group.

In the next column the deliverable is described; a specification of the output expected or a more general output like “ranking of the results”, or “categorization of the ideas”. In the fifth column the thinkLet and pattern intended are indicated. For each thinkLet it is important to specify the constraints, or to explain how they should be established. Constraints are for instance the voting criteria, the scale, discussion topics, etc. Also the tools need to be specified (GSS tool, required resources) to instantiate the capabilities required for the thinkLet. In the last column the estimated time for each activity is indicated.

Based on the agenda the process prescription for transfer can be documented, but often it is best to first perform the validation steps and document only the final design. The agenda will describe all information required to validate the design.

Step 5: Design validation

There are four ways to validate the design: pilot testing, walk-through, act it out (simulate), and discuss with another colleague:

- *Pilot testing:* This is a small scale implementation of the collaboration process which will allow the team members to assess the quality of the process. A pilot validation will reveal whether the process can be executed given the limitations of the resources, the stakeholders involved, and the skills of the practitioners. Furthermore, the pilot will give an indication of the quality of the results. Pilot testing will give insight in the effect of each intervention and thus increase the predictability of the process.
- *Walk-through:* A final assessment of the collaborative processes can be done performing a walkthrough with the practitioners and the client or a few of the participants. This validation will reveal pitfalls and difficulties for the practitioner, the likeliness of acceptance by stakeholders, the expected quality and efficaciousness of the results and the reusability. A walk-through will not really increase predictability as the process is discussed rather than tested.
- *Act it out (Simulate):* By simulating the design, the design team tries to answer the questions posed, and considers if these answers can be used in the next activity. The next question is whether the intended participants can also perform those activities. This requires that all information is available, and that the participants have the expertise to answer the questions posed. Furthermore, assessment is required to determine whether the resources are sufficient, and whether the result will be efficacious to the goal. Last, it is important to consider whether the practitioner can perform this activity. This validation tests the logic of the design. A simulation can be done using role-playing. However unless the real stakeholders are involved, predictability will not increase.
- *Expert Evaluation:* As each facilitator or collaboration engineer has his or her own style, each will have different solutions for a collaboration challenge. Discussing the design with colleagues will help to find better solutions for difficult activities and different thinkLets or methods for a certain challenge. The validation may help identify inefficient parts of a design and can be used to assess the different quality dimensions. A colleague can offer alternative approaches and verify the expected outcomes. However, this approach will not largely increase the predictability of the design.

During the validation, the list of quality dimensions offered in section 6.1.5 needs to be reconsidered. This can be used as an evaluation framework for the simulation or pilot and it can guide the walk-through or the discussion with a colleague.

Step 6: Documentation

To document a collaboration process each element of the template for the process prescription described in chapter 5 needs to be filled out. First, the assumptions that were derived in the task analysis phase should be documented in an assumptions document. Furthermore, the facilitation process model needs to be drawn. Next, the script for the practitioner should be written. The thinkLet scripts will be offered as a generic template in which the capabilities, roles, and, constraints should be instantiated or an instruction for the practitioner on how to establish these parameters should be prescribed. This instantiation will be further explained in chapter 7. Last, we need to offer the practitioner the set of cue cards. Furthermore, the collaboration engineer can make slides to introduce the goal, the results, the program, the way of working and the different questions and assignments to support the practitioner in instructing the group.

6.3.3 Case study: evaluation of the design support

To validate the design approach and the support offered in each phase of the design process we let a group of students and academics use a manual describing this approach in combination with a set of thinkLets, to design a collaboration process.

Method

We documented the design approach in a design support booklet; a manual to design a collaboration process prescription. The booklet contained the following information:

- An introduction to explain how to use the booklet
- A description of requirements to a high quality design
- A step-by-step explanation of the design approach with a running example for each step
- A list of design guidelines; tips and tricks

The booklet was not focused on creating a process prescription for transfer, and thus predictability and transferability were not addressed as requirements. Instead the evaluation assumed that peers should be able to facilitate the design. We therefore evaluated the ability of the design approach to create efficaciousness, acceptable process prescriptions that make optimal use of resources in a single instance of the task, and whether it could be facilitated by a peer. The subjects only had to write a description of the collaboration process, to build the agenda, and to create the facilitation process model. They did not have to create the entire process prescription template, as this was beyond the learning goals of the participants. The process prescription has been evaluated separately in the case studies described in chapter 5.

To evaluate the design approach we wanted to know if the design approach was in fact supporting the users in their effort to design a collaboration process. We used the following metrics to measure whether the different elements of the design support booklet were helpful in designing a collaboration process; meaning they offered effective and efficient support and cannot be improved:

- Use
- Usefulness
- Supportiveness for time saving
- Understandability
- Ease of use
- Supportiveness to improve quality
- Need for improvement

Furthermore, we collected their grades and we asked the students to indicate the effort of the design task. Last, we asked to specify what could be improved to support them better in the design assignment. The case, assignment and questionnaire that were used can be found in appendix 8, the design format was only used in the second case.

Case 1 design assignment for (under)graduate students

To evaluate this design approach we let 26 students at the University of Nebraska at Omaha, and Delft University of Technology in the Netherlands design a collaboration process based on a case description. The group was a mix of graduate and undergraduate students that participated in a course on Facilitation, GSS and Collaboration Engineering. The students received a booklet with the design approach and a set of thinkLet descriptions. The case was a real project description of a GSS session run in the Netherlands by facilitators of the Delft University of Technology, but names of the organizations involved were changed. The students were graded for this assignment. The students had to design a collaboration process according to the case description, and afterwards filled out a questionnaire.

We asked the students for tips for improvement both in the questionnaire and in informal evaluation sessions and interviews. Based on these insights we revised the booklet several times. The first evaluation of version 1 revealed the following suggestions for improvement:

- Add further support for the validation and simulation of design with checklist and visualization
- Explain the task thinkLet choice; explain differences between thinkLets, similarities, strengths of each thinkLet and weaknesses
- Add better explanation of the patterns of collaboration
- Elaborate on the thinkLet choice with respect to appropriation and efficiency
- Offer videos of successful and failed thinkLet implementations with tips and guidelines
- Offer thinkLet memorization support
- Add a template for the design
- *Add examples*
- *Add tips on how to get started, an introduction for the novice*
- *Add further explanation of the requirements to the design documentation and models*
- *Explain elements of high quality design in a consistent way and with explanation on how they affect the quality of collaboration*
- *Add a further step by step explanation of the design approach especially on thinkLet task choice and decomposition, with examples, learning checkpoints, and a small exercises*

The italic improvements were made in version 2. In this version we added a running example, as this was a need, indicated by many students. Additionally, we added an introduction on how to use the booklet, describing the purpose of each part, and extended and updated the descriptions of the approach and documentation requirements. Last, we added an explanation of the criteria for high quality design. In versions 3 we updated the design approach explaining its iterative nature and integrating the high level and low level decomposition to one step. Furthermore, we added a preliminary checklist for the analysis. The explanation of the choice among thinkLets and the tradeoffs involved were extended in this version as well, and finally, the requirements to the agenda documentation where sharpened. The suggestions for improvement based on this version were very limited and mostly addressed the need to further explain how each element should be used. However, we still had some improvement suggestions from version 1 that could be implemented.

In version 4 we further updated the quality of design description; we extended the checklist for analysis, explained the quality trade-offs for the choice among thinkLets from a facilitation perspective and explained how choice and decomposition can affect each other. Next, we added classifications of the thinkLets based on results and patterns, and the choice map. We also updated the documentation requirements. Furthermore, we extended and updated the design guidelines.

Results

In Table 6.6 - 6.11 we will present the results of the questionnaire among students for version 1, 2, 3 and 4. For all tables the scale was 1-5 (1) being strongly disagree and (5) being strongly agree, unless indicated otherwise.

Question	***v1***	***v2***	***v3***	***v4***
	n14	***n5***	***n5***	***n2***
I found the design exercise difficult	3.90	3.00	2.80	2.50
I found the design exercise took me a lot of effort	4.00	4.20	2.00	3.50
Average time spend in hours	5.95	11.30	9.30	10.50
Average grade (scale 1-10, 1 =lowest, 10 highest)	7.90	7.80	7.90	-

Table 6.6. Effort of the design exercise compared to result.

In Table 6.6 we see that it became less difficult to design a process with the new versions of the booklet. However, the time spend on the design and the effort required for the exercise varied. An explanation could be that while we improved the booklet, the amount of information increased. This presents a tradeoff; more support might be helpful but also increases the workload of the task. The average grade of the students remained stable. In comparison, a group that had to design a process based on a similar case description (n15) in a similar setting had an average grade of 7.5, slightly lower.

Design approach	v1	v2	v3	v4
	n 14	n5	n5	n2
I used this information	4.14	4.60	4.80	4.00
I found this information useful	4.07	4.60	4.60	4.00
This information saved me time	3.57	4.20	4.80	4.00
I found this information easy to use	3.57	4.00	4.60	4.50
I fully understood this information	3.93	4.00	4.60	4.00
This information helped me to improve my design	4.07	4.60	4.60	4.50
This information should be improved	*3.54*	*2.00*	*1.80*	*2.00*

Table 6.7. Design approach evaluation.

Quality criteria	v1	v2	v3	v4
	n 14	n5	n5	n2
I used this information	4.00	4.00	4.20	3.50
I found this information useful	3.93	4.00	4.60	3.00
This information saved me time	3.36	4.00	3.40	3.25
I found this information easy to use	3.64	4.60	4.20	4.00
I fully understood this information	3.71	4.20	4.60	3.50
This information helped me to improve my design	4.07	4.20	3.80	3.00
This information should be improved	*3.15*	*2.75*	*2.20*	*2.50*

Table 6.8. Quality criteria evaluation.

Facilitation process model	v1	v2	v3	v4
	n 14	n5	n5	n2
I used this information	4.36	4.80	4.40	4.00
I found this information useful	4.14	4.60	4.20	4.00
This information saved me time	4.14	4.60	4.40	3.50
I found this information easy to use	4.00	4.80	4.60	3.50
I fully understood this information	3.92	4.60	4.75	4.00
This information helped me to improve my design	4.14	4.20	4.20	3.00
This information should be improved	*2.64*	*2.50*	*3.60*	*3.50*

Table 6.9. Facilitation process model evaluation.

Agenda format	v1	v2	v3	v4
	n 14	n5	n5	n2
I used this information	4.14	5.00	4.25	4.00
I found this information useful	4.07	4.80	4.00	4.00
This information saved me time	3.71	4.80	4.00	4.00
I found this information easy to use	3.50	4.40	4.00	4.00
I fully understood this information	3.86	4.40	4.00	4.00
This information helped me to improve my design	4.00	4.60	4.25	2.00
This information should be improved	*3.36*	*2.00*	*2.75*	*2.00*

Table 6.10. Design format evaluation.

Design guidelines	v1	v2	v3	v4
	n 14	n5	n5	n2
I used this information	3.50	3.60	3.80	4.00
I found this information useful	3.64	4.40	4.20	4.00
This information saved me time	3.50	4.25	3.60	3.00
I found this information easy to use	3.57	4.25	4.20	3.00
I fully understood this information	3.57	4.50	4.40	4.00
This information helped me to improve my design	3.64	4.50	3.80	4.00
This information should be improved	*3.36*	*2.00*	*2.40*	*3.00*

Table 6.11. Design guidelines evaluation.

The step-by step design approach was used more with each version, was considered more useful, it saved more time and can considered more easy to use. It was better understood and it helped more to improve the designs of the students. Last, it needed less improvement.

The quality criteria showed a different pattern; they were used more, and considered more useful and understandable but they did not save more time, did not become easier to use and did not help more to improve the design. At the same time it was not indicated that it needed improvement. The same conclusion can be drawn for the design guidelines and the facilitation process model. These aspects of the design support have been extended in the subsequent versions and therefore their use took more time and effort. The evaluation of the agenda format was most positive in version 2, but did not change much in subsequent versions and got a stable evaluation in version 3 and 4.

Case 2 Graduate students and faculty design workshop

We further updated the design approach with insights from the choice study (See chapter 4 textbox 4.5) and implemented it in a final case study with graduate students and faculty members of the Manchester Business School. 16 people participated in a two day workshop. Some participants were novices to facilitation, others were experienced facilitators but used different approaches (mostly based on decision theories). In the first day they experienced and got familiar with approximately 10 thinkLets in a GSS setting. On the second day the participants were first introduced to the design approach and the supporting materials. Then they had two hours to work on a case-based collaboration process design. Some of the staff members had to do other things and spend less time on the assignment. Others made the assignment after the course and took more time. Not all participants were able to finish the process design, but most managed to create a sequence of thinkLets that they would use to support the group. After the design exercise, the participants filled out a questionnaire to collect their perceptions on the design approach as presented in the booklet and the lecture. The questionnaire, case, assignment and design format were the same and can be found in appendix 8.

Results

Tables 6.12-6.16 present the results of the questionnaire among participants for the final version of the design support booklet. Two questionnaires were excluded because they were not entirely filled out, so n=14 for all questions. For all tables, the

scale is 1-5: (1) strongly disagree, (2) disagree, (3) neutral, (4) agree, and (5) strongly agree. We did not grade the assignments as results would not be comparable; the background and experience of the participants was too varied.
The results of the final version were:

Question	Final n 14
I found the design exercise difficult	3.7
I found the design exercise took me a lot of effort	3.1
Average time spend in hours	1h20

Table 6.12. Design effort and result.

Design approach	Average	Stdv
I used this information	3.57	1.09
I found this information useful	4.00	0.55
This information saved me time	3.93	0.62
I found this information easy to use	3.86	0.53
I fully understood this information	4.14	0.53
This information helped me to improve my design	3.43	0.51
This information should be improved	*2.50*	*0.65*

Table 6.13. Design approach.

Quality criteria & Design guidelines	Average	Stdv	Average	Stdv
I used this information	3.08	1.10	2.79	1.25
I found this information useful	3.77	0.43	3.57	0.65
This information saved me time	3.38	0.76	3.21	0.70
I found this information easy to use	3.23	0.61	3.29	0.73
I fully understood this information	3.38	0.93	3.50	0.52
This information helped me to improve my design	3.46	0.65	3.29	0.73
This information should be improved	*3.08*	*0.82*	*2.69*	*0.75*

Table 6.14. Quality criteria and design guidelines.

Agenda format and Facilitation process model	Average	Stdv	Average	Stdv
I used this information	3.79	0.89	3.43	1.09
I found this information useful	4.00	0.39	3.86	0.53
This information saved me time	3.79	0.58	3.57	0.76
I found this information easy to use	3.68	0.72	3.86	0.53
I fully understood this information	3.64	0.74	3.86	0.66
This information helped me to improve my design	3.86	0.53	3.57	0.85
This information should be improved	*2.71*	*0.91*	*2.86*	*0.86*

Table 6.15. Design format and Facilitation process model.

Choice map and Classifications	Average	Stdv	Average	Stdv
I used this information	3.36	1.28	3.14	1.10
I found this information useful	4.07	0.73	3.86	0.53
This information saved me time	3.64	0.84	3.36	0.63
I found this information easy to use	3.64	0.84	3.64	0.63
I fully understood this information	3.93	0.62	3.71	0.47
This information helped me to improve my design	3.43	0.65	3.64	0.63
This information should be improved	*2.69*	*0.95*	*2.69*	*0.63*

Table 6.16. Choice map and classifications.

Since the participants had only two hours to perform the design exercise, the use of the supporting materials is somewhat limited, and had a high standard deviation. This is consistent with the fact that some of the participants were able to create a process design within the two hours while others did not manage to do so. However, since we presented the information and explained it in an example we feel that the results on usefulness, time saving, ease of use, understandability and supportiveness are representative. Each of these factors scored slightly too fairly positive. The design approach, agenda format and the choice map were considered most useful, followed by the classification and facilitation process model. The design guidelines and quality criteria scored lower on usefulness and also on time saving. This is not surprising as these aspects are mostly used for validation and further iterations of the design approach.

Further, it was stated that the design guidelines were considered useful only for users that already know a set of thinkLets by heart. The suggestions for improvement included the following: for the design approach, more examples were considered useful; for the quality criteria the participants requested more explanation and more examples about the tradeoffs and the specific tradeoffs that emerge when choosing among thinkLets. Some participants also suggested that the guidelines should be clustered, for instance to the patterns of collaboration. The choice map and classification were found to be rather complex, yet at the same time useful. The agenda format and facilitation process model were considered overlapping. However, while the agenda was considered more useful, the facilitation process model was easier to use and more understandable. The agenda format and the classification of the thinkLets were considered most supportive. The design guidelines and choice map were considered least supportive. All supporting elements scored low on the need for improvement.

Although positive, the results of this final case study were slightly less positive than the results of the students. We can offer three explanations for this. Firstly, the time spend on the task was limited and therefore supporting materials were used only shortly, which might have affected results. Second, the background of the participants were very different, while the students from Delft and Omaha had very limited background in collaboration and group support, many of the graduate students and faculty in Manchester had experience or expertise on various aspects of group work and were therefore able to compare this approach and the supporting materials with other approaches and materials, and with their own approach. Furthermore, as we learned from cognitive load theory in chapter five, support for novices might have reverse effects (Kalyuga et al., 2003) on experienced or experts on the task as it interferes with their own approaches, as was mentioned by several faculty members in the evaluation. Third, while improving the booklet its complexity and the amount of information increased. The suggestions for improvement now mostly suggest ways to offer more overview and insight in the existing information, rather than additional information. Based on this last insight we think that we should not pursue to improve the design approach in the form of a booklet, but rather we need to build a tool that supports the analysis, the sequencing of the activities, the selection of thinkLets and their instantiation in various ways.

6.4 Efficaciousness and acceptance support for Collaboration Engineering

Based on the experiences with 40 users, we conclude that the Collaboration Engineering design approach, as described in the booklet, offers useful and effective support, especially for novice collaboration process designers. However, it takes a considerable amount of time to digest and use the information and supporting materials. Some elements are considered rather complex. We feel that it will be very hard to further improve the current approach on 'paper'. The minor improvements suggested regarding the final version of the booklet will only increase the amount of information without adding new insights and support. To better support collaboration process design efforts, we feel it is critical to create a computer based expert tool. We will further elaborate on this in chapter 9 when discussing future research.

In this chapter we presented the design approach and design support to help the collaboration engineer in designing a collaboration process that is efficacious to its goal and acceptable for the stakeholders involved. To support efficaciousness we offered selection guidance: support in creating combinations of thinkLets, and a classification of thinkLets to support the choice of thinkLets to create an efficacious result and pattern of collaboration. To further support the design effort thinkLets should be documented with explicit notion of the possible combinations, alternatives, and classifications. Furthermore, insights on when to use the thinkLet can be recorded. Such information could ultimately be used to create an expert system which supports the collaboration engineer in selecting and sequencing thinkLets to design a collaboration process. To create very specific effects with the thinkLets, collaboration engineers should also have the ability to make small modifications for specific effects. In the thinkLet conceptualization we will explain how we can further support this. As indicated in the section on analysis, the goal of a collaboration process can both be a tangible goal (e.g. a solution or decision), or an experience goal (e.g. awareness of a problem). To support efficaciousness of both we need to understand the effects of thinkLets both in terms of results and in terms of patterns of collaboration.

Besides support for efficaciousness, the collaboration engineer should also analyze the group and stakeholders involved as described in the design approach, to increase acceptance. ThinkLets will also offer some support for acceptance of the collaboration process design. Since thinkLets are best practices they have been accepted by groups in many situations. The reason for this is that many thinkLets have build in principles such as equity and democracy because they originate from techniques used in combination with GSS, in which such principles are incorporated (Vreede and Bruijn, 1999, Bostrom and Anson, 1992). For instance in thinkLets that instantiate the 'generate' pattern, each participant will get equal opportunity to contribute and in most voting thinkLets a democratic voting method is used. When thinkLets without these principles are used, the collaboration engineer should very carefully script the explanation for this to ensure that the practitioner can explain such choice to the group.

In the design approach we did mention the analysis of the available resources, which should be taken into account while designing the collaboration process. In the thinkLet concept (described in the next chapter) we will specify the resources required. With this specification, the practitioner can prepare the resources. This will support reusability of the collaboration process. Another factor that supports reusability is the predictability of the thinkLets. When thinkLets create the same effect in each instance of the collaboration process, our process can be re-used for each of these instances of the process. In the next section we will further explain how thinkLets support the reusability and predictability of the collaboration process design.

Chapter 7. Predictability and reusability: the pattern language

In chapter 5 and 6 we explained in our conclusions how we supported transferability efficaciousness and acceptance with the design and transfer approach. In this chapter we will add predictability and reusability of the collaboration process design based on the thinkLet concept. However, since thinkLets will also support the design and transfer approach, they also support efficaciousness, acceptance and transferability. We will reflect on each of these effects in the conclusion but we will focus first on increasing reusability and predictability.

Reusability is the extent to which the design can be used successfully in different instances of the task. Reusability can be approached in two ways:

- Fit to the resources available. As discussed in chapter 6; if the resources required for the process are not available in all instance of the recurring task, the process will not be reusable.

- Predictability of the pattern of collaboration and result of each step of the process prescription. If the instructions for activities in the process prescription render different outcomes in each instance of the task, it can not be used for a recurring task.

Predictability of the thinkLets is the extent to which the design, when used as prescribed, creates a process and results as intended by the collaboration engineer. Predictability of the thinkLets used in the collaboration process design can be improved in three ways:

- The first approach is through post-analysis such as empirical testing, pattern analysis, and the harvesting of best practices (Briggs and Vreede, 2001, Briggs et al., 2001, Kolfschoten et al., 2004a, Enserink, 2003, Vreede and Briggs, 2001). ThinkLets have been documented, and they have been recognized in transcripts of GSS sessions. Furthermore, they have been used and evaluated in several case studies (Alaa et al., 2006, Appelman and Driel, 2005, Vreede et al., 2005, Bragge et al., 2005, Briggs and Grunbacher, 2001, Harder and Higley, 2004, Harder et al., 2005). These studies confirm the recurring nature of the thinkLet effects.

- A second approach is through theory building. Theoretical understanding of the thinkLet intervention and the resulting patterns of collaboration and results will help collaboration engineers to understand and predict the effect of the thinkLet (Briggs, 1994, Briggs et al., 2005, Briggs et al., 2004, Briggs et al., 2003b, Santanen et al., 2004). So far, theory has been developed about satisfaction (Briggs et al., 2004), productivity (Briggs, 1994), consensus (Briggs et al., 2005) and creativity (Santanen et al., 2004).

- The third approach is the rigorous and precise documentation of the intervention using the thinkLet conceptualization framework. This last aspect will be the main contribution of this chapter.

Before we will explain the theoretical basis for the thinkLet concept which will offer the framework for rigorous documentation, we need to understand to what extent we can predict the outcomes or effect of interventions. Two outcomes of each thinkLet should be predictable; the pattern of collaboration that emerges in the group, and the type of result. The result is a knowledge-outcome of the collaboration process. It can be a data set or a modification of a data set, but it can also be some kind of understanding ranging from awareness, to shared meaning, to empathy. A pattern of collaboration characterizes the ways in which group activities can move a group toward its goal (Briggs et al., 2006b). We described these outcomes in detail in chapter 6.

In the next section we will describe the theoretical basis for the thinkLet concept. In section two we will present the thinkLet concept, the aspects that need to be documented to support the design and transfer of the collaboration process prescription and to support the different quality dimensions. In the third section we will describe the resulting patterns of collaboration in more detail. We will end the chapter with conclusions on the contribution of the thinkLet concept to each of the quality dimensions.

7.1 Theoretical basis: the pattern language and rule based interventions

To offer a theoretical basis for the thinkLet conceptualization we looked at the design pattern concept used in architecture and software engineering. Furthermore, we will try to gain a through understanding of the elementary content of action interventions though the used of parameterized action representation as used in artificial intelligence.

7.1.1 Design patterns

The original pattern concept described by Alexander (Alexander, 1979) has been widely adopted in the software engineering world after introduction by the gang of four (Gamma et al., 1995). For example, Lukosch and Schümmer (2006) propose a pattern language for the development of collaborative software. Patterns are successfully used in related fields such as communication software (Rising, 2001), productive software organizations (Harrison and Coplien, 1996), e-learning (Niegemann and Domagk, 2005) and for knowledge management(May and Taylor, 2003) Alexander (1979) originally described his design patterns as re-usable solutions to address frequently occurring architectural problems. In Alexander's words: "a pattern describes a problem which occurs over and over again and then describes the core of the solution to that problem, in such a way that you can use this solution a million times over, without ever doing it the same way twice (p. x, Alexander et al., 1977)" Patterns thus not only offer us a basis for segmentation of the process to reduce intrinsic cognitive load (see section 5.3.1) but they also link solutions to problems, as discussed in section 5.3.1. This might enhance the building of high quality schema, which supports transferability. ThinkLets have the same intention; they are best practices of expert facilitators to support groups in their collaborative efforts to achieve goals without running into the challenges that group work poses. The challenges that thinkLets address are recurring situations where thinkLets can be used to advance the group to their goal. Alexander (1979) suggests a number of different purposes for design patterns and the pattern languages they comprise. We summarize these below and reflect on the similarities with thinkLets as design patterns for Collaboration Engineering (Vreede et al., 2006a):

Providing a convenient common language for communication
Design patterns are a language, a vehicle for communication. They enable the users of the pattern language to name and share complex processes without having to explain them over and over again in detail. We know from our interviews and surveys that facilitators often share their facilitation techniques with colleagues. ThinkLets will further support this. ThinkLets will also support practitioners of the same thinkLets-based collaboration process to share insights and experiences in communities of practice (Vreede and Briggs, 2005). For this purpose, patterns require powerful identification methods.

Inspiring and designing new or improved patterns
Patterns describe solutions to recurring problems. A problem as described by Alexander (1979) is a design requirement, a need of people that can be fulfilled with the creation of an artefact. Patterns can be used to build a solution to a problem, but they can also be used to inspire designers to create new patterns. We saw in Chapter 6.2.2 that 77% of the facilitators that answered our questionnaire (see textbox 4.4 in chapter 4 for method) created their own facilitation techniques to solve their group's problems and to support them in achieving their goals. One of the approaches to design new patterns can be the combination of smaller elements of pattern-solutions. This approach can be supported by composing thinkLets of distinguishable, reusable rules, as further explained below.

Designing larger systems based on individual patterns
Alexander's (1979) patterns do not only support the construction of a house, rather they provide solutions for living in a broader perspective. His patterns can be used to create houses, towns, and communities. ThinkLets should offer a similar function, they should support groups to collaborate more efficient and effective and they should support not only the achievement of a single collaborative task, but they should also offer practitioners, facilitators and organizations an instrument to teach groups to collaborate more successful.

Teaching, capturing, and sharing expert design knowledge
Patterns were originally intended by Alexander (1979) to support teaching, capturing, and sharing expert knowledge on building and architecture. ThinkLet patterns were derived in a similar fashion. The origin of the thinkLet concept lies in the capturing of expert facilitators' best practices. The initial set of thinkLets included only facilitation techniques that have been used successfully in the field by expert facilitators (Briggs and Vreede, 2001). They can be used to teach practitioners and novices, but they will also offer a valuable library for expert facilitators.

Enabling 'anyone' to create with patterns

Alexander's (1979) original idea was to enable "anyone" to build a house or town/community. He believed that his books should enable people to design high-quality houses for themselves. Collaboration Engineering does not entirely support this ambition. While the patterns should be transferable to practitioners, we do not intend to let practitioners design their own collaboration process. However, there are situations in which we will have to find the edges of process transfer and self-design by practitioners. For instance in processes with high uncertainty and dynamics such as crisis situations, a practitioner should be offered a far more flexible process than in a design for a monthly progress evaluation.

Creating designs that improve the quality of life

Alexander's (1979, 1980) pattern language serves a higher purpose: it aims to enable the creation of buildings that are lively, that improve the quality of human life. The patterns he and his colleagues described should create morally sound objects. In like manner, the Collaboration Engineering approach, seeks to create collaboration processes that constitute high quality collaboration which is characterized by efficiency, effectiveness, productivity, satisfaction and commitment. Further, many thinkLets have built-in quality requirements, comparable to Alexander's forces and ethics regarding collaboration in groups such as principles of democracy and equality. Most thinkLets enable each person in the group to contribute equally, as is also the intrinsic value of the GSS environment (Vreede and Bruijn, 1999). However, not all thinkLets work on democratic principles; some for instance prescribe specific rights for experts.

Creating coherent systems

The hierarchical nature of pattern languages seeks to engender the creation of a whole coherent system, instead of loosely-coupled individual components that are not in harmony with their environment (Alexander, 1979, 1980). In Collaboration Engineering, the design of a collaboration process also is more than the combination of a number of disjointed thinkLets. The design process takes into account the environment in which the thinkLet is to be executed, e.g. in terms of organizational culture, nature of the expected group members, and skill level of the practitioner. It takes into account the activities that will precede and follow a given thinkLet, and the means by which each thinkLet advances the team toward their agreed goal. Additionally, we offer a similar hierarchy with the classifications of thinkLets based on results and patterns of collaboration.

The conceptualization of the design pattern has been altered by Gamma et al. (Gamma et al., 1995) based on the original concept by Alexander. They defined design patterns for object-oriented software design as descriptions of communicating objects and classes that are customized to solve a general design problem in a particular context. Shortly after Gamma et al. (1995) introduced the object oriented design patterns, Alexander was asked to reflect on the use of patterns in software engineering (Alexander, 1996). He expressed two key concerns about the way pattern languages are used for software engineering. First, he argued that software design patterns are not based on a philosophy of the quality of human life; they appear to be only a vehicle for communication. Second, he argued that software design patterns appear not to focus on creating a whole and coherent system, rather they aim to design an independent object without taking into account how it should contribute to the larger whole. Like architectural and software design patterns, thinkLets should serve as a vehicle of communication, and serve to increase the efficiency of the design process. However, where Alexander focuses on a ‘general’ quality of life concept, thinkLets focus on quality of collaboration. Design process efficiency, effectiveness and productivity is one goal of Collaboration Engineering, but additionally, the Collaboration Engineering design should increase the ability of groups to not only attain the group goal, but to accommodate individual goals of the participants, and to increase satisfaction and commitment of group members.

In the field, the thinkLet pattern language serves the design of a complete, coherent system of collaboration. ThinkLets by themselves serve little purpose; very few collaboration processes consist of only one thinkLet. A key value of thinkLets as design patterns lies in their combinability into larger repeatable sequences, and eventually into integrated systems of collaborative processes that further the larger goals of an organization and the people it serves. Collaboration engineers seek to assure that the processes they design integrate seamlessly with other processes that precede them, parallel them, and follow them to crate a coherent collaborative solution to the challenges of group work.

With these basic contours of the design patterns and the pattern language, we need to derive a conceptualization of thinkLets that will increase not only transferability and reusability, but that will support the designer in making design decisions that increase acceptability and efficaciousness and that increase the predictability of the different interventions of the practitioner. To fulfill this last requirement to design quality we will look at interventions on a micro level to find an approach to increase their predictability.

7.1.2 Rule based interventions

Besides the use of design patterns as a basis for the structure of the thinkLet concept, we can also offer structure on the level of the specific group instruction or facilitation intervention. Facilitation interventions are a type of communication, and can thus be analyzed as a communication system.

In communication theory many different models are proposed. Craig argues that the transmission model is a useful lens to analyze communication as an intentional act to achieve some anticipated outcome, and to study distortion and misunderstanding (Craig, 1999). We will apply this model to explain the communication of a facilitation intervention.

A facilitation intervention or instruction is in essence an advanced type of communication. It consists of a message that is transmitted with use of a channel/vehicle from a sender (practitioner) who can encode the message and send it to a receiver (participant/group) who can decode this message. From this decoding effort, a meaning can be distilled and in response an action can be performed. A response can be either in terms of feedback, i.e. the transmission of a return message, or a communication effect, i.e. any other action in response to the message (Krone et al., 1987) which will constitute a pattern of collaboration, and a specific result. In such a generic system there is no predictability of the action as an effect of the message or its content, to increase predictability several conditions should be met.

A basic set of pre-conditions and circumstances is required to create a situation in which the communication effect (the action in response to the instruction) is in line with the intention of the message sender. In other words, conditions for the group to accept and execute the instructions of the facilitator/practitioner. Such conditions involve first of all the absence of noise that can prohibit or diminish the quality of transmission. Thus the language of instructions to the group should be clear and unambiguous. Further, there is a need for some level of goal congruence (Briggs, 1994, Briggs et al., 2004) where the intentions of the sender (facilitator/practitioner) are not conflicting with those of the receiver (group, and individual participants) and some level of trust where the receiver believes in the truthfulness of the intentions of the sender. When the receiver does not want to listen or does not trust the sender, the interpretation of the message might not match the intention of the sender, in other words the group does not accept the instructions from the practitioner.

Under these conditions, a message containing an instruction for a specific action is likely to result in an intention (Briggs et al., 2005) to (attempt to) perform/execute that action. In other words, when the group trusts and receives the instruction from the practitioner, and supports its goal, it is more likely that the group will follow the instructions and consequently that the intended pattern of collaboration and result will be achieved.

Assuming these conditions for congruence of intention between sender and receiver are in place, there are two more reasons not to perform the action as intended by the sender. The first reason is an inability that occurs when the receiver does not have the resources, skills or knowledge required to execute the action instructed by the practitioner. The second reason is that the instruction is unclear, too complex or incomplete and hence the encoding of the intention of the sender or decoding of the message by the receiver does not result in complete transmission of the instruction. In other words, the group does not understand the instruction of the practitioner as it was intended by the collaboration engineer. For this purpose the message should be complete; it should precisely describe the action intended by the sender. When information about the intended action is not communicated, the response action will not fully meet the intention. We therefore need to find a way to describe actions, which is complete enough to create the intended effect.

To realize an intention by means of instruction, two types of instructions are required (Badler et al., 1998). First, there are static instructions (commands) as explained above in which one or more instructions are given to initiate the key activities of the process. Secondly there are dynamic instructions in which the actions performed by the group are adjusted to resolve a discrepancy from the intention. These interventions are conditional. We will refer to the first type of intervention described above, and illustrated in figure 7.1, as an instruction intervention.

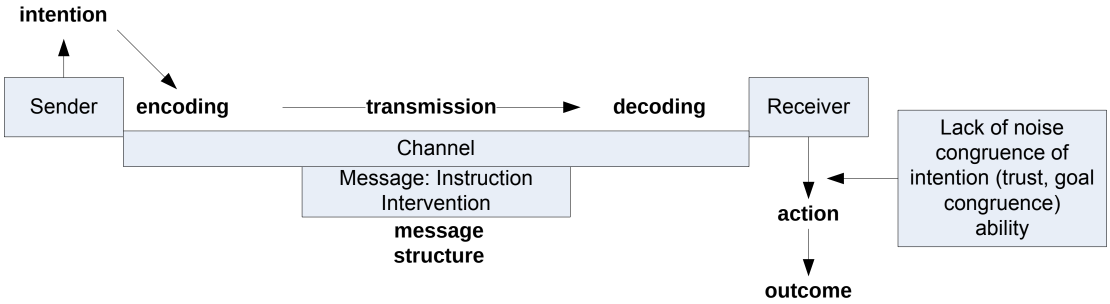


Figure 7.1. The process of an instruction intervention.

To create a predictable group process with predictable outcomes, a practitioner should not only give instructions that evoke a predicable action, (s)he should also make interventions that are intended to adjust an action when it does not correspond with the intention (labeled discrepancy). For different types of discrepancy, different interventions are required. For instance, if the action of the group is to add solutions, the group can diverge from the intention of the instruction in that the solutions they add are not specific enough or in that the solutions do not solve the problem. In the first case a message should be conveyed to evoke more specificity of the solutions. In the second case a message is required to create more shared understanding of the problem. For this type of messages we also offer a message structure to increase predictability. We will call these messages adjustment interventions. Adjustment interventions are thus instructions of the practitioner with the intention to adjust the actions of the group members in order to ensure that the outcome of these actions meet the intention of the process prescription; the intended pattern of collaboration and the intended result. The resulting framework is illustrated in figure 7.2

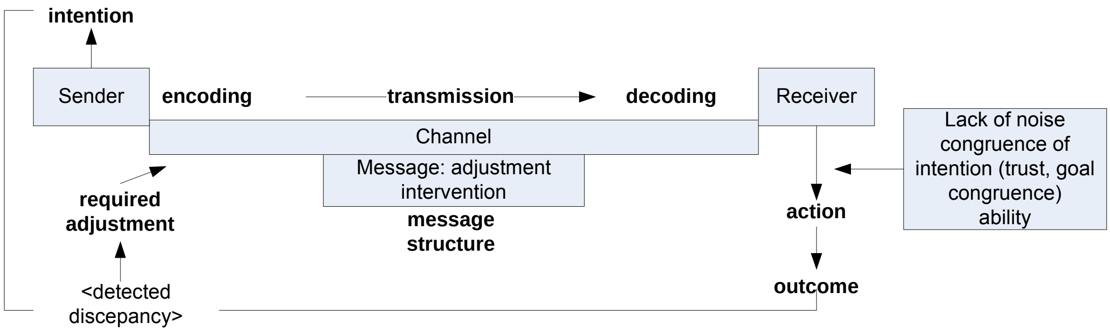


Figure 7.2. The process of an adjustment intervention.

To increase the predictability of the practitioner interventions, its encoding should be done in a way that minimizes the chance of errors in decoding. For this purpose we want the instruction to be precise and complete, but it should not be too complex or cause an information overload. To derive a template for the structure of the practitioner interventions to instruct the group in performing the intended actions we studied production systems and representation systems as found in artificial intelligence.

Artificial intelligence literature shows different ways for mimicking human behavior (Badler et al., 1998, Newell and Simon, 1972, Rich, 1983, Winston, 1984, Zhang, 1997). We can use instructions to evoke this behavior as plain actions (if condition applies, carry out action), for problem solving (if problem occurs, search for action that makes the problem go away), and for goal satisfaction (see how a desired state can be achieved from the current state using the production rules). The structure of the thinkLet can benefit from parameterized action representation as described by Badler et al (Badler et al., 1998, Badler et al., 1999). Badler specifies the actions of virtual humans (avatars) in natural language in a parameterized action representation. This rule structure is useful as a basis for the thinkLet concept. We will therefore apply parameterized action representation to our framework for instruction and adjustment interventions to increase the completeness of the interventions made by the practitioner and designed by the collaboration engineer. Using the action representation from Badler we want to derive a structure for interventions that contains all elements required to make the group perform the required actions. When elements of this representation are not communicated, the action will be less predictable.

The difference between an instruction intervention and an adjustment intervention is that the instruction intervention only conveys a rule that describes the actions each group member needs to perform to realize a specific effect, while an adjustment intervention is conditional; if a certain divergence between intention and effect is detected, the rule should be applied. In some cases the practitioner should monitor whether the condition applies, in some cases the participants can do this themselves. An instruction to end a specific action can also be conditional. An example of a conditional rule for a practitioner could be "if the solutions generated by the group are not specific enough, repeat the brainstorming rule: participants add solutions to the page that solve the problem defined". An example for a conditional rule for participants could be "if you want to respond to a solution you read, add a yellow note to the solution and add your response to the note"

An action as described by Badler et al (Badler et al., 1998) contains some activity performed by an agent with the use of, or applied to some objects in some specific manner and according to a specific path and under some conditions. For thinkLets we will not focus on movement actions per se and therefore we will set the parameters of the action instruction a bit more generic. In thinkLet rules we will specify a role (agent) who has to perform an activity (action) using or applied to some type of physical resource labeled 'capability' under some constraints. In thinkLets, for instance in the case of our instruction to add solutions, we can specify that the members of a **group** (role) should **add** (action) their s**olutions to the problem** (constraint) on a **page** (capability) using an **input device** (capability). A constraint is thus any specification of or limitation to the action. In our example, the constraint to the action "add" is that the addition should involve solutions that solve the specific problem described. By describing required capabilities instead of specific physical resources (page can be instantiated with a flip-over, input device can be instantiated with a marker) we make our rules platform independent, they can be implemented with a variety of tools that avail the required capabilities. Collaboration Engineers can instantiate these capabilities in the process prescription or they can instruct the practitioner to choose appropriate resources for instantiation. To store our rules we should not only describe the condition under which it should be executed but also its intended effect.

To offer a framework for the message structure we can use this rule concept to structure interventions as illustrated in figure 7.3.

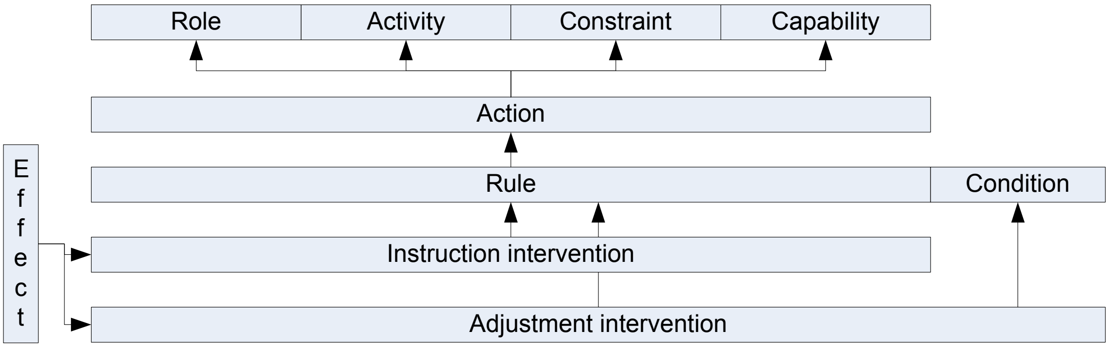


Figure 7.3. The elements of rule based interventions.

This rule concept can be used to make complete instruction and adjustment interventions that describe very precisely the action that a participant in the collaboration process should perform. Therefore we will use this parameterized action representation to describe the rules that are communicated to the group by the practitioner to instruct or adjust their action in the thinkLet conceptualization. When elements of the rules are not communicated to the group, the intervention will be less predictable.

7.2 Increasing design quality: the thinkLet concept

In order to further support the Collaboration Engineering approach we will further develop its most important basis; the thinkLet concept. We will first summarize the requirements for the concept which are not only derived from the theory above but also from the chapters on efficaciousness, acceptance, and transferability. Next, we will present the conceptual design.

7.2.1 Requirements to the thinkLet concept

To derive an overview of the requirements for the thinkLet concept we will first identify requirements for the concept based on the theory above and the insights on transfer and design in chapter 5 and 6. Next, we will derive the requirements to the documentation of the thinkLet. This will offer the basis for the conceptual design of the thinkLet.

Conceptual requirements
Based on the insights on cognitive load and the material required for the training of practitioners and for the design by Collaboration Engineers we derived a large set of requirements for a pattern language for Collaboration Engineering. A pattern language offers a set of design patterns that can be combined and instantiated into a collaboration process prescription (Alexander, 1979). We call our design patterns ThinkLets (Vreede et al., 2006a). They should serve two purposes; they should support practitioners in executing the collaboration process and they should support the designer in choosing and combining activities to create an efficacious and

acceptable design that is reusable and transferable to the practitioner and has predictable results.

The thinkLets concept thus has requirements derived from each of the dimensions of 'quality of design' as listed in table 7.1

Transferability	***Efficaciousness & Acceptance***	***Predictability & Reusability***
Parsimonious instructions; script, supporting layout	Selection guidance: combinations, alternatives, when to use, classifications	Rigorous and precise documentation format for thinkLets including required resources
Identification, memory support	Based on intrinsic quality concepts such as equity and democracy	Based on patterns in best practices of experts, empirical testing and with theoretical understanding
What-will-happen description; enabling anticipation for far transfer	Ability to make small modifications for specific effects. Key interventions harvested from literature and best practices	Documentation of the outcome of (combinations of) thinkLets

Table 7.1. Requirements to the thinkLet concept.

To support the design of a collaboration process prescription that will perform high on the last two quality dimensions; predictability and reusability, we need to further develop the structure of the thinkLet concept based on the design pattern concept described above. For this purpose we need to further define the requirements to the documentation of the thinkLets.

ThinkLet documentation requirements

Besides information that needs to be documented in the collaboration process prescription template presented in chapter 5, we need to derive an overview of the information that should be captured for the different activities in the collaboration process design; in the thinkLets. As a basis for this we will use the design patterns concept to combine problem and solution in one thinkLet and the parameterized action representation described above.

Gamma et al.'s patterns are documented in terms of 12 aspects that bear great similarity to Alexander's architectural patterns and that we can use as a basis to derive the thinkLet design patterns. Table 7.2 compares the two pattern documentation styles.

Several aspects for identification are mentioned in both pattern conceptualizations. For the thinkLet concept we find the identification and the ability to support memorization of the thinkLets very important as it supports the creation of a shared language and helps to build richer schema that are easier to retrieve.

Next, the solution in design patterns is offered in thinkLets as the set of rules that creates the pattern of collaboration intended. Besides these rules we offer the script to communicate these rules.

ThinkLets, like design patterns are classified and contain several information elements that are used to support the designer in combining patterns, in selecting the right pattern and in instantiating the pattern. For this purpose the pattern offers an overview of related patterns to support choice (alternatives) and combination (precedents and successors).

Alexander's pattern	***Gamma et al.'s pattern***	***Purpose***
Name	Name Also known as	Identification, memorization, communication, training
Picture	–	Memorization, training
Problem	Intend	Goal description, what is the purpose of this pattern, what problem does it solve?
Solution	Consequences	Description of result of pattern execution (goal achievement and trade offs)
Solution	Structure	Documentation of comparable predictable, reusable solution that can be used for design
Hierarchy	Classification	Classification
Relations with other patterns	Related patterns	To inform possible combinations, to support choice among patterns
Context description	Motivation Applicability	Describing situations in which to use the pattern, putting the solution in perspective
Example	Sample code Known users	Illustration on different application domains, further explanation and visualization of solution
_	Collaborations	The ways participants interact in the pattern
_	Participants	Classes/objects or people participating in the pattern
_	Implementation	Guide and support in implementation

Table 7.2. Design pattern conceptualization.

Last, the patterns offer information that describes the situation in which the solution offered can be applied. This information and also the example will support the choice among similar patterns. The design patterns of Gamma et al. offer information to support the implementation of the process; participants and the way they interact. Such information could also help collaboration engineers to instantiate patterns to transfer them to practitioners.

Besides the problem and the rules to invoke the solution, more information is required to support both the practitioner and the collaboration engineering when using the thinkLets to design and execute the collaboration process prescription. In order to derive this information we used several information sources and data sets.

First we asked facilitators in a questionnaire (see textbox 4.4 in chapter 4 for method) whether they identify (name, number) and document the techniques they use. 36% of the facilitators (novices, experienced and expert facilitators) named or numbered their techniques and 59% documented their techniques. Facilitators use different documentation approaches. We can classify these approaches under three intentions; the intention to structure/order the techniques, to create a database, the intention to share/teach techniques and the intention to create a personal documentation as a reference and archive.

Besides their documentation approach, we also asked the facilitators what they document about a facilitation technique. We made the following aggregation to summarize the aspects documented by facilitators about the techniques they use.

- Name
- Objective, goal and deliverables
- Steps, script, instructions, procedures, activities, techniques and assignments
- "How to" or method
- Conclusion, experience, group reaction and effect
- Pros and cons, notes, tips, pitfalls and challenges
- Time frame
- Tools, material, supplies, resource requirements and set-up of the technique
- When to use, the phase, task, context and situations
- Group size
- Adaptation, variations, changes, modified forms and improvements
- How to do it for a novice
- Roles
- Narrative and examples
- Assumptions
- Resistance, group demographics, culture, difficulties

These aspects will be accommodated in the documentation format of the thinkLets.

With these conceptual guidelines the thinkLet concept now offers information for the practitioner to increase transferability, and information for the collaboration engineer to support selection, combination and choice of thinkLets and thus support efficaciousness. Furthermore, thinkLets are based on best practices that have intrinsic quality to increase acceptance. We offered guidelines to precisely document the instructions in the thinkLet and specified that the capabilities required need to be documented to improve reusability. Last, we specified that the effect of the thinkLet needs to be documented to increase predictability. With these guidelines we will conceptualize the thinkLet. Additionally, we will discuss the ability to make small modifications to thinkLets to evoke specific effects. After presenting the thinkLet conceptualization we will reflect on the other two approaches to improve the predictability of thinkLets; theoretical understanding of the thinkLets and empirical research to test their effect.

Based on the original thinkLet concept, these guidelines and the process prescription template described in section 5.3.2 we will now present the thinkLet concept.

7.2.2 Conceptual design of the thinkLet concept

The thinkLet concept was invented in 2001 by Briggs and de Vreede (Briggs and Vreede, 2001, Briggs et al., 2001). In this concept thinkLets were presented as a combination of a specific tool, its configuration and a script to use it. The documentation of a thinkLet was tool specific; it was a description on how to use GroupSystems™ software and contained instructions on when to choose the thinkLet, what to prepare, what to do, what to say, and a section with insights, a success story and an explanation of the mnemonic behind the thinkLet name. Over 70 thinkLets were documented in this style and they have been used since.

With the search on patterns in the use of thinkLets, insights were derived that led to a conceptualization that was more tool independent, had several levels of hierarchy, and was based on the pattern concept (Kolfschoten et al., 2004a, Kolfschoten et al., 2004b, Kolfschoten et al., 2006a, Vreede et al., 2006a).

Based on this initial concept we will further extent the conceptualization to support the different quality dimensions. There are three class diagrams of the thinkLet concept; the entire master thinkLet that is used as a complete documentation template of each thinkLet and two separate first level instantiations, the design pattern, containing all information required to support the design of a collaboration process with thinkLets and the instantiation template, used to document the thinkLet script for the practitioner.

The master thinkLet Class

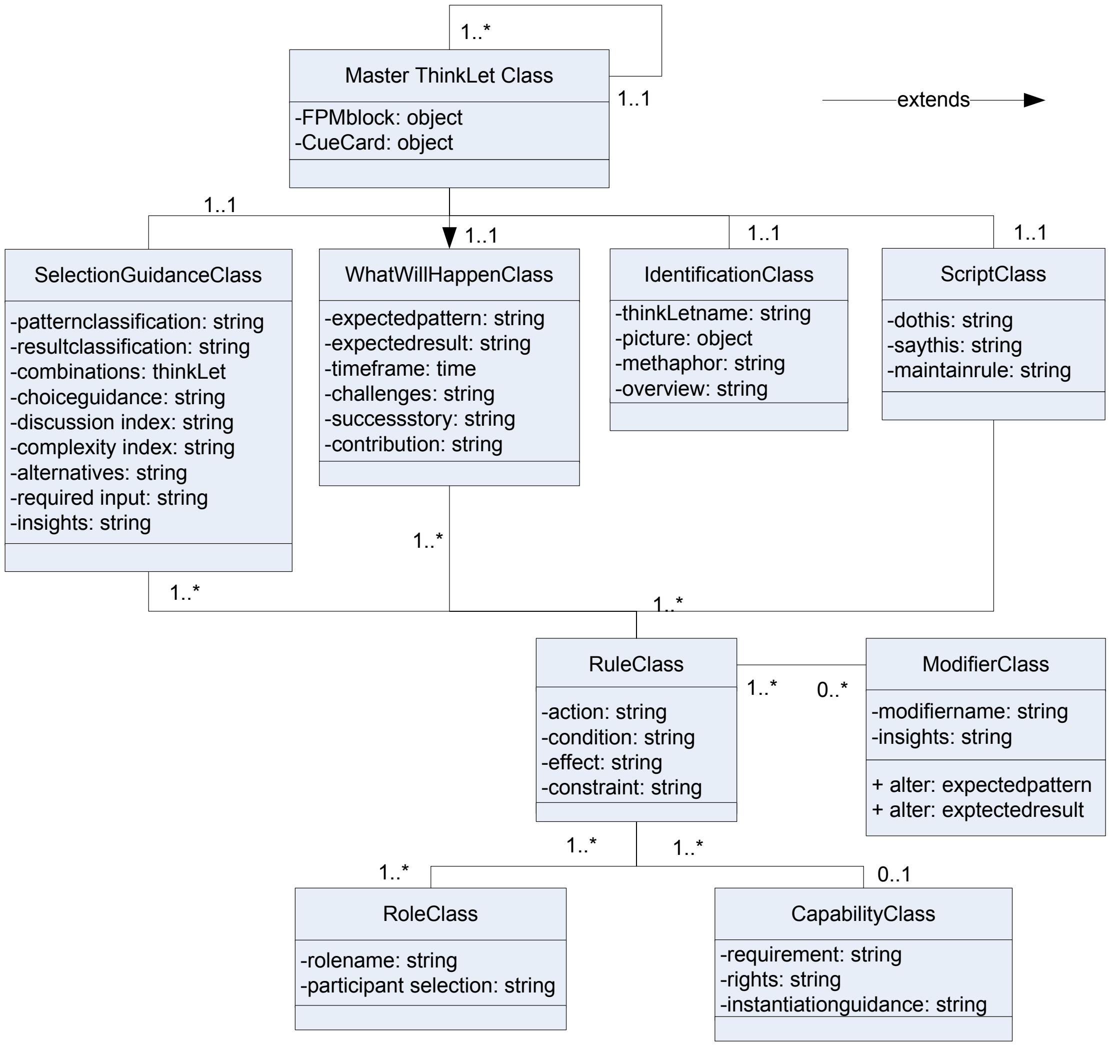


Figure 7.4. Master thinkLet conceptualization.

In figure 7.4 the UML Class Diagram model of the thinkLet concept is depicted. Below we will explain each element of this so-called master thinkLet. The master thinkLet describes everything that needs to be documented for any thinkLet. Below the design pattern and the instantiation template are depicted. The design pattern supports the collaboration engineer in creating the collaboration process sequence. The instantiation template will offer a default script for each thinkLet that can be edited to create the process prescription for the practitioner.

In the Master thinkLet, the rule class and the related classes contain the rule based interventions that form the basis of the thinkLet. These rules are extended in the script, the what-will-happen and the selection guidance.

The Master ThinkLet class: ThinkLets should offer the most parsimonious set of rules required to create a predictable pattern of collaboration and a specific type of result. The parsimonious set of rules is stripped of anything that could be removed without harming the pattern of collaboration or the result created. However, thinkLets still can be used to create different lower level patterns. For instance, the OnePage thinkLet, where a group brainstorms contributions on a single topic and in parallel to the same page, can be used to generate creative solutions (creativity pattern), to gather information for a more thorough analysis of the situation at hand (gather pattern). Each of these lower level patterns are sub-patterns of the higher level 'generate' pattern in which participants move from having less to having more contributions on the topic at hand (Briggs et al., 2006b). The pattern of collaboration a thinkLet produces depends on the modifiers used and their instantiation. Modifiers (see below) fine-tune the effect created by thinkLets. This effect can be either with respect to the result or with respect to the pattern of collaboration. ThinkLets can consist of a basic thinkLets composed of one or more rules and of modifiers. Furthermore, compound thinkLets can be built out of a combination of thinkLets and modifiers to create more complex methods. For instance in some techniques it is useful to perform several modifications or evaluations in a sequence, while making only one would also create a specific pattern of collaboration and result. An example is to rephrase (clarify) and classify (organize) each item in a list during a discussion. It is less efficient to first clarify all items, and then organize them; therefore these two rules can be integrated into one thinkLet.

We will now discuss each element of the thinkLet in detail. Each element has an icon. In the different thinkLet descriptions these icons can be used to quickly find a specific thinkLet element, they can be used as buttons or hyperlinks in a database to quickly find the thinkLet information one is looking for. The icons can be found in appendix 6. In appendix 9 an example of a master thinkLet is presented.

The Identification class: The first attribute is intended to give the thinkLet an identity. It should have a catchy name that can serve as a mnemonic and this metaphor should be briefly explained so it will support the memorization of the thinkLet. Additionally, a picture or icon that visualizes the mnemonic will support recognition and memorization of the thinkLet. Last, a short overview of the process should be included.

The Rule class: The rules in the thinkLet are the essential instruction- and adjustment interventions required to create the intended pattern of collaboration and result. The rules specify what participants should do (action) under which constraints and with which effect. Some rules are conditional, meaning that they need to be executed only when certain conditions apply. The effects of thinkLets are its result and the pattern of collaboration it evokes in the group. Furthermore, rules contain capabilities that need to be instantiated for its use. Last, a rule is specified for a specific role. The role and capabilities are described as separate classes.

The Role class: a role is an actor that is assigned to a collection of rules that guide the actions of this actor. The roles defined in the thinkLet have a name and a description that can serve as a basis for the instruction to the practitioner on how to select people to fulfill the role. This can be simple; assign people with this job description to that role, or it can require a more complex selection, based on stakeholder or skill and personality aspects of the participants.

The Capability class: a capability is a requirement to the physical workspace of the group that is needed to perform the action describe in the rule. The capabilities required are specified in terms of requirements and rights. Requirements can be a page to add contributions to, an input device such as a pen or a keyboard, discriminator such as a voting mechanism, dots or makers etc. rights can be rights to view, modify, move, etc. These capabilities can then be instantiated using software or manual tools such as pens and paper. For each capability the collaboration engineer should either specify a tool (or a few to choose from) that can be used or he should offer guidelines on how to instantiate the capability.

The Modifier class: Modifiers are named changes-of-rules or additional rules that can be applied to one or more thinkLets to create a predictable variation of the expected pattern of collaboration or the expected result the thinkLets evokes. For modifiers we specify the (change of) rule. Furthermore, we document the insights behind the effect of this (change of) rule.

We analyzed more than a dozen generation techniques from the thinkLet library and examined their differences on a rule basis (Kolfschoten and Santanen, 2007). Based on this analysis we could distil four basic thinkLets and twelve modifiers that could be applied to create variations on one or more of the basic thinkLets. Modifiers do not seem to alter the general pattern of collaboration in the thinkLet. However, some modifiers add a pattern, e.g. the qualitative evaluation modifier for generate thinkLets, as the name indicates, adds an evaluative character to the pattern of collaboration. Other modifiers alter the type of content that is created, for instance the comparative generation modifier increases the generation of contributions that excel on a specific aspect. To summarize, modifiers have one or more of the following characteristics:

- They can add a new rule to the thinkLet.
- They can alter a rule in the thinkLet.
- They can alter the pattern of collaboration among the participants.
- They can alter the nature of the result of the group effort.

Therefore, we define a *modifier* as a repeatable variation to create a predictable change in the pattern of collaboration or the result that a thinkLets produces. This ability offers the collaboration engineer more flexibility in designing precise interventions in collaboration processes that can be transferred to practitioners.

The Script class: The script contains a set of things for a practitioner to do and say to invoke the rules of the thinkLet. The "do this" section of the script is to be performed to execute the thinkLet. In the "say this" section the instructions to the participants are spelled out. A last section is a reminder to the practitioner on which rules to maintain. Mostly these rules are also indicated in the say this section, but the rules in this particular section offer a practitioner a basis for monitoring the group effort. When one of the rules is broken, they need to be re-enforced to prevent the group from diverging from the intended result and intended pattern of collaboration.

The WhatWillHappen class: In the "what-will-happen" document the effect of the thinkLet is described. The section pattern of collaboration explains the primary pattern and secondary patterns. The patterns should not just be classified, it should also be explained what will happen in the group, and what the participants will do. Next, the expected result should be specified. The type of result and detailing adjectives, structure adjectives and qualifications of the group's perception that characterize the result should be specified here. A next section is the timeframe. This will be difficult to specify as a design guideline, as it depends on several characteristics of the task and the group, of which size (task size and group size) is the most important. Still offering a bad guess of the time required with limitations is better than offering none. A collaboration engineer should offer a practitioner a specified timeframe for each step, or a calculator that enables the practitioner to calculate the time required based on several characteristics of the specific instantiation. Next, each thinkLet has several known challenges. These should be specified combined with a solution that will help a practitioner when he encounters the challenge. The next section of this document specifies a success story example of the thinkLet; a situation in which it was successfully used. The last section is the contribution section. In this section the collaboration engineer explains the practitioner why this thinkLet is important to achieve the group goal; this will support the practitioner when challenged by the group about the rationale for choosing this activity.

The SelectionGuidance class: The selection guidance section offers information that will support the collaboration engineer in his design effort, particularly in the choice and decomposition phase. First there is a classification of the thinkLet with respect to the pattern of collaboration it creates and when applicable, secondary patterns it instantiates. Second, there is a classification of the thinkLet result. Next, possible combinations with other thinkLets are indicated. There are successor combinations and predecessor combinations. Furthermore, the modifiers that can be applied to create variations on the thinkLet are indicated in this section. To be able to use thinkLets individually, or to choose the first thinkLet in a process, the required input should also be specified; not all thinkLets can be used from scratch (without input). The next section is the insights section in which each rule is explained in detail, and where the effect of that rule is elaborated upon. The section choice guidance explains in which situations the thinkLet can be applied. Alternatives, their purpose and difference are indicated and last, the thinkLet is characterized with a discussion index and complexity index that can be used for the validation of the thinkLet choice and particularly the fit with the practitioner and the group.

Next, we will discuss the two types of thinkLets that extent several of the classes described above; the thinkLet design pattern and the ThinkLet instantiation template.

The thinkLet instantiation template
The instantiation template is used by the collaboration engineer to create the documented process prescription for the practitioner. It extends the script, identification and what-will-happen document. Besides this script document, the instantiation is also used for the cue card and the block of the FPM.

The CueCard: The cue card offers a summary of the thinkLet. It contains the thinkLet name, the overview, a summary of the script, the challenges, the contribution, and the facilitation process model block described below.

The FPMblock: The FPMblock is the representation of the thinkLet in the facilitation Process model and is used to support recognition of the activity in the script document and on the cue cards. The FPMblock describes the activity, the thinkLet name, the timeframe, the sequence number, the pattern of collaboration the result and the duration. Furthermore, it contains a small version of the picture.

The instantiation template can be edited to instantiate the thinkLet for the specific situation. In the script the constraints, capabilities and roles should be instantiated and the 'do this' instructions should be adjusted to accommodate the specific instantiation of the capabilities. A rule "add contributions to the page" can be instantiated as write your requirements on post-its and add them to the wall. Or "click on the add button in the GSS tool, write down your solution and click on submit." Also the what-will-happen document should be instantiated so it specifies in more specific terms what the effect of the thinkLet is in the specific context of the collaboration process.

Instantiation can be supported with the use of a documentation system that contains a library of thinkLets, possibly in different languages. In this library a sequence of thinkLets can be selected (possibly with the choice support tool described above), and next the aspects that require instantiation can be filled in, while other sections can be edited. A sophisticated system would enable the specification of a constraint or capability for several thinkLets and instantiate it in every instance where it occurs. For instance, if a process is designed to brainstorm and choose requirements, several thinkLets will have 'requirements' as instantiation for the constraint "contribution specification", which is used throughout the script. A smart system would require the collaboration engineer to specify this only once. The model of the instantiation template of a thinkLet is displayed in figure 7.5.

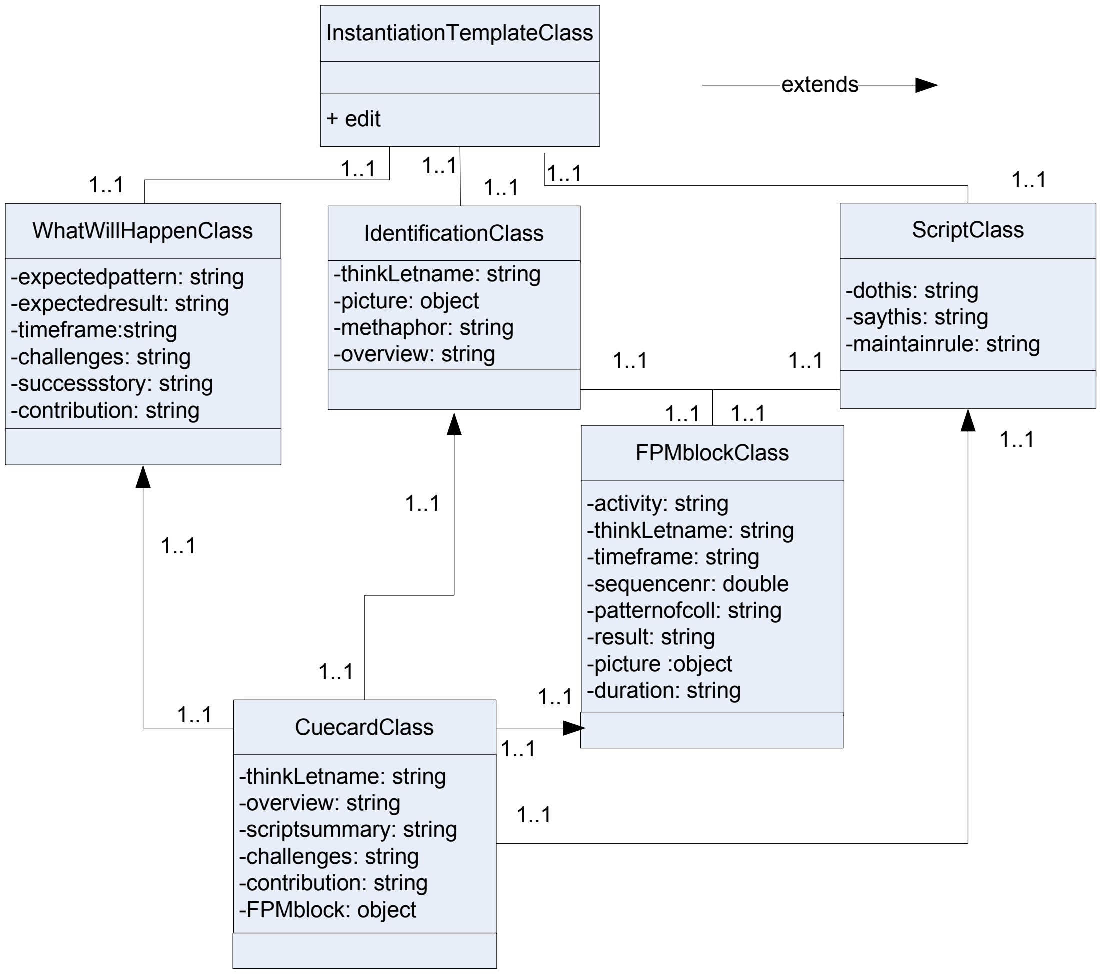


Figure 7.5. Instantiation template for the thinkLet.

The thinkLet design pattern

The thinkLet design pattern is used by the collaboration engineer to design the sequence of activities for each collaboration process. It contains the class selection guidance, what-will-happen and the identification class. The insights section in the selection guidance extends the rule concept. The design pattern thinkLet supports the choice action, based on the selection guidance it should offer an overview of possible thinkLets for a specific situation and it should give insight in the tradeoffs that remain for this choice.

The thinkLet design patterns can be collected in a database that functions as an expert system. A user can select different characteristics of a situation and based on these characteristics the system eliminates possible thinkLets. The remaining thinkLets will be displayed by the system, and the tradeoffs between these options will be highlighted. This will support collaboration engineers to select among thinkLets in the database. The design pattern is depicted in figure 7.6

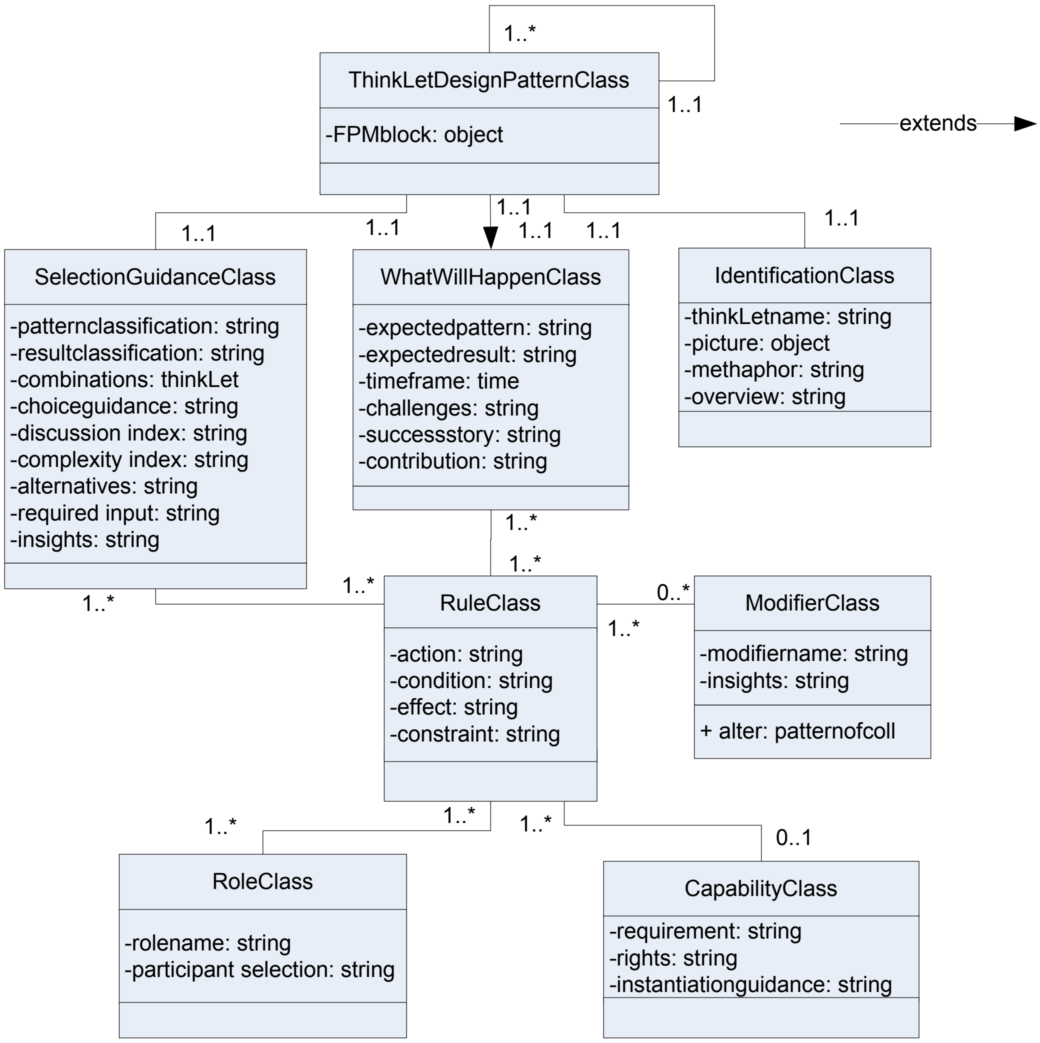


Figure 7.6. ThinkLet design pattern.

7.3 ThinkLet content: patterns of collaboration

Having determined the thinkLet documentation requirements based on experience and the rule and pattern concept, we need one last basis for the thinkLet conceptualization: the content of the patterns. In chapter 6 we introduced an overview of the patterns of collaboration. Each of these patterns can be evoked by an instruction intervention. The literature analysis used to create this overview is made in collaboration with Dean, Lowry and Kamal (Kolfschoten et al., working paper).

Generate

The generate pattern is defined as moving from having fewer to having more concepts in the pool of concepts shared by a group (Briggs et al., 2006b). There are three types of generation; creativity, gathering and reflection. Creativity or ideation is by far the most researched form of generation. Most ideation literature is based on the work of Osborn who described four rules for brainstorming (Osborn, 1953). However, especially in computer supported ideation many conflicting outcomes were found with respect to these rules. Recent work by Santanen (Kolfschoten and Santanen, 2007, Santanen, 2005, Santanen et al., 2004), and Briggs and Reinig (Briggs and

Reinig, 2007) addressed these conflicting findings and offered new insights in successful ideation, focusing not only on idea quantity but also creativity, uniqueness and quality. Gathering or elaboration is the second form of the generate pattern. This subpattern has the objective to accumulate relevant information—as opposed to finding new or unique solutions. In this subpattern the amount of information is less important than getting the relevant information required for the purpose of the decision at hand. A third subpattern is reflecting, in which the group generates insight in the relative value of concepts shared by the group. This last subpattern is also a subpattern of evaluate.

Reduce

The reduce pattern of collaboration deals with moving from having many concepts to a focus on fewer concepts that a group deems worthy of further attention (Briggs et al., 2006b). A first approach to reduction is to filter information based on a criterion. This can be achieved either by removing what is not wanted or selecting what is wanted (Belkin and Croft, 1992). A second approach to reduction is a form of summarization. Rather than selection, the aim of this approach is not to select part of the information, but rather to capture the essence of information with less information elements. There are again several approaches to summarizing; selecting unique information, merging similar contributions or selecting an instance of similar contributions to represent multiple instances. A third approach to reduction is to reduce information through abstraction. The purpose of abstraction is to make content more intellectually manageable by allowing group members to pay attention to relevant information and to ignore other details. Smith et al. (Smith et al., 1977) describe two approaches for abstraction: generalization and aggregation. Generalization refers to an abstraction in which a set of similar objects is regarded as a generic object. Aggregation refers to an abstraction in which a relationship between objects is regarded as a higher-level object. In such an abstraction many details of the relationship may be ignored.

Clarify

The clarify pattern of collaboration deals with moving from having less to having more shared understanding of concepts, words, problems, and possible solutions (Briggs et al., 2006b). The clarify pattern can best be understood in terms of two subpatterns: development of shared understanding and sense making. Mulder, Swaak, and Kessels (Mulder et al., 2002) explain the notion of shared understanding as implying mutual knowledge, mutual beliefs, and mutual assumptions. Groups achieve shared understanding when the group comes to a common understanding of concepts and words that are used for the task at hand. Sense-making usually requires some development of shared understanding of concepts and terms but also includes the development of a common understanding of the problem, the context of the problem, and the possible actions the group might take to solve the problem. In other words, sense making is done to prepare the group to act in a principled, and informed manner (Ntuen et al., 2005). Sense-making tasks often involve searching for ideas and input made by others in a group that are relevant for the purpose at hand and then extracting and reformulating the information so that it can be used (Weick, 1995).

Organize

The organize pattern involves moving from less to more understanding of the relationships among concepts the group is considering (Briggs et al., 2006b). When a group organizes the relations between a set of concepts, they develop an understanding of the relationships among the concepts. They consider possible relationships among concepts, and determine which relationships exists among which concepts. Categorization, sometimes referred to as classification, is the most common form of organize, and represents the basic cognitive process of arranging into classes or categories. For example, Chen et al (Chen et al., 1994) provided an artificial intelligence algorithm that tried to classify the most relevant ideas in a brainstorming session. The second organize subpattern pertains to developing sequence relationships among concepts; in other words, the sequencing of information and tasks. This pattern of collaboration is generally found in workflow management (Gronemann et al., 1999), collaborative scheduling (Chan et al., 1999), collaborative project planning (Romano et al., 2002), collaborative process planning (Kempenaers et al., 1996) and collaborative process modeling (Vreede, 1997). A third subpattern in organizing covers the causal relations between concepts. Causal modeling in collaborative setting is researched in Group Model Building literature (Rouwette et al., 2002) and in the soft systems methodology (Checkland, 1981) and in strategy making based on strategic options development and analysis (Eden, 1992, Ackermann and Eden, 2005).

Evaluate

The evaluate pattern involves movement from less to more understanding of the relative value of the concepts under consideration (Briggs et al., 2006b). The two purposes for evaluation are to support decision making and to support group communication (Cheng and Deek, 2007). Evaluation can be done through a voting or rating mechanism as well as through evaluative dialog (feedback, reflective discussion or qualitative evaluation). Voting and rating are the key approaches to numerical or quantitative evaluation. Both voting and rating include two key steps: collecting individual preferences and aggregating these into some form of group preference (Levin and Nalebuff, 1995, Gavish and Gerdes, 1997). Evaluation is commonly used to support communication as a means of surfacing preferences, assumptions, agreement, and disagreements, or to make an assessment or estimation of specific values or characteristics of an object under consideration. Voting allows groups to explore the reasons for the agreement and disagreements while working towards some ultimate decision. In some cases evaluation is used for choice or decision making. In such cases aggregation methods are used to combine individual preferences into a group result. Many aggregation methods are found in the literature (e.g., Levin and Nalebuff, 1995, Cheng and Deek, 2007, Gavish and Gerdes, 1997, Balthazard et al., 1998). The various methods fall into one of the following three categories: (1) majority rule, (2) consensus rule, or (3) a selection based on expertise. With these aggregation methods groups can share preferences either to make a rational choice or to build consensus (Balthazard et al., 1998). Besides voting, qualitative evaluation can be performed by having group members textually comment on the desirability of different options. In effect, group members state pros and cons of different potential solutions. This pattern is different from the gathering pattern in the sense that the gathering pattern is used for knowledge construction or sharing while qualitative evaluation is focused on reflection (Lowry et al., 2004) or illumination and verification (Shah and Vargas-Hernandez, 2003). Reflecting is not about eliciting preference directly, but rather about explaining specific qualities of the concepts under consideration as a basis for a numerical evaluation method, (mostly) rational choice, such as evaluating the feasibility of solutions. This sub-pattern is also a sub-pattern in of generate.

Consensus building

Consensus is usually defined as an agreement, acceptance, or lack of disagreement or some other indication that stakeholders commit to a proposal (Briggs et al., 2005). Methods to achieve consensus are either through the aggregation of preferences as in social choice or through resolving different aspects of disagreement or conflict through negotiation, creating shared understanding or through the inclusion or exclusion of additional factors/stakes. According to Briggs et al. there are five types of disagreement; differences of information where people possess different non overlapping information and therefore have different perceptions to a proposal, differences of meaning where people attribute different meaning to the same label, differences of mental models where chains of cause and effect are perceived differently leading to different expected outcomes of a cause of action, differences of goals, where strategic problems occur and the difficult to resolve differences of taste (Briggs et al., 2005). Resolutions to consensus building often involve methods to created more shared understanding and supporting multiple perspectives, but can also involve negotiation strategies and communication of preferences. We distinguish two patterns in consensus building, one focusing on building agreement, and one focusing on building commitment.

Based on the original definitions of the patterns of collaboration (Briggs et al., 2006b), their use in literature described above (Kolfschoten et al., working paper), and the current thinkLet set (Briggs and Vreede, 2001), we derived a key instruction intervention for each pattern of collaboration, displayed in table 7.3. This overview offers a basic framework to build a complete set of facilitation interventions. Besides these interventions rules should be added to build thinkLets that offer support for a specific pattern of collaboration. Additional rules can be used to accommodate different characteristics of the specific task and group.

Pattern	Sub-patterns	Pattern definition	Intervention
Generate	Creativity	Move from having fewer to having more new concepts in the pool of concepts shared by the group	Participants add ideas within the scope of the topic
	Gathering	Move from having fewer to having more complete and relevant information shared by the group	Participants add information to describe the topic of interest
	Reflecting (idem evaluate)	More from less to more understanding of the relative value or quality of a property or characteristic of a concept shared by the group	Participants assess the value or quality of a property or characteristic of a concept shared by the group

Table 7.3. (continued) Patterns of collaboration and interventions to create them.

Pattern	Sub-patterns	Pattern definition	Intervention
Reduce	Filtering	Move from having many concepts to fewer concepts that meet a specific criteria according to the group members	Derive criteria and let participant choose concepts from the set that meet these criteria
	Summarizing	Move from having many concepts to having a focus on fewer concepts that represent the knowledge shared by group members	Distill the key concepts that represent the information generated or remove Information that overlaps or that is redundant
	Abstracting	Move from having many detailed concepts to fewer concepts that reduce complexity	Identify more abstract concepts that are useful for the group' purposes. Test whether these encompasses the information under consideration and meet the objectives of the group
Clarify	Sense making	Move from having less to having more shared meaning of context, and possible actions in order to support principled, informed action	Discuss the context and possible actions that can meet the group objective
	Building shared understanding	Move from having less to more shared understanding of the concepts shared by the group and the words and phrases used to express them.	Discuss different meanings of concepts and words to achieve shared meaning
Organize	Categorizing	Move from less to more understanding of the categorical relationships among concepts the group is considering	Participants identify categories and categorize concepts among the categories or let participants cluster concepts and label the clusters
	Sequencing	Move from less to more understanding of the sequential relationships among concepts the group is considering	Participants discuss to determine the sequential order of concepts and visualize the resulting sequence
	Causal decomposition	Move from less to more understanding of the causal relationships among concepts the group is considering	Participants identify and visualize causal relations among the concepts

Table 7.3. (continued) Patterns of collaboration and interventions to create them.

Pattern	Sub-patterns	Pattern definition	Intervention
Evaluate	Choice: social/rational	Move from less to more understanding of the concept(s) most preferred by the group	Participants express their preference and aggregate a group vision, then discuss to negotiate a social choice or to determine the rational best choice
	Communication of preference	Move from less to more understanding of the perspective of participants with respect to the preference of concepts the group is considering	Participants express their preference and perspectives on the concepts under evaluation
	Reflecting (idem generate)	More from less to more understanding of the relative value or quality of a property or characteristic of a concept shared by the group	Participants assess the value or quality of a property or characteristic of a concept shared by the group
Consensus building	Building agreement	Move from less to more understanding of the difference in preference among participants with respect to concepts the group is considering	Participants express their preference and assess agreement and disagreements among the group, then attempt to resolve disagreements with respect to the outcomes
	Building commitment	Move from less to more understanding of the willingness to commit of participants with respect to proposals the group is considering	Participants express their willingness to commit to a proposal, negotiate, modify the proposal or argue to increase commitment for a (modified) proposal

Table 7.3. (continued) Patterns of collaboration and interventions to create them.

Based on this framework of basic interventions to create patterns of collaboration, variations can be made to design thinkLets that offer groups predictable and efficacious activities to achieve their goal. ThinkLet design as opposed to thinkLet "harvesting" is a new direction in the Collaboration Engineering research that requires further study (see chapter 9 on future research).

7.4 ThinkLet support for Collaboration Engineering

In this chapter we offered a pattern language and thinkLet conceptualization. Besides the conceptual support for predictability and reusability, thinkLets are the building blocks for design and transfer. ThinkLets support each of the five dimensions of quality of design. We will shortly discuss each of the quality dimensions.

Efficaciousness

Efficaciousness is the extent to which the design, when used as prescribed will focus the expense of resources to achieve the group goal

To focus expense of resources we need to select interventions that are instrumental to goal achievement. This is only possible when we can, to some extent, determine the outcome and effect of the interventions. ThinkLets are documented best practices that create a predictable pattern of collaboration and a result with predictable characteristics – generating ideas, reducing ideas, establishing shared understanding, organizing ideas, evaluating ideas, and building consensus around proposals. For each sub pattern of collaboration we described the key intervention required to create this effect. ThinkLets are classified among the patterns they create. Based on the documented effects of thinkLets, a sequence of thinkLets can be created that will move a group through a process toward achieving its goal. Furthermore, thinkLets have been used by many practitioners, students and facilitators with similar results. In combination with the choice support offered in the selection guidance class of the thinkLet, an efficacious collaboration process design can be made.

Acceptance
Acceptance is the extent to which the design when used as prescribed accommodates individual stakes sufficiently, to motivate stakeholders to commit the required resources for goal achievement.

Acceptance of a given thinkLet or sequence of thinkLets cannot be guaranteed. Acceptance depends heavily on whether thinkLets are deployed in a way that participants believe will help them to attain their goals. However, thinkLets are best practices that have been demonstrated to be acceptable in the field in a wide variety of circumstances. It can be asserted that they are at least not inherently unacceptable to many groups. Any special challenges or risks with respect to acceptance posed by a thinkLet are recorded in its documentation, along with guidance about how to mitigate those risks. These risks may include explanations of circumstances under which participants might question the use of the thinkLet or in which they might tend to refuse to participate, or to reject the results. Furthermore, thinkLets have been derived from experiences with GSS. The characteristics of GSS described in chapter 1 resolve several challenges in group work, especially challenges related to acceptance and participation of stakeholders.

Reusability
Reusability is the extent to which the design can be used successfully in different instances of the task

ThinkLets, like other design patterns, can be used in a variety of circumstances. They are documented in a way that a collaboration engineer can implement them with different technology or tools, in different domains and with different types of groups. Most thinkLets can be performed with both pen and paper. Some require data processing capacity as offered in GSS. Many thinkLets can be executed more efficiently with the use of GSS. Each thinkLet has a number of constraints that can be instantiated at process-design time or at execution time, to customize the thinkLet for a specific task in a specific domain. ThinkLets mostly define one participant role, but can be modified to accommodate different roles. Last, thinkLets can be modified or instantiated to fit different time constraints within some range. These features enable collaboration engineers to create a reusable process, as they support accommodating the available resources, while at the same time offering the flexibility required to accommodate changes in the available resources amongst different instances of the recurring task.

Transferability
Transferability is the 'ease of training' and the 'ease of execution' from the perspective of the practitioner

A large number of facilitators, students and practitioners have been trained to use thinkLets in collaborative effort. ThinkLets are easy to learn because their documentation is structured to contain the essential information (parsimony) thus limiting their complexity (intrinsic cognitive load) to a minimum. Furthermore, they have mnemonics to make it easier to memorize them and to use them as a shared language in communities of practice. Last, thinkLets are described as problem–solution combinations and therewith enable the building of high quality schema in memory as explained in chapter 5. Ease of execution is supported by a parsimonious instruction to the practitioner containing all essential rules that need to be used to instruct and adjust the behavior of the group.

Predictability
Predictability is the extent to which the design, when used as prescribed, creates a process and results as intended by the collaboration engineer.

Facilitators, collaboration engineers, and practitioners have executed thinkLets repeatedly in a variety of contexts over a number of years, and report that each execution produces a similar pattern of collaboration (e.g. generate, reduce, build consensus), and a similar result (Harder and Higley, 2004, Harder et al., 2005, Fruhling and Vreede, 2005, Vreede et al., 2006b, Bragge et al., 2005, Appelman and Driel, 2005, Acosta and Guerrero, 2006). Thus, thinkLets can be said to have predictable effects on group process and their outcomes, and these effects have been recorded in thinkLet documentation. Researchers have also verified these effects by reviewing the transcripts of hundreds of GSS sessions (Kolfschoten et al., 2004a). For some thinkLets, experimental research has been performed to compare their effects (Santanen et al., 2004). To further increase predictability, for some thinkLets theoretical models have been developed to understand their effects on the pattern of collaboration and results that are created when they are used (Briggs, 1994, Briggs et al., 2006a, Santanen et al., 2004). The reduction of errors between encoding and decoding through the use of parsimonious rules is likely to increase the chance on predictable group behavior and therewith predictable, outcomes (Santanen, 2005, Vreede et al., 2006a, Schank et al., 1993). The use of thinkLets will not remove the tradeoffs between the quality dimensions as described in chapter 3. However, it will support the design and transfer of the collaboration process and enable informed choices to improve the quality of the process design within the constraints of a specific Collaboration Engineering task.

In the test case study, described in the next chapter we will use the design approach and the design support concepts, the pattern language, the design template and the training approach, to create a collaboration process and transfer it to practitioners in the organization using the training approach.

Chapter 8. The effect of Collaboration Engineering on the quality of collaboration

In this chapter we present the case studies that are performed to evaluate the effect of the design and transfer of a collaboration process according to the Collaboration Engineering approach on the quality of collaboration. We will first present and explain the hypotheses of this study as introduced in chapter 4. Next, we will discuss the methods, measurement framework and the instruments used for the case study. Last, we will describe the two cases and the results.

8.1 Hypotheses

In this final case study we want to use the approaches and support developed in chapter 5-7 to evaluate the effect of the Collaboration Engineering approach on the quality of collaboration. In chapter 3 we defined quality of collaboration as the efficiency, effectiveness, productivity, commitment of resources and satisfaction with results and process. Based on this, we will examine the following hypotheses:

A practitioner who executes a collaboration process design created and transferred according to the Collaboration Engineering approach is not outperformed by a professional facilitator on:

a. satisfaction with the process
b. satisfaction with the results
c. commitment to the process
d. efficiency of the process
e. effectiveness of the process
f. productivity of the process

To accept these hypotheses the participant perception for a recurring collaborative task on each of these factors should not be significantly different in two treatments;

1. Process guidance by a practitioner
2. Process guidance by a professional facilitator

Besides collecting the data to confirm the hypotheses we need to collect data to be able to distinguish practitioners from professional facilitators. Furthermore, we want to know whether the practitioners felt supported by the training and collaboration process prescription (s)he received, and whether the process was executed as intended and resulted in predictable patterns of collaboration and results. In the next section we will discuss the research method and measurement framework for the study.

8.2 Research method

In the case studies the following steps of the Collaboration Engineering approach will be evaluated:

Design Approach and support:

- The collaboration process will be designed according to the approach and with use of the design support, presented in chapter 6.
- The collaboration process design will be documented according to the process prescription template described in chapter 5 and according to the thinkLet documentation framework described in chapter 7.
- Adjustments and improvements to the collaboration process prescription after the pilot will be documented

Pilot by professional facilitator
- The collaboration process prescription will be executed by one or more professional facilitators
- The participant's perception on quality of collaboration will be measured

Practitioner selection
- Practitioners will (when possible) be selected according to the criteria in chapter 2
- The facilitation/ group support experience of the practitioners will be assessed

Training
- The practitioners will be trained using the training approach in chapter 5
- The practitioner's perception of the transfer and supportiveness of the collaboration process prescription and training will be measured

Practitioner Session
- The practitioner interventions and the collaboration process will be compared with the intended interventions and effects as prescribed in the thinkLets
- The participants' perception on quality of collaboration will be measured
- The practitioners perception on his performance and the transferability of the collaboration process prescription will be evaluated

For this process the following research instruments will be developed and used:

1. A questionnaire to measure the participant's perception on quality of collaboration
2. A questionnaire to evaluate the initial experience of the practitioners with facilitation, GSS and group support.
3. A questionnaire to evaluate the practitioner's perception of the transfer and supportiveness of the collaboration process prescription and training
4. An observation protocol based on the script to compare the practitioner interventions and the collaboration process with the intended interventions and effects as prescribed in the thinkLets
5. An interview protocol to evaluate the practitioner's perception on his performance and the transferability of the collaboration process prescription

Participant's perception on quality of collaboration
The instrument we use to measure the hypotheses described above is a questionnaire on the participant's perception of quality of collaboration. This questionnaire measures 6 constructs; efficiency, effectiveness, productivity, commitment of

resources and satisfaction with results and process, defined in chapter 3. For each construct, five questions have been used with a 7 point Likert scale from 1 (strongly disagree), to 7 (strongly agree). The questions for satisfaction have been copied from (Briggs et al., 2003b). The others are developed in a similar manner based on the definitions in chapter 3. We used the questionnaire to evaluate a GSS workshop facilitated by the researcher in education setting. For each construct we calculated Cronbach's alpha. Cronbach's alpha should be higher than 0,6 for acceptable construct validity. The results are listed in table 8.1.

Construct	# Questions	# Respondents	Cronbach's α
Satisfaction with process	5	83	0,906
Satisfaction with result	5	83	0,936
Commitment of resources	5	83	0,856
Effectiveness	5	83	0,844
Efficiency	5	83	0,705
Productivity	5	83	0,875

Table 8.1. Construct validity quality of collaboration questionnaire.

The construct validity of efficiency is a lower than those of the other constructs. This can be explained by the fact that the efficiency is measured with respect to different types of resource (time efficiency, knowledge efficiency, effort efficiency and input efficiency in general). The complete questionnaire can be found in appendix 10a

Questionnaire for practitioner experience in group support

To evaluate the experience of the practitioner in group support we made a selection of the questions of the role interview protocol described in textbox 4.1 chapter 4. From this protocol we used only the questions that ask the respondents their experience in group support. The complete questionnaire can be found in appendix 10b

Questionnaire for training evaluation

To evaluate the training we used the same measurement framework as described in section 5.4.3, but worked it out in more detail, evaluating more detailed aspects of the process prescription and the training approach. The evaluation was done using a questionnaire. The complete questionnaire can be found in appendix 10c

Observation protocol

The observation protocol is used to identify where the practitioner deviated from the process prescription. It consists of the list of rules and instructions in the script that should have been communicated to the group. The researcher used this to observe the practitioner and documented deviations from the script to evaluate the session with the practitioner to understand when and why the practitioner deviated from the script. The protocol differs for each session.

Interview protocol for session evaluation

To evaluate the practitioner performance and the support of the Collaboration Engineering approach in transferring collaboration process prescriptions we evaluated the following constructs:

Predictability of the process prescription

- Were the session and the results as expected by the practitioner?
- Were there any deviations from the script and why?

Supportiveness of the process prescription

- Did the practitioner miss any information/training/support?
- Self-preparation effort besides the training

Difficulty of execution

- Was it difficult to execute the process?
- What would the practitioner do different next time?

Cognitive load of execution (numeric)

- Mental effort, difficulty, and effort of execution

The interview protocol can be found in appendix 10d

8.3 Transfer case 1: Half way project evaluation at a ministry

The collaborative task and context

In this first case the Collaboration Engineering task was a new task to the organization. The head of a corporate learning center of a ministry wanted to create a recurring process to let groups perform a mid - term reflection in which the group reveals mistakes, challenges and risks and transforms these to solutions and lessons learned, and in a next step to action points for which participants take responsibility. The session is a half day session. While project managers in the ministry do project evaluations, they usually do not perform such half- way evaluations, even through this point in time offers the opportunity to still learn from mistakes and to still resolve problems. The session is again supported with GSS and is designed for small project groups, usually between five and ten members.

The design of the collaboration process prescription

As the task was newly defined, there was limited information about the group and the task. The session was based on the idea that learning from mistakes will not only improve future project work, but will also create a culture in which mistakes are no longer hidden or covered up but openly discussed to quickly resolve them. The process was designed based on a regularly used sequence of thinkLets with the combination FreeBrainstorm and FastFocus, followed by a multi level LeafHopper brainstorm in which solutions/ lessons learned and actions are brainstormed and discussed. The thinkLet combinations used were preferred combinations and often used by the Collaboration Engineer (the researcher). The task was designed for GSS but contains thinkLets that can easily be performed with pen and paper. The session was piloted once with a previous version of the design and adjusted to the process described above. The pilot was facilitated by the researcher.

The pilot results

For the pilot session only a small number of participants were invited. Since some did not show, the final group (four participants) was very small. Groups this size suffer less from the problems in collaboration, especially in the case of a team without significant conflict, which was the case. Therefore, GSS support and facilitation for this group size is unlikely to offer large improvements. We did an informal evaluation of the process with the participants and they indicated several suggestions for improvement. A key suggestion for improvement was to change the rather negative tone of the process introduction, which we revised in the final process. The results of the pilot are not very positive. The process did not reveal very new issues and therefore result had limited impact. See table 8.2.

	Researcher		
Construct	n	μ	stdv
Satisfaction process	4	4.0	0.9
Satisfaction result	4	4.5	0.7
Commitment	4	5.1	1.0
Efficiency	4	4.4	1.2
Effectiveness	4	3.8	0.7
Productivity	4	4.0	1.2

Table 8.2. Quality of collaboration for the pilot session.

With the improvements discussed above we revised the process design and started recruiting practitioners. Unfortunately we did not get the chance to do a second pilot session based on the revised process design.

The practitioners

In this case we had the opportunity to recruit practitioners based on a profile. While this profile was communicated to the potential practitioners, no selection was made among the participants. The profile was:

- Affinity with groups
- communicative
- analytical
- flexible and self conscious
- medium computer skills required

While most practitioner had affinity with project management, the practitioners did not have to perform the session as part of their formal job description. Nine practitioners were trained. The average age of the practitioners was 45. All practitioners were employed by the ministry. Most of the practitioners, except for 2 had a senior function and most of the practitioners had a (project) management role. Experience with groups varied from experience with facilitation of brainstorm and risk management sessions to training to supporting teambuilding and peer review. Only one practitioner had some experience with facilitation in combination with GSS support. Two practitioners indicated that they designed processes and that they created new techniques to support groups. All other practitioners supported workshops or group processes based on existing methods.

The training

7 practitioners handed in the training and manual evaluation. The result are displayed in the table 8.3

Question scale: 1-7	Average	Stdv
Was the manual complete? (very incomplete-very complete)	5.17	1.33
Was the material explained well? (very insufficient-very well)	5.71	0.76
Was the order of the training useful? (very insufficient-very well)	5.43	0.53
What did you think of the following aspects of the manual? (very un-useful -very useful)		
- Assumption document	6.00	0.63
- ThinkLets	6.17	0.75
- Identification	6.00	0.71
- Script	6.00	0.63
- What-will-happen	6.17	0.75
- Facilitation process model	6.00	0.63
- Cue cards	6.20	0.84
How would you judge the following aspects of the training? (very insufficient-very well)		
- Introduction of the task	5.17	0.75
- Introduction to collaboration and Collaboration Engineering	5.43	0.53
- Introduction of the process overview	5.29	0.76
- Explanation of the thinkLet concept and the structure of the manual	5.71	0.76
- Introduction of the thinkLets used	5.50	0.55
- Self-construction of the process	5.50	0.87
- Per step going through the script and presentation	5.86	1.07
- Per step exercise with the system with challenges based on case	5.50	0.87
- Per step discussion on what is difficult or can go wrong	5.43	0.79
- Per step looking at the cue cards	5.57	0.53
- Repeat difficult parts	5.17	0.75
- What if, discussion of doom scenario's in facilitation	5.33	0.52
How do you estimate the mental effort of preparation and training? (low-middle- high)	4.07	1.30
How difficult was the training? (low-middle- high)	2.93	1.30
How tiring was the training?	3.57	1.72
How difficult was the training compared to other trainings you did within the context of your profession? (low-middle- high)	2.57	1.13
Do you feel equipped to facilitate the session? (very insufficient-very well)	4.86	0.69
Did the training meet your expectations? (not at all-very)	5.00	1.29
Were you satisfied about the training? (not at all-very)	5.57	0.98
Were you satisfied if you compare the training to other trainings that you followed during your professional career? (not at all-very)	5.14	1.35
Did you enjoy the training? (not at all-very)	5.57	0.79

Table 8.3. Evaluation of the training and process prescription.

Remarks on the training concerned two key concerns: First, the time to practice with the GSS software was too limited; participants felt they needed more support in using the technology. Second, the practitioners had very different backgrounds; therefore some practitioners wanted more information and background while others found the information sufficient. Both the process prescription and the training were evaluated rather positive, and the training was considered not very difficult or tiring. Practitioners felt equipped to facilitate the session and mostly indicated that they wanted a bit more time to get acquainted with the technology. The training met the expectations and was enjoyable.

The practitioner performance

The first practitioner prepared the session well, re-read the process prescription and adjusted the presentation, and practiced with the system. The practitioners spend about two and a half hours on this. The practitioner indicated that (s)he did not miss any training, support or information. The results are displayed in table 8.4.

Construct	*Average*	*Stdv*	*n*
Satisfaction process	4.05	1.04	8
Satisfaction outcome	3.50	0.82	8
Commitment	5.18	0.98	8
Efficiency	4.55	0.99	8
Effectiveness	3.28	0.85	8
Productivity	4.03	1.10	8

Table 8.4. Results practitioner 1 case 1.

The session went as expected but the quality and quantity of the results were slightly disappointing. This was mainly due to the fact that a large conflict was revealed during the session, in which emotions and discussion were challenging to deal with. The practitioner resolved this problem, and some level of consensus was achieved. However, some participants remained unhappy with the outcomes of the meeting. The practitioner reported that it was not difficult to facilitate the session, although (s)he reported a very high mental effort due to the conflict that had to be resolved. The difficulty was rated average but the session was also rated rather tiring. The practitioner found the results of reasonable quality and did think (s)he would be able to teach others to facilitate this session.

The second practitioner ran a more successful session. The participants had no large conflicts and were therefore much more satisfied with the results. The practitioner followed the process prescription and did not miss significant information, but indicated that (s)he could have prepared better. The practitioner spent two hours on the preparation, to study the manual and to read some material provided by the problem owner. The researcher did not make many interventions, except for one case where the practitioner wanted to alter the script, and by asking for clarification during the reduction activity.

Construct	*Average*	*Stdv*	*n*
Satisfaction process	5.68	0.80	5
Satisfaction outcome	5.40	0.76	5
Commitment	5.80	0.87	5
Efficiency	5.64	1.11	5
Effectiveness	5.12	0.67	5
Productivity	5.36	1.04	5

Table 8.5. Results practitioner 2 case 1.

The practitioner reported a low mental effort, difficulty and did not find the execution task tiring. The practitioner indicated that the results were of sufficient quality and that he would be able to teach someone else to run the session.

The third practitioner did not prepare for the facilitation task, and therefore, the researcher had to intervene several times to give instructions to the group. A challenge in this session was that some participants indicated that they were not able to give much input given the scope of the meeting. The results are presented in table 8.6.

Construct	Average	Stdv	n
Satisfaction process	5.09	0.60	9
Satisfaction outcome	4.84	0.52	9
Commitment	4.87	1.01	9
Efficiency	5.02	0.72	9
Effectiveness	4.47	0.69	9
Productivity	4.89	0.65	9

Table 8.6. Results practitioner 3 case 1.

In the results we found a slightly lower commitment level. Results and effectiveness were therefore rated lower than some other sessions. The practitioner was well able to lead the discussions, but it was difficult to get the group moving, given their limited interest in the outcomes of the meeting. The practitioner indicated that he did not miss support or information and only technical training. The practitioner reported a high metal effort and above average level for difficulty and tiring. The practitioner indicated that he would not be able to train others to run this session.

The last practitioner had to execute the session unexpected, the practitioner planned to be the chauffeur for this session, but the practitioner who should have taken run the process could not come. Therefore the practitioner had only one hour to prepare and re-read the manual. However, the practitioner did observe the two previous sessions. The group size was small, and the practitioner knew the group members. The results listed in table 8.7 are slightly positive, however, there was some variation in the results.

Construct	Average	Stdv	n
Satisfaction process	4.52	1.12	5
Satisfaction outcome	4.36	1.08	5
Commitment	5.12	0.93	5
Efficiency	5.08	0.76	5
Effectiveness	4.60	0.82	5
Productivity	4.68	0.80	5

Table 8.7. Results practitioner 4 case 1.

The practitioner indicated that a manual just for the technology would have been useful, no other information or training was lacking. The practitioner reported just above average mental effort and difficulty and did not find the session very tiring. The practitioner indicated that (s)he would be able to train others to run the session.

Conclusions for the quality of collaboration hypotheses

We compared the results from the practitioners with the results of the facilitators using an independent-samples t-test. The assumptions for a t-test are the following:

- There is a continuous scale used for each dependent variable: we used a 1-7 Likert scale.
- Random sampling: this assumption is not met, while we did not choose the participants, they all work for government organizations in the Netherlands, mostly for the specific ministry for which this Collaboration Engineering case was designed. Also the practitioners have been selected based the practitioner profile discussed above.
- Independence of observation: this assumption is violated as well, as participants collaborated in groups during the sessions. Statistics manuals suggest the use of a more stringent alpha. We therefore used a smaller α value; .01.
- Normal distribution: for a sample size of 30+ violation should not pose a problem. In this case we could not get this sample size for both groups and we only had the opportunity to do one very small pilot session. We therefore used a smaller α value; .01.
- Homogeneity of variance: We used Levene's test for equity of variances, this assumption was not violated.

The groups we compared are the professional facilitator's n=4 and the practitioner's n=27. (See table 8.8) We found that for all quality dimensions there was no significant difference between practitioners and facilitators (α =0.1). Also the effect size eta squared was calculated. According to Cohen (Cohen, 1988) this is a very small effect, less than 4% of the effects is explained by the difference between facilitators and practitioners.

Construct	Sig. α 0.1	Effect size
Satisfaction process	0.111	0.0362
Satisfaction outcome	0.937	0.0088
Commitment	0.854	0.0005
Efficiency	0.143	0.0305
Effectiveness	0.304	0.0150
Productivity	0.137	0.0314

Table 8.8. Independent-samples t-test practitioners vs. facilitators.

Limitations

Striking in this case is the low number of sessions both for the pilot and the number of practitioners that actually implemented the process. A key reason for this is that the process describes a new task that could be used on a recurring basis but that is not embedded in any approach, and for which no managerial incentives are implemented. While the project initiator got several positive reactions and the practitioners all recognized the added value of the process it took some time before the first practitioners performed their session. (4 months). To further deploy the collaborative task it would be good to install some management incentives and to include the process as a standard method in the portfolio of project leaders.

8.4 Transfer case 2: Integrity assessment in government organizations

The collaborative task and context

This second case study was performed for a small Dutch government agency which is one of five future centers in the Netherlands. The mission of this agency is to promote and support alternative methods for group work such as creativity methods and GSS support. The agency was hired by another government agency which is responsible for supporting the government at large in creating, maintaining and using their integrity policy to facilitate integrity assessment workshops with GSS support. Since it is the mission of the second agency to have all government organization do the assessment, the future center needed more facilitators to support the recurring collaborative workshop. The integrity support agency and the future center embraced the Collaboration Engineering approach for two reasons: first because it needs to enlarge it capacity of ‘facilitators’ to run the assessment. Second, because they want to structure and standardize the workshop to ensure its quality, even when it is performed by a variety of practitioner. Furthermore, groups will feel more comfortable in an integrity assessment facilitated by a member of their own or a similar organization. The session is an integrity assessment of the organization, similar to a risk assessment but focused on possible integrity violations. The topic is possibly sensitive and the anonymity of GSS support is therefore used. The session takes a full day and contains mostly evaluation steps, both qualitative and quantitative. However, discussion is required to build consensus and to integrate brainstorming results to gain a group result.

The design of the collaboration process prescription

The process for the integrity assessment was already designed by both agencies and some external experts. For the Collaboration Engineering approach we modified only a few steps to slightly simplify the process and to avoid unpredictable outcomes of some of the steps. Furthermore, some of the constraints in the thinkLets were slightly changed to clarify the instruction and the intended result. To make these modifications, the researcher observed two facilitators from the future center while they executed the process, and discussed proposed changes with both the integrity support agency and the future center. Next, the researcher determined the thinkLets used in the process and documented the collaboration process according to the collaboration process prescription template in chapter 5. To validate the resulting process design it was discussed again with the facilitators from the future center and a pilot session based on the new process prescription was facilitated by the researcher.

The pilot results

Both the researcher and the facilitators of the future center have facilitated many sessions with a variety of organizations. All facilitators charge a fee for the sessions they facilitate, and facilitate in service of clients of the organization for which they work, and thus can be regarded as professional facilitators. Each facilitator roughly performed the same process as described in the process prescription with only marginal differences in thinkLets used and instructions to the group. The results are presented in table 8.9. The differences between the performances of the facilitators are marginal and the standard deviations are not very high either. We will use these results as a benchmark to assess the practitioners' performance.

Construct	Facilitator 1 future center			Facilitator 2 future center			Researcher			Combined results		
	n	μ	stdv	n	μ	stdv	n	μ	stdv	n	μ	stdv
Satisfaction process	12	5,3	1,1	15	5,8	0,8	14	5,7	0,7	41	5,6	0,9
Satisfaction result	12	4,7	1,2	15	5,3	0,9	14	5,0	0,8	41	5,0	1,0
Commitment	12	5,7	0,8	15	5,7	1,0	14	5,7	0,9	41	5,7	0,9
Efficiency	12	5,1	1,0	15	5,7	0,9	14	5,7	0,8	41	5,5	1,0
Effectiveness	12	4,6	1,1	15	5,0	0,9	14	5,0	0,9	41	4,9	1,0
Productivity	12	5,0	0,8	15	5,6	0,8	14	5,3	0,9	41	5,3	0,9

Table 8.9. Quality of collaboration as a result of facilitation by professional facilitators. scale 1-7 , 1 being low, 7 being high.

The practitioners

The practitioners in the case were all employed by large government organizations. Some had a function related to integrity; others had affinity with (technical) facilitation. Again, none of the facilitators had to perform the session as part of their formal job description. Most of the practitioners had some experience in supporting groups, either in the role of trainer or as teacher, or project leader. Some facilitated workshops or worked as a technical facilitator but not for many sessions. Some worked as trainer. Most had an academic education level. The average age was 43, four were female, and three were male. The recruitment of practitioners could not be influenced by the researcher.

The training

Seven practitioners participated in two separate training sessions. Six handed in the evaluation of the training and the process prescription. The results are listed in table 8.10. The manual was considered complete; all aspects were considered useful. The material was explained well and the order of the training was useful. Each aspect of the training was rated sufficient, except the revision of the cue cards. This step was skipped in most cases as it was redundant with the discussion of the script. The manual was considered quite extensive, and some more organization of the different parts using tab sheets would have been useful. Most of the process steps were focused on the evaluation or assessment of an organization and since the trainees worked at different organizations, it was difficult to exercise or simulate these steps, which resulted in the fact that some steps could not be experienced. It was recommended by the practitioners to improve the training in this respect, but this will be difficult. Some practitioners had to opportunity to attend a session before they first executed it. The difficulty, tiring effect and mental effort of the training was estimated medium. Practitioners felt equipped to execute the session, but wanted to see a real session before they executed their own, when possible. The training was satisfying.

Question scale: 1-7	Average	Stdv
Was the manual complete? (very incomplete-very complete)	6.17	0.75
Was the material explained well? (very insufficient-very well)	5.50	0.55
Was the order of the training useful? (very insufficient-very well)	5.00	0.63
What did you think of the following aspects of the manual? (very un-useful -very useful)		
- Assumption document	5.00	1.26
- ThinkLets	4.50	1.76
- Identification	4.33	0.82
- Script	5.00	1.10
- What-will-happen	5.00	1.10
- Facilitation process model	5.00	0.89
- Cue cards	5.17	1.47
How would you judge the following aspects of the training? (very insufficient-very well)		
- Introduction of the task	5.00	0.89
- Introduction to collaboration and Collaboration Engineering	5.17	0.75
- Introduction of the process overview	5.17	0.75
- Explanation of the thinkLet concept and the structure of the manual	5.33	0.52
- Introduction of the thinkLets used	5.17	0.75
- Self-construction of the process	5.17	0.75
- Per step going through the script and presentation	5.67	0.52
- Per step exercise with the system with challenges based on case	4.50	1.38
- Per step discussion on what is difficult or can go wrong	5.17	1.17
- Per step looking at the cue cards	3.67	1.21
- Background on the subject of the session	4.50	1.38
- Repeat difficult parts	4.83	1.33
- What if, discussion of doom scenario's in facilitation	5.33	0.82
How do you estimate the mental effort of preparation and training? (low-middle- high)	4.33	1.37
How difficult was the training? (low-middle- high)	4.00	1.41
How tiring was the training?	4.33	1.63
How difficult was the training compared to other trainings you did within the context of your profession? (low-middle- high)	3.83	1.47
Do you feel equipped to facilitate the session? (very insufficient-very well)	4.33	1.03
Did the training meet your expectations? (not at all-very)	4.50	1.05
Were you satisfied about the training? (not at all-very)	5.00	0.63
Were you satisfied if you compare the training to other trainings that you followed during your professional career? (not at all-very)	4.83	0.75
Did you enjoy the training? (not at all-very)	5.33	0.82

Table 8.10. Evaluation of the training and process prescription.

The practitioner performance

Four practitioners executed the process. The researcher observed the sessions and intervened only when required. In one session the researcher was replaced with someone else. The researcher or replacing observer also operated the 'leader station' of the GSS. The practitioners reported back on several questions through writing or interview. The observer made notes about deviations from the script and interventions that were made to support the group that should have been made by the practitioner.

The first practitioner did not prepare the first session very well. The time between the training and the first session was almost three months and the practitioner did not observe a session of another practitioner or facilitator. To support the group the practitioner relied heavily on the sheets provided with the instructions to the group. The practitioner did not use the cue cards, but did look at the process prescription every now and then, especially at the facilitation process model. Most instructions were understood by the group, but one instruction was unclear and did not directly result in action and had to be clarified by the researcher. The group came in late which required the practitioner to speed up the process. Other than the increased speed there were no deviations from the script. The researcher had the role of observer and technical facilitator. The practitioner did not monitor the input of the group and therefore it was difficult to stimulate the group to improve the quality of the input. Also making choices with the group and presenting the voting results was difficult. In these steps the researcher had to step in and intervene to guard the quality of the session.

However, except for one instruction, the practitioner got the group to perform the required activities. The results are displayed in table 8.11.

Construct	Average	Stdv	n
Satisfaction Process	5.4	0.7	13
Satisfaction Result	5.0	0.9	13
Commitment	5.7	0.8	13
Efficiency	5.6	0.9	13
Effectiveness	4.8	0.9	13
Productivity	5.2	0.7	13

Table 8.11. Practitioner 1 case 2.

The participants were more satisfied with the process and sufficiently satisfied with the results. They were committed to the process and found the process efficient. Effectiveness and productivity scored slightly lower. The practitioner indicated the mental effort of facilitation slightly higher than average, and indicated that the task was not very difficult and not tiring. The execution task went as expected, and the practitioner was satisfied with the result. The practitioner indicated to be able to train another practitioner to execute the process. As indicated the practitioner did not prepare well and did indicate that this was needed. Furthermore, the practitioner would have liked to do a try-out.

The second practitioner prepared better and spent about 3 hours on reading about the organization that was participating in the session and revising the script. Furthermore, (s)he observed a session run by another facilitator. The practitioner indicated that the session went as expected, except for two things, the cultural differences; the group came from a different (Belgium) country than the practitioner, and the fact that the organization just had a re-organization, which caused some stress and uncertainty about tasks, roles and positions. This caused some additional challenges in focusing the discussion. The results of the session are listed below (table 8.12). The researcher only intervened to clarify a few of the instructions, and to encourage the group to improve the quality of their contributions.

Construct	Average	Stdv	n
Satisfaction process	5.67	0.67	11
Satisfaction outcome	5.22	0.79	11
Commitment	5.82	1.02	11
Efficiency	5.69	0.94	11
Effectiveness	5.05	0.87	11
Productivity	5.36	0.97	11

Table 8.12. Practitioner 2 case 2.

The practitioner reported that no support or information was missing except for the challenges with respect to culture and the re-organization. The practitioner reported that the session was challenging, and therefore not 'easy' but not very difficult and tiring, and reported a just above average mental effort. The practitioner indicated that (s)he would be able to train others to run the session if (s)he would have some more experience.

The third practitioner also prepared well; re-read the manual and the sheets and the cue cards and spend about 4 hours on this. The third practitioner also observed the session performed by another practitioner. The practitioner indicated that the session went as expected, but the group seemed to expect more from the session than what was accomplished. The group liked the discussions in the session but some resisted the 'assessment elements' in the session. This posed some challenges. The results are listed in table 8.13.

Construct	Average	Stdv	n
Satisfaction process	4.94	1.33	10
Satisfaction outcome	4.68	1.22	10
Commitment	5.34	1.42	10
Efficiency	5.18	1.34	10
Effectiveness	4.30	1.04	10
Productivity	4.86	1.43	10

Table 8.13. Practitioner 3 case 2.

The practitioner indicated that the execution of the script was not very difficult, only to determine when to end a discussion was considered difficult. (s)he indicated a slightly above average mental effort and average difficulty and tiring level. The practitioner did not miss training or support or information only some experience and also indicated that (s)he forgot a few small details of the instructions, especially with respect to the GSS used. The practitioner indicated that (s)he would be able to train others to execute the session if (s)he gained some more experience.

The fourth practitioner also prepared for about 4 additional hours but did not have the opportunity to observe a session run by someone else. The practitioner revised the manual and practiced the instructions. The session went as expected. The technical facilitator had to intervene at a few points to clarify especially the use of the system and to clarify some instructions based on questions of the participants. The practitioner indicated that (s)he did not miss support. The practitioner would have liked some more training to learn how to set the right ‘mood’ in the group in the introduction, and found the information provided a bit too extensive. The results are listed in table 8.14.

Construct	Average	Stdv	n
Satisfaction process	5.62	0.83	12
Satisfaction outcome	5.30	1.12	12
Commitment	5.95	0.79	12
Efficiency	5.48	0.85	12
Effectiveness	5.30	1.05	12
Productivity	5.42	1.05	12

Table.8.14. Practitioner 4 case 2.

The practitioner indicated that it was not difficult, but challenging to run the session. (s)he indicated a rather high mental effort and found it slightly more tiring than average and reported average difficulty. The practitioner indicated that (s)he would not be able to teach others to run this session as (s)he had no affinity with the technology used.

Conclusions for the quality of collaboration hypothesis

We compared the results from the practitioners with the results of the facilitators using an independent-samples t-test. The assumptions for a t-test are the following:

- There is a continuous scale used for each dependent variable: we used a 1-7 Likert scale.
- Random sampling: this assumption is not met, while we did not choose the participants, they all work for government organizations in the Netherlands or Belgium. Also some practitioners have been selected based on their background on integrity.
- Independence of observation: this assumption is violated as well, as participants collaborated in groups during the sessions. Statistics manuals suggest the use of a more stringent alpha. Therefore we used an alpha of .01.
- Normal distribution: for a sample size of 30+ violation should not pose a problem, still we used the same significance level as in case 1; .01.
- Homogeneity of variance: We used Levene’s test for equity of variances, this assumption was not violated.

The groups we compared are the professional facilitator’s n = 50 participants and the practitioner’s n = 45 participants. (see table 8.15).

Construct	Sig. α 0.01	Effect size
Satisfaction process	0.800	0.0009
Satisfaction outcome	0.191	0.0236
Commitment	0.863	0.0004
Efficiency	0.980	0.0009
Effectiveness	0.365	0.0114
Productivity	0.762	0.0013

Table 8.15. Independent-samples t-test practitioners vs. facilitators.

We found that for all quality dimensions there was no significant difference between practitioners and facilitators (α = 0.01). Also the effect size eta squared was calculated. According to Cohen (Cohen, 1988) this is a very small effect, less than 3% of the effects is explained by the difference between facilitators and practitioners

Limitations

A key limitation in this research is the observing role of the researcher. As the sessions are held in commercial setting the researcher cannot allow the session to go wrong entirely, and thus, when practitioner mal-performs, the researcher has to intervene. As we tried to limit this as much as possible, the interventions as reported will have had an effect on the quality ratings. Another limitation is that while the task is identical, the groups are not and due to the sensitive topic of this case, some sessions can be significantly more difficult than others. This poses a limitation to the comparisons across sessions. A last limitation is the relatively low number of practitioners and professional facilitators.

8.5 Conclusions from the case studies

In the first case study all practitioners ran the entire process and created the intended results. However, the case had a number of limitations including the limited number of facilitators, practitioners, and especially the limited number of participants in the groups. Furthermore, a difficult aspect of the session was that it could reveal and highlight conflicts and therefore was more challenging to support by the practitioners. Measured results varied regarding effectiveness, and satisfaction with outcomes. This is mainly due to the fact that the issues and problems that were revealed in some cases generated the need for additional work while most groups coped already with a workload problem. The practitioners dealt with these conflicts by themselves.

The second case was more successful. The results of this case were in general more positive, both for facilitators and practitioners. There were more opportunities to run this session with both professional facilitators and practitioners as the groups were larger in number and size. This enabled us to make a better comparison. Again practitioners could make most improvement with respect to the outcomes of the sessions. Supporting the group to create high quality results is very difficult without a frame of reference with respect to the quality of the outcome. When practitioners execute the session for the first time, it is therefore difficult to manage the quality of the outcomes. The practitioners that observed a session executed by another practitioner emphasized more rules related to the quality of outcome.

Some interesting observations were made. One practitioner in case 2 did not prepare the execution and therefore presented the group with the instructions and background of the session by more or less ‘reading the slides out loud’. Although the participants noticed this, they were not disappointed in the results and generally satisfied with the process. This indicates that the transferability of the instructions has become very substantial. A second observation is that a practitioner in case 1 managed to resolve a very tense conflict in the group. While this practitioner indicated that this was demanding in terms of mental effort, the conflict was resolved. Over all sessions it was observed that ratings of mental effort increased if the practitioner or facilitator had to deal with conflict in the group. Finally, it seemed that part of the differences in the results from case 1 and case 2 can be explained by the nature of the session. The collaborative task in case 1 can result in more variation with respect to the instrumentality of the results for the participants and the group. Case 2 has an outcome that is in most cases instrumental for the organization, while it is generally not very instrumental to the participants, except when it enables the participants to reveal significant problems in which they are stakeholder. In both sessions we see the tradeoffs reflected in the design, but the emphasis of the session design was focused on different quality dimensions. It would be interesting to study patterns in the relation between quality of design and quality of collaboration.

Both cases showed that there was no significant difference between facilitators and practitioners. In case 1 we violated an important assumption for the t-test. Therefore we must conclude that the results are encouraging, but a larger amount of sessions in both cases, and several additional different cases would help to further confirm this conclusion.

Chapter 9. Engineerability and transferability

The aim of this research was to offer a theoretical foundation for the design and transfer of collaboration processes according to the Collaboration Engineering method to gain further understanding of the quality of a collaboration process design. The objective of this research was to identify, define, operationalize and test the quality factors of collaboration process design to create collaboration process prescriptions that can be transferred to practitioners in an organization. In chapter 2 we formulated the following research questions:

1. *What are the quality dimensions of a collaboration process design that is transferred as collaboration process prescription to be executed by a practitioner, and how can we define these quality dimensions?*
2. *Knowing these dimensions, how can they be operationalized to optimize the quality of the design, to increase practitioner performance and therewith the success of the collaborative effort?*
3. *Does the use of the design and transfer support indeed enable the support of a collaboration process by practitioners with professional quality?*

We have presented the quality dimensions of 'collaboration' and collaboration process design in chapter 3. They are listed in table 9.1

Quality of collaboration process design	*Quality of collaboration*
Efficaciousness	Satisfaction with process and results
Acceptance	Commitment
Transferability	Efficiency
Reusability	Effectiveness
Predictability	Productivity

Table 9.1. Quality dimensions of collaboration process design and collaboration.

Next, we operationalized the dimensions of 'quality of collaboration process design' in chapter 5, 6 and 7. In chapter 5 we offered a transfer approach consisting of a training approach, and a supporting process prescription template to support transferability. In chapter 6 we offered a design approach and supporting models to support efficaciousness and acceptance. In chapter 7 we further conceptualized the thinkLet concept to support efficaciousness, acceptance, transferability, predictability, and reusability.

In chapter 8 we evaluated the design and transfer support and found that it enabled practitioners to achieve similar results as professional facilitators with respect to 'quality of collaboration'.

In the introduction we described the need for collaboration support, and the challenges with respect to the implementation of collaboration support. We explored the requirements and challenges of collaboration support from different perspectives. We identified two types of collaboration support; process support and technology support. This research focuses on process support. The thinkLet concept, transfer and design approach support both 'engineerability' and transferability and therewith offer a theoretical foundation for the implementation of the Collaboration Engineering approach. Furthermore, it offers a basis to offer design support for facilitators and training support for novice facilitators. In combination with technology support some rules of the thinkLets can be enforced. Furthermore, the pattern and result classification offer insights to the need for new (combinations of) capabilities, that can be implemented in technology support.

In this chapter we will further reflect on the two challenges we identified in chapter 2; the design challenge and the transfer challenge. The design challenge can be assessed by analyzing the 'engineerability' of collaboration. The approach to support the transfer of a collaboration process has been evaluated in the final case studies in chapter 8 and will be discussed to reflect on transferability. Besides these reflections we will summarize the implication of this work for research and practice and we will discuss limitations and further research.

9.1 Engineerability

In this thesis we looked at collaboration from an engineering perspective. An engineering perspective on collaboration would display collaboration as a input-process-output system, as visualized in figure 9.1.

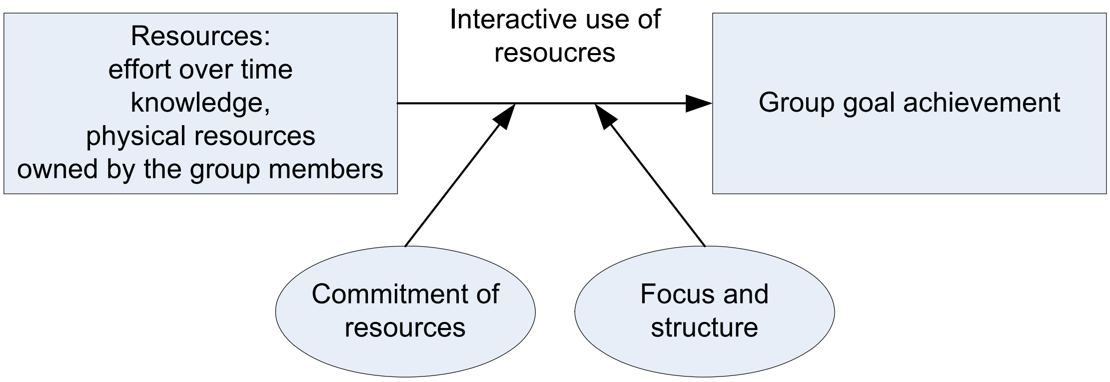


Figure 9.1. The engineering perspective on collaboration.

As discussed in chapter 3, the input of a collaboration process consists of the resources that we can use in this process. Resources in a collaboration process are availed and spend by the group members or participants of the process. The key resources in a collaborative activity are effort over time and knowledge. Furthermore, technology, tools and other physical resources such as money or a room can be used in the process. The key conditions that enable or thwart that the interactive use of these resources leads to group goal achievement, is that the required resources are committed to the process and that the use of these resources is structured or focused towards the group goal.

A second requirement to an engineering approach is that an engineering approach uses a structured method to create an object with certain specific qualities. In this research we propose an engineering approach for collaboration process design based on the dimensions of high quality collaboration and high quality collaboration process design.

A third condition for 'engineerability' of collaboration is the ability to predict the effect of the group interventions prescribed; the pattern of collaboration and the result. If we assume that interventions in collaboration processes have a random unpredictable and uncertain effect, we cannot design collaboration.

Summarized, for collaboration to be 'engineerable', there are three key requirements;

- The use of a systematic approach to the design of collaboration processes
- The ability to structure and focus joint effort towards a group goal
- The ability to evoke with some predictability the commitment of resources to a group goal

The Collaboration Engineering design approach is presented and evaluated in chapter 6, and in chapter 7 we presented the thinkLet concept in which we offer support for each of the five dimensions of high quality collaboration process design. Each of these dimensions contributes to the ability to increase focus and commitment:

Efficaciousness
If the design is efficacious it will support the group in achieving the goal. A highly efficacious design leads to goal achievement and therefore it must offer focus and structure. It will also increase the likelihood assessment of the participants with respect to goal achievement, and thus in case of goal congruence, it will increase commitment. ThinkLets have known effects (pattern, result) which, in combination with the design support will enable the design of an efficacious process.

Acceptance
If acceptance of the design is low, stakeholders will reject the process or intended outcomes and thus will not commit the resources required to achieve the goal. Acceptance thus has a direct effect on commitment. Acceptance can be related to the group goal (stakes are not sufficiently accommodated), but it can also be a direct result of the process (unfair or disrespectful process). Both can lead to a negative utility of participation, or to a low trust in the likelihood of goal achievement. ThinkLets accommodate characteristics that increase their acceptance, and the design approach supports the consideration and validation of acceptance challenges.

Reusability
Process design has to accommodate the requirements and constraints with respect to resources available in multiple instances of the recurring task. When these instances vary too much the design might not fit the resources available. Through the reusability of thinkLets and the flexibility in the instantiation level some variation in the recurring task can be accommodated, but there are boundaries to this. It is therefore very important to indicate these boundaries. Reusability also depends on the commitment of resources to the process. While the practitioner can agree with the initiator about a timeframe, participants and physical resources availed to the collaboration process, he cannot 'agree' on the effort made and the knowledge shared by the participants. This will depend once more on the willingness to commit resources before and during the process.

Transferability
The transferability is an enabler of successful intervention in the collaboration process. If we cannot train the practitioner to execute the process prescription as intended, its effects will be weakened or thwarted. Transferability is supported both by the thinkLet concept, the thinkLet documentation, the process prescription template and the training approach.

Predictability
Predictability is the key enabler of design or engineering. If the outcome of our interventions is random, we cannot design collaboration. Full predictability is however impossible, since the success of the collaborative effort depends on the willingness of participants to commit resources throughout the process. Predictability in this research has focused on the patterns of collaboration and the characteristics of the result, which help the collaboration engineer to estimate the efficaciousness of the design. Predictability of the other dimensions of high quality collaboration process design is partially supported. Predictability of acceptance is supported by the use of GSS characteristics that increase acceptance. To support predictability with respect to reusability and transferability, specifications in the thinkLet and guidelines in the design and transfer approach support the collaboration engineer to accommodate requirements with respect to the resources and the practitioners.

9.2 Transferability

After improving the transferability of the process prescription and positive evaluation of the training approach, we examined the ability to transfer the collaboration process design to practitioners to execute by them selves. The transfer cases in chapter 5 and chapter 8 showed that practitioners without any facilitation experience were able to execute the collaboration process design, and that most participants were satisfied with the results and the process, and judged it to be efficient, effective and productive. Furthermore, we found no significant differences between practitioners and professional facilitators, an encouraging conclusion. The mental effort reported by the practitioners varied. We think this effect was caused both by the group dynamics (conflict) and by the confidence of the practitioner. Further research is required to statistically confirm the transferability hypothesis. In this analysis the effect of group dynamics and self efficacy of the practitioner should be accommodated.

9.3 Implications for research

In this research we offered a theoretical basis for design and transfer challenges in Collaboration Engineering. The main research contribution is the theory in chapter 3, in which we explain quality of collaboration and the tradeoffs in design that need to be accommodated in order to create commitment and focus and therewith enable goal achievement in collaborative setting. The second contribution is the thinkLet conceptualization which does not only support the design and transfer of collaboration processes but also offers a framework for research on interventions in collaborative effort. Last, an implication for research is the cognitive load perspective that offers a theoretical basis for the transfer approach. We will explain the implications of each contribution.

We made three significant steps in our understanding of quality of collaboration. First, we presented the theory of collaboration in Chapter 3. Second, we have introduced a system perspective on collaboration which shows us its key input, enablers and output. This perspective allowed us to derive metrics for high quality collaboration; satisfaction with process, outcome, efficiency, effectiveness, productivity and commitment of resources. Third, the quality dimensions have been operationalized in a measurement instrument to evaluate the quality of collaboration which can be used in various collaborative settings within but also outside Collaboration Engineering research. The theory on collaboration we presented in chapter 3 needs further empirical research to test is falsifyability.

Quality tradeoffs in designing collaboration processes offer a useful basis for facilitators and collaboration engineers to consider the different implications of choices in collaboration process design. This framework can be used as a descriptive research framework to inform and analyze collaboration process design. Each dimension has been operationalized as a set of requirements to a high quality collaboration process design. The requirements are qualitative measures.

The second contribution to research is the thinkLet conceptualization. Through the framework for parsimonious rules to describe interventions in collaborative effort we can enable more precise comparison of collaboration support interventions and their effects. When comparing interventions defining the rules and thinkLets used, we can make a more specific comparison, than when 'with or without GSS' or 'with or without facilitation' are compared as treatments. This will enable further research in which we can compare specific interventions in collaboration in order to improve our understanding of efficient and effective collaboration support (Santanen, 2005, Kolfschoten and Houten, 2007, Kolfschoten et al., working paper, Kolfschoten and Santanen, 2007).

Last, we introduced the cognitive load theory from Sweller (Sweller, 1988, Sweller et al., 1998b) and colleagues as a basis for the transfer approach. The cognitive load theory offers us an instrument to understand the complexity of the practitioner task and to offer support to reduce this complexity and to enable efficient and effective transfer of the knowledge and skills required to support groups in achieving their goals. Cognitive Load theory has many more implications for our understanding of collaboration and collaboration support. This will be further discussed in the further research section.

9.4 Implications for practice

The practical contributions of this research include the approaches for design and transfer and the support for design and transfer, both based on the thinkLet concept. We will briefly address each.

The transfer approach is mainly a training approach. The key purpose of the training is to equip practitioners to run the collaboration process by themselves. The ability to get a group moving is critical and can vary amongst practitioners. While all practitioners that we trained in this research were able to perform the process interventions and achieve the result intended, results varied. It requires further research to determine the effect of our support in relation to the self-efficacy and

experiences of the practitioners. To support the transfer we offered the process prescription template. The template offers the practitioner all information required to instruct and support the group in achieving their goal. The different elements are defined based on cognitive load theory to limit and focus the cognitive effort required to learn and execute the collaboration process.
The design approach and design support (classifications and models) helped both collaboration engineers and facilitators in designing successful collaboration processes. The approach has not only been used in this research but also in various research projects in which (GSS supported) facilitation interventions for specific tasks and specific domains were designed (Acosta and Guerrero, 2006, Kamal et al., 2007, Bragge and Merisalo-Rantanen, 2005). Our evaluation of the design approach has shown that the approach was useful, easy to use, timesaving, and supportive in improving the design. Furthermore, users indicated that it did not need further improvement. A next step in the improvement of the design support is to capture the expert information offered in the pattern classifications, choice map, validation support and design guidelines and further developed in a Computer Aided Collaboration Engineering tool (CACE tool) This will enable more simple search for a specific thinkLet though the use of automated guidelines and choice support. The CACE tool could further enable the instantiation of a sequence of thinkLets to a thinkLet script and the instantiation of the thinkLets in a GSS tool. This will enable collaboration engineers and facilitators to quickly design and instantiate the extensive manual and the complex configuration of the GSS. The system can be further enhanced with different validation and evaluation methods to support validation of the process prescription and the performance of the facilitators and practitioners (Kolfschoten et al., 2007).

The thinkLet concept and the pattern language of thinkLets have been further developed to support both execution by the practitioner, and design by the collaboration engineer. Requirements from both perspectives have been accommodated to create a thinkLets that offers support to accommodate each of the five quality dimensions of design. The usefulness of the thinkLet concept can exceed the Collaboration Engineering domain; it will offer design and execution support for facilitators, and it can provide a basis for the development of a new generation of GSS application that accommodates a more complete scope of patterns in collaboration as presented in chapter 7.

9.5 Limitations and future research

Future research based on the theoretical foundations in this dissertation can be performed in various directions. We identified nine interesting directions for further research. Some of these directions have already been explored.

First, the Collaboration Engineering approach, presented in chapter 2 has three sub phases that were only marginally addressed in this research which are the deployment of the Collaboration Engineering work practice, and the closely related investment decision. A limitation in this research was the relative short timeframe for the case studies, and therefore the limited number of sessions that we were able to evaluate. Longitudinal studies are required to be able to fully confirm the effect of the Collaboration Engineering approach on sustainability, and to enable further confirmation of the results with respect to transferability. The cognitive load theory lens used to study transfer in combination with the value frequency model (Briggs, 2006) can help us to further predict the transfer and sustained implementation of collaboration support. A longitudinal study should analyze the learning curve of practitioners, the project management involved and the development of communities of practice around the Collaboration Engineering work practice. This study can be based on the principles of the capability maturity model as presented in (Santanen et al., 2006). Furthermore, the investment decision and the applicability of the Collaboration Engineering approach in various domains needs to be further analyzed. A limitation to this research is that it has been applied in two cases in a government setting (chapter 8) and one case in business setting (chapter 5). More variation in the application domain would enable further generalization on the applicability of the approach. One particularly interesting application domain is e-democracy and e-government (Kolfschoten, 2007).

Second, the thinkLet concept and the patterns of collaboration offer a basis for a new area of GSS and facilitation research. In this research the level of analysis was on a higher level; the effect of a collaboration process design on 'quality of collaboration.' The thinkLet concept described in chapter 7 is more detailed; it describes a specific intervention in collaboration processes. ThinkLets can be compared on their effects and the patterns they create. This enables research on a higher level of detail. The different patterns of collaboration offer a framework for the development of new GSS tools that offer a more complete set of capabilities to support each of the different sub-patterns (Kolfschoten et al., working paper). Furthermore, the rule concept enables us to make more specific comparisons between capabilities instantiated with the use of GSS tools and paper based collaboration support.

Third, a limitation is that the theory of quality of collaboration requires further evaluation to test its propositions. The questionnaire results described in chapter 8 will provide data to make a first analysis of the mediating role of commitment for the success of collaboration. Furthermore, the model should be extended to incorporate the effects of the 'quality of design dimensions' described above.

Fourth, the research did not directly evaluate the relation between 'quality of design and 'quality of collaboration' especially with respect to efficaciousness, re-usability and acceptance this would be interesting to study. For this purpose an instrument should be developed to evaluate the quality of the design besides its transferability. As a basis for this instrument the validation criteria described in section 6.3.1 can be used.

Fifth, the design approach can be further supported by building a Computer Aided Collaboration Engineering tool. (CACE tool) a conceptualization of this tool has been proposed (Kolfschoten et al., 2007) and in earlier research a prototype has been made (Kolfschoten and Veen, 2005). A CACE tool could offer support in the choice among thinkLets and in the instantiation of the process prescription. Furthermore, it could be used for validation of the design and evaluation of its implementation. A CACE tool could be combined with a GSS suite, in such way that the design created with the CACE tool can be imported in the GSS to directly instantiate capabilities with right tools and configurations. It would be useful if such tool instantiation could be saved as a stand-alone application for the specific Collaboration Engineering task.

Sixth, the thinkLet-rule concept will offer new insights in the structure of the current thinkLet concept. When all thinkLets are defined in terms of rules it may be found that many thinkLets are variations on other thinkLets, and this should enable us to distinguish basic thinkLets that create a specific pattern of collaboration and modifiers that create variations on this pattern of collaboration and variations in its outcome (Kolfschoten and Santanen, 2007). Besides modifiers, general design guidelines and their effect on the resulting collaboration process prescription should be captured and further research is required to determine their applicability and effect.

Seventh, the thinkLet conceptualization in which we prescribe precise capabilities required to create specific patterns of collaboration, will help us to compare the capabilities required in virtual, distributed and asynchronous settings to reduce the challenges of a lack of face to face contact. Further research is required to create a set of requirements to the capabilities in GSS that will enable facilitators to execute the thinkLets in a virtual environment.

Next, the theory on cognitive load could be used to further increase our understanding of patterns of collaboration. Heninger et al (2006) and Santanen (2005) both indicated the value of a cognitive perspective on patterns of collaboration, and the value of such perspective in understanding the effect of interventions in collaborative settings.

Finally, several challenges in the execution of the practitioner task of the facilitator remain, especially with respect to the adjustment interventions required to manage the quality of outcomes and decisions and to involve and gain commitment of stakeholders. Learning these tasks could be further supported with the use of video and/ or learning games in which these challenges are further illustrated.

Appendix 1. The task of the facilitator

In this appendix the tasks of a facilitator are listed, based on different sources in literature. In the second and third column we indicated whether the task is a design or execution task. When both columns are checked, the practitioner is supported in this task by the design. The execution tasks should be executed by the practitioner; the design tasks are tasks for the collaboration engineer.

Task	***Design***	***Execution***
Agreeing on 'quick' wins (Ackermann, 1996)	x	x
Asking difficult or sometimes obvious questions (Ackermann, 1996)	x	x
Being sensitive to the group (Vreede et al., 2002)	x	x
Being sensitive to the meeting content/topic (Vreede et al., 2002)	x	x
Brining the group to results/effectiveness (group) (Vreede et al., 2002)	x	x
Building rapport with problem owner (Vreede et al., 2002)	x	x
Confronting group regarding its process (Hayne, 1999)	x	x
Considering the actions in light of the responsibilities (Ackermann, 1996)	x	x
Content knowledge (Vreede et al., 2002)	x	x
Creates and reinforces an open, positive and participative environment (Clawson et al. 1993)	x	x
Creates comfort with and promotes understanding of the technology and tech. outputs (Clawson et al. 1993)	x	x
Detecting variance from structures (Hayne, 1999)	x	x
Enabling participants to contribute freely (Ackermann, 1996)	x	x
Enabling the group to concentrate on the task being addressed (Ackermann, 1996)	x	x
Encourages/supports multiple perspectives (Clawson et al. 1993)	x	x
Ensuring everyone has an opportunity to contribute to the discussion and decisions (Hayne, 1999)	x	x
Ensuring members identify and maintain a discussion focus and a procedure for that focus (Hayne, 1999)	x	x
Ensuring that participants perceive themselves to be equal for the event (Ackermann, 1997)	x	x
Equalizes participation of participants (Dickson et al. 1996)	x	x
Establishing a model of behavior (Hayne, 1999)	x	x
Explaining/resuming/interpreting group output and giving feedback (Vreede et al., 2002)	x	x
Giving free reign/tightening the reign (Vreede et al., 2002)	x	x
Guarding the discussion focus (Vreede et al., 2002)	x	x
Guides the agenda (Dickson et al. 1996)	x	x
Identifies decisions (Dickson et al. 1996)	x	x
Intervening when appropriate at level of group instead of individual (Hayne, 1999)	x	x
Introduction/explanation of GSS technology (Vreede et al., 2002)	x	x
Introduction/explanation of meeting process & rules (Vreede et al., 2002)	x	x
Introduction/explanation of meeting topic (Vreede et al., 2002)	x	x
Keeps discussions on topic (Dickson et al. 1996)	x	x
Keeps group outcome focused (Clawson et al. 1993)	x	x
Listens to, clarifies and integrates information (Clawson et al. 1993)	x	x
Making regular reviews of the material (Ackermann, 1996)	x	x

Task	*Design*	*Execution*
Managing the group's direction and progress (Ackermann, 1996)	x	x
Presents information to the group (Clawson et al. 1993)	x	x
Providing closure (Hayne, 1999)	x	x
Providing structure to focus group limits and boundaries (Hayne, 1999)	x	x
Sensitivity to time management (Hayne, 1999)	x	x
Solicits feedback (Dickson et al. 1996)	x	x
Stressing to the client the importance of implementing outcomes (Ackermann, 1996)	x	x
Structuring discussions (Vreede et al., 2002)	x	x
Task focus (Niederman et al. 1996)	x	x
Time management (balancing time and results) (Vreede et al., 2002)	x	x
Determining levels of consensus (Hayne, 1999)	x	x
Promulgating actions achieved (Ackermann, 1996)	x	x
Providing the client with some form of control (Ackermann, 1996)	x	x
Recognizing implicit vs. explicit decisions (Hayne, 1999)	x	x
Summarizes (Dickson et al. 1996)	x	x
Synthesizing information and building cognitive maps (Hayne, 1999)	x	x
Understanding the group and its objectives (Niederman et al. 1996)	x	x
Creating and displaying an overview of the issue/problem (Ackermann, 1996)	x	
Creating situations conductive to learning (Hayne, 1999)	x	
Develops and asks the right questions (Clawson et al. 1993)	x	
Ensuring that a match is made between the problem task and the facilitator's skills (Ackermann, 1996)	x	
Giving advice to the client concerning the potential dangers of participative methods (Ackermann, 1996)	x	
Knowledge of group processes/group dynamics (Vreede et al., 2002)	x	
Managing the process of review and control (Ackermann, 1996)	x	
Paying attention to group membership (Ackermann, 1996)	x	
Planning the meeting (Hayne, 1999)	x	
Plans and designs the meeting (Clawson et al. 1993)	x	
Preparation of script (Vreede et al., 2002)	x	
Providing a clear set of objectives and corresponding agenda (Ackermann, 1996)	x	
Providing an explanation of the process (Ackermann, 1996)	x	
Providing information on the benefits gained from participative methods (Ackermann, 1996)	x	
Providing the client with some control over the meeting (Ackermann, 1996)	x	
Putting aside time to review the outcomes (Ackermann, 1996)	x	
Recognizing stages of group process (Hayne, 1999)	x	
Selects and prepares appropriate technology (Clawson et al. 1993)	x	
Structure group activities (Dickson et al. 1996)	x	
Technical/GSS knowledge (Vreede et al., 2002)	x	
Understanding more about the organization (Ackermann, 1996)	x	
Understands technology and its capabilities (Clawson et al. 1993)	x	
Actively builds rapport and relationships (Clawson et al. 1993)		x
Being available/approachable (Vreede et al., 2002)		x
Choosing/preparing meeting accommodation (Vreede et al., 2002)		x

Task	Design	Execution
Clarifies and rephrases issues (Dickson et al. 1996)		x
Demonstrates flexibility (Clawson et al. 1993)		x
Demonstrates self-awareness and self-expression (Clawson et al. 1993)		x
Demonstrating flexibility (Hayne, 1999)		x
Directs and manages the meeting (Clawson et al. 1993)		x
Discussing the location of the workshop/meeting (Ackermann, 1996)		x
Ego-less facilitation (Niederman et al. 1996)		x
Exhibiting energy and enthusiasm (Ackermann, 1996)		x
Flexibility (Niederman et al. 1996)		x
Good communication skills (Niederman et al. 1996)		x
Identifies communication problems (Dickson et al. 1996)		x
Keeping the energy and enthusiasm alive (Ackermann, 1996)		x
Leadership (Niederman et al. 1996)		x
Leading the group and its discussion in general (group) (Vreede et al., 2002)		x
Maintaining awareness of own feelings as an indicator (Hayne, 1999)		x
Manages conflict (Dickson et al. 1996)		x
Manages conflict and negative emotions constructively (Clawson et al. 1993)		x
Managing group creativity, anxiety, and conflict (Hayne, 1999)		x
Motivating/stimulating group (meeting process) (Vreede et al., 2002)		x
Observing communication patterns (Hayne, 1999)		x
Process adaptivity (Vreede et al., 2002)		x
Promotes ownership and encourages group responsibility (Clawson et al. 1993)		x
Provides and aids the group’s emotional climate (Dickson et al. 1996)		x
Providing motivation (Hayne, 1999)		x
Reexamining agreed actions (Ackermann, 1996)		x
Reformulates questions or problems (Dickson et al. 1996)		x
Respecting the group results (Vreede et al., 2002)		x
Self projection (Vreede et al., 2002)		x
Social skills (Vreede et al., 2002)		x
Test agreements among participants (Dickson et al. 1996)		x
Understanding group values and providing new values in process (Hayne, 1999)		x

Appendix 2. Quality of collaboration

This table lists the factors used to describe quality of collaboration in the literature. Similar factors are listed in the same row. The final column describes the generic quality dimensions.

(Fjermestad and Hiltz, 1999) 230 articles	*(McLeod, 1992)* 13 studies	*(Baltes et al., 2002)* 27 studies	*(Hwang, 1998)* 28 studies	*(Tyran & Shepherd, 1998)* 12 studies	*(Dennis et al., 2001)* 61 studies	*Generic quality dimensions of collaboration*
efficiency						efficiency
decision time	time to decision	time to decision				
number of decision cycles						
time spent in activities			speed		time	
time spent waiting for responses						
time to consensus						
				perceived distraction		
effectiveness		effectiveness				effectiveness
communication			communication			
number of comments					number of ideas	
idea quality			quality of outcome			
decision quality	decision quality				decision quality	
decision confidence						
process quality						
creativity/innovation						
level of understanding						
task focus	task focus					
depth of evaluation						
commitment to results						
				learning performance		
				writing performance		
				learning retention		
Satisfaction (sat.)	sat.	member sat.	sat.		sat.outcome	sat. results
participation	participation		participation	participation rate		commitment
cohesiveness						
conflict management						
influence						
confidence						
attitude						
general satisfaction						
decision satisfaction						
				dominance		
				flaming		
					sat. process	sat. process
consensus	consensus					
decision agreement						
commitment						

Appendix 3. Roles in collaboration support

Interview protocol about the tasks of a facilitator, their role, their organizational setting and the successfulness of collaboration support in the organization.

Interview Protocol – Facilitator job and skill characteristics – March – August 2006 ***Individual characteristics: Current situation!***	
What is your job title?	Job title:
How many reporting levels are there between you and the head of your organization?	Nr. Levels:
Do you supervise other employees in your organization or do other employees report to you?	Description position in hierarchy:
Are you self-employed	Yes_______ No _______ Partly (what %?)_______
If **not** self-employed: What is the overall mission or product of your organization?	Internal/external profile:
Do you work for an organization that specializes in providing group support?	Internal/external profile:
What is the role of group support in your organization?	Internal/external profile:
What other services besides group support (in general) does the organization provide?	Internal/external profile:
Are the groups you support groups in your own organization or in other organizations or both?	Internal/external profile:
Next, we are going to ask you some questions about what you fundamentally do when you support groups. **Role pattern:** **Over the years= in entire career for group support** **Current role = current employment situation**	
Please describe the things that you do when you support group work.	
How many years have you worked in group support?	# Years ___________________________
Please estimate the number of workshops/meetings you have supported in these years.	# Workshops _______________________
Please estimate the number of different groups you worked with in these years?	# Different groups/clients ____________ (There can be different groups within an organization)
Please estimate the number of different organizations you worked with in these years?	# Different organizations ____________ (There can be different groups within an organization)

Please estimate the extent to which your current professional work consists of group support How was this in the past?	Current % group support role ____________________ Past % group support role ____________________
Can you give a few examples of different types of group processes that you supported over the years? (e.g. strategy building, requirement negotiation, etc.)	Examples: type of group processes:
How many different types of group processes did you support over the years?	# different types of group processes
Do you often support similar/recurring group processes, or do you support very different group processes each time? Over the years… In your current role…	Recurring/ad-hoc distinction over the years Recurring/ad-hoc distinction in current role
Over the years, how many times (number/percentage) have you used interactive group support **technology** for any portion of the meeting?	# times:
For meetings where you have used **technology**, for what amount of the meeting do you tend to use it?	% **Not use = store away**, not discussion based on output
If you use **technology** in the meeting, do you operate it yourself? (if not by whom?)	(chauffeur or not) always, frequently, sometimes, never
Is the **technology** you use in the meeting, designed by yourself? (if not by whom?) Do others use this technology as well?	(technology designer or not) Yes/no other operator Yes/no
Is the **technology** you use in the meeting, customized to the specific meeting by yourself? (if not by whom?) Does this customizing require coding, or rather configuration?	(tailored technology design, or configuration) always, frequently, sometimes, never
Do you design **processes** (agenda's) for the groups you support in advance?	(collaboration engineer/facilitator) always, frequently, sometimes, never
Do you use **process designs** or agenda's made by others? How are these transferred to you?	(practitioner/facilitator) always, frequently, sometimes, never
Do you **design processes** (agenda's) for others to use? How do you transfer these to others?	(collaboration engineer) always, frequently, sometimes, never
Do you use known process **methods/ facilitation techniques** (agenda items)? Where do you find/learn these?	always, frequently, sometimes, never

Do you adjust these **methods** to the specific process? How and when do you make these adjustments?	always, frequently, sometimes, never
Do you also design new process **methods**? How and when do you design these methods?	always, frequently, sometimes, never
Do you teach/share those new **methods** to/with others? How do you teach/share these?	always, frequently, sometimes, never (teach/share)
	1=very unskilled 7=very skilled
How skilled do you consider yourself with respect to group support?	1 2 3 4 5 6 7
If you support groups, are you the primary leader of the process? What is your role as a process leader?	Role description leadership

Show this overview to the subject

Which role (one or more) would best describe the way you offered group support over the years?
Internal group process technical designer = design group support technology that is used in the organization where you work

- External group process technical designer = design group support technology that is used in other organizations than the organization where you work
- Internal facilitator = offer process support for groups in the organization where you work
- Practitioner (internal facilitator for single recurring process)
- Internal all-round facilitator (for ad-hoc tasks)
- External facilitator = offer process support for groups outside the organization where you work
- External Practitioner (facilitator for single recurring process)
- External all-round facilitator (for ad-hoc tasks)
- Collaboration engineer (external process designer) = design processes that you transfer to practitioners or facilitators
- Group chauffer / technographer = operates technology for a group but does not offer process support

Are there other roles you perform when you support groups?

Which of these roles do you perform most often in your current role?

Training& skills

Please tell us about your formal educational background; What is the highest diploma you have? Did you follow other education that does not generally precede obtaining this diploma?	
How did you learn to support groups as you do? For process support? For technology support?	Group process training or learning activities Technology training or learning activities

What **skills** are critical to offer the group support you offer? Which skill (s) is/are most important of these?	Critical skills
What kind of personality attributes are critical to offer the group support you offer? Which personality attribute (s) is/are most important of these?	Critical personality
What kind of knowledge areas are critical to offer the group support you offer? Which knowledge area (s) is/are most important of these?	Critical knowledge
What skills, personality attributes or knowledge would you like to further develop or acquire to better perform your role?	
Effect of your role/situation	
In your current role, are your services to support groups used frequently? More or less frequently than you would like? More or less frequently than needed to justify the resources allocated to group support by your organization? Do you consider the organizational circumstances in which you perform your role successful? Why (not)? How could it be improved? What are critical success factors? Do you consider the organizational circumstances in which the technology you use is implemented successful? Why (not)? How could it be improved? What are critical success factors? In your experience, are group support technologies widely used in business and industry? In your experience, is facilitation or similar group support service widely used in business and industry?	
What are technological barriers to adopt group support technology in general and in your organization in specific?	Usefulness for task Technical difficulty Access difficulty /hardware requirements Conceptual difficulty Appearance User-friendliness Reliability
What are managerial barriers to adopt group support technology in general and in your organization in specific?	Investment /business case HRM Logistics Resistance to change

What are managerial barriers to adopt process support (facilitation) in general and in your organization in specific?	Investment /business case HRM Logistics Resistance to change Cognitive capacity group members Education background group members Expertise/experience group members Computer/typing skills group members
What attributes of groups themselves can pose barriers to adopt technology and/or process support?	Difficult to train group members Difficult to steer/guide group members Other
How do you see the role of group support changing as supporting technologies continue to evolve?	
In your role for collaboration support, to what extent do you take actions with the intention of not only creating specific process or outcomes, but for the purpose of permanently adjusting the way that the group operates?	
Are there important issues about group support that we haven't covered in this interview?	
For the purpose of adding details regarding the nature of our sample	
What is your age?	
What is your gender?	Male Female Note to interviewer: you probably won't have to ask, but just indicate on the form
THANK YOU	

Appendix 4. Design challenges

Survey used in the study with respect to design challenges

Questionnaire: Preparation of facilitated workshops	
This questionnaire aims to identify challenges during the preparation of facilitated workshops. It will address the approach and facilitation techniques that you use as a facilitator to prepare a workshop. It will also consider the importance of preparation in respect to the effort made. All results will be handled anonymously Please estimate your experience as a facilitator and workshop designer with the questions below.	
Facilitation experience	
How skilled do you consider yourself with respect to designing workshops?	1=very unskilled 7=very skilled 3 4 5 6 7
How many years have you worked as a facilitator?	# Years ____________________
Please estimate the number of workshops you have facilitated.	# Workshops _________________
Please estimate the number of different groups/clients you worked with? (There can be different clients or groups within an organization)	# Different groups/clients __________
Please estimate the number of different organizations you worked with? (There can be different clients or groups within an organization)	# Different organizations __________

Workshop Design Approach	
Do you prepare the workshops you facilitate?	Yes No
How long does a typical preparation take?	
For a workshop of half a day	# Hours____________________
For a workshop of a whole day	# Hours____________________
Whom do you involve in this preparation?	Roles:
Do you use a specific checklist for the preparation of a workshop?	Yes No
→ Would you share this with us?	
Which preparation activities do you regularly execute? Please select as many as apply.	
Analysis of task	Yes No
Analysis of group and context	Yes No
Define tasks/steps	Yes No
Define sub tasks/steps	Yes No
Explore possible techniques	Yes No

Evaluate possible techniques	Yes	No
Choose techniques	Yes	No
Create a detailed hour by hour time frame	Yes	No
Create agenda	Yes	No
Document design	Yes	No
Try design on test group	Yes	No
Do you use tools that support facilitators in the preparation of workshops?	Yes	No
→ References:		
Do you re-use your preparation for similar workshops?	Yes	No
Further comments on your workshop design approach:	______________________	

Workshop design aspects							
To what extent do you consider the following collaboration aspects important during the design of a workshop?	1=very unimportant				7=very important		
Task goal	1	2	3	4	5	6	7
Task complexity	1	2	3	4	5	6	7
Task deliverables	1	2	3	4	5	6	7
Task size	1	2	3	4	5	6	7
Task time frame	1	2	3	4	5	6	7
Group size	1	2	3	4	5	6	7
Group # stakeholders/ interest groups	1	2	3	4	5	6	7
Group education level	1	2	3	4	5	6	7
Group organization culture	1	2	3	4	5	6	7
Group institutionalized methods	1	2	3	4	5	6	7
Generally how available is the information considering the following collaboration aspects during the design of a workshop?	1=very unavailable				7=very available		
Task goal	1	2	3	4	5	6	7
Task complexity	1	2	3	4	5	6	7
Task deliverables	1	2	3	4	5	6	7
Task size	1	2	3	4	5	6	7
Task time frame	1	2	3	4	5	6	7
Group size	1	2	3	4	5	6	7
Group # stakeholders/ interest groups	1	2	3	4	5	6	7
Group education level	1	2	3	4	5	6	7
Group organization culture	1	2	3	4	5	6	7
Group institutionalized methods	1	2	3	4	5	6	7

Facilitation techniques

A facilitation technique is a method that a facilitator uses to have the group perform specific kinds of tasks.

A facilitation tool is an instrument used to perform a facilitation technique such as a GSS or yellow stickies.

Do you have a standard set of facilitation techniques that you regularly use?	Yes	No
If no, go to question 25		

Approximately how many of such standard facilitation techniques do you use?	# Facilitation techniques__________
What kind of facilitation tools do you predominantly use to perform these techniques?	Facilitation tools:
Do you execute the same techniques with certain tools one time and with other tools another time?	Yes No
Do you name and/or number the techniques you use?	Yes No
Do you document your techniques?	Yes No
What kind of information do you document per technique?	______________________________
How do you learn about new facilitation techniques?	
Do you develop your own facilitation techniques?	Yes No
Do you often adjust (customize) your techniques to the situation at hand?	Yes No
How do you choose the techniques to be used in the workshop? What steps do you take?	______________________________
Which criteria do you use to make this choice?	______________________________
When do you make this choice?	Only during the preparation Only during the meeting Both
Further comments on facilitation techniques and tools:	______________________________

Importance of design and preparation	
	1=not very critical 7=very critical
How critical do you consider preparation to be for the success of the workshop you facilitate?	1 2 3 4 5 6 7
	1=very unimportant 7=very important
How important do you consider it to document your workshop preparation?	1 2 3 4 5 6 7
	1=very small effort 7=very large effort
Do you feel that it requires a large effort to prepare a workshop?	1 2 3 4 5 6 7
	1=very small effort 7=very large effort
Do you feel that it requires a large effort to document your workshop design?	1 2 3 4 5 6 7
To conclude, please indicate whether you are familiar with or use the thinkLet concept to plan and execute facilitation interventions.	Never heard of I heard of it but do not use it I know and use it
Final comments:	______________________________

Appendix 5. Choice criteria interview protocol

In this appendix we describe the interview protocol used to determine the criteria to choose among facilitation techniques.

Introduction

This is a depth interview in which we try to find the criteria and assumptions that you use to choose among facilitation techniques.

Let me tell you what I mean with facilitation techniques: A facilitation technique is for instance brainstorming, or clustering of ideas, it is an activity that is a sub step of a collaboration process. A facilitation method can have many sub steps, for instance an often used process in facilitation is brainstorming ideas, then clustering them, and then prioritizing them in each cluster. This method exists thus of 3 facilitation techniques.

Do you understand what we mean with facilitation techniques now?

Do you know and use facilitation techniques? (If no, find out what it is that they use and if it's no facilitation techniques, then end interview)

To understand how you use and choose among your facilitation techniques, I want to ask you to tell me how you would approach the following collaboration process: from that approach we will elicit the steps and choice you made, and try to find the criteria that you use to make these choices.

The fictive case is the following:

Case description

Background
The Hague University is a modern education institute, striving to create optimal study- and work-conditions for its students and staff. Parts of these conditions are the available facilities, material and immaterial. Besides access to state-of-the-art equipment and up-to-date information, these facilities include a comprehensive system of student coaching and counseling.
A wide range of computers is available for the (exchange) students of The Hague University. As each faculty has different needs and capabilities, the software available varies from faculty to faculty. However, each computer lab has the basic capabilities of Microsoft Office available.
At the ICT-Service desk, students can revalue print accounts, scan documents and pictures, print in full color, plot, buy CD's and borrow audio-visual equipment.

Goal
The goal of this afternoon is to render input for a new ICT strategy for education the next 10 years. They want to explore different scenario's and find out what current and future actions are required to make that happen. The session should deliver clear action points for now and indicators for required action points in the future.

Participants
To create an overview of opinions and possibilities, the participants represent all aspects of the organization, management, supporting staff, ICT experts, teachers and students. Some might know each other, but there is no team bond or group history. The relations between students, teachers and management and ICT support are hierarchic and some friction and politics are going on like in many organizations, there are no particular problems between the participants and all are committed to give their input to this meeting.

History
The last strategy report is called “vision for the year 2000.” However, the new building (6 years old) has good ICT management and both hard- and software are up to date.

Session details
Time: The group has an afternoon meeting, for about 4 hours planned.
Participants: 20
Resources: any resources that you need in terms of tools and facilities can be arranged

Interview procedure

Tape the whole interview

Questions:
How do you propose to support this group?
Which steps should the group take to create the deliverables?
→Capture steps on whiteboard for instance
Which facilitation techniques should be used for each step?
→Capture names of the techniques on whiteboard for instance
Why did you choose this approach?
Why did you choose step 1, step 2, etc?
→Capture choice criteria
→Categorize the criteria to their sources and add new sources if you find them

You choose a facilitation technique based on assumptions and choice criteria that relate to your perspective of the quality of collaboration; you think it will help the group to achieve their goal in a pleasant and efficient process. In this interview we will challenge you to elicit the criteria and assumptions that you use. We will also ask you to categorize these criteria to

Your personal preference (I am good at this, have experience with this)
Your perception of good collaboration (I think that this enhances good collaboration)
Your perception of the task (I think this is required to create the deliverables)
Your perception of the group (I think the group needs this)
Part of a standard method (because it is prescribed by my method)
Other reason.

If the answer is something like, “that is what the group needs”, why does the group need that, how does it contribute to creation of the deliverables? How does it contribute to the efficiency or pleasance of the process? Or “that creates a fair process” ask why is that fair? According to which principles? Your own? Others?
If people answer the question why did you choose this technique they will probably explain its benefits, but not the assumption behind that choice

Capture the main steps, techniques and choice criteria on a whiteboard or something like that to be able to refer back to it during the interview.

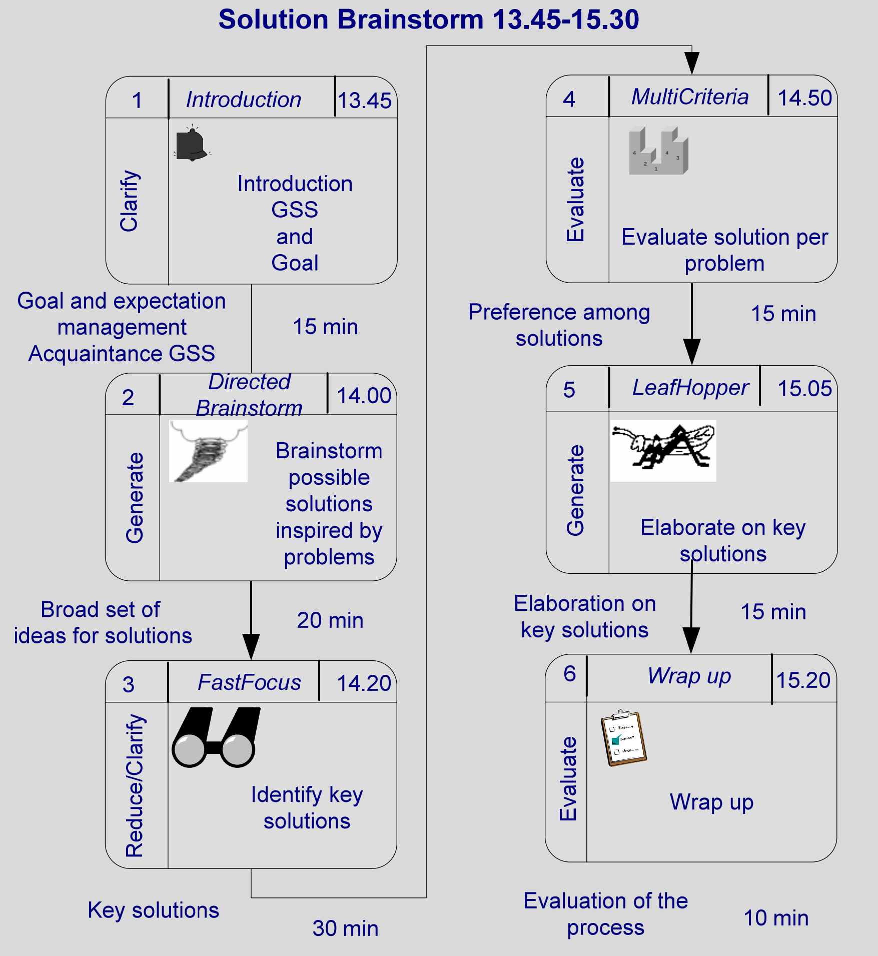
Additional questions for after the interview:

How many different facilitation techniques do you know?
How many do you use regularly?
Did you ever think about how and why you choose certain techniques?
Do you think that this choice is important?
Do you find choosing among facilitation techniques difficult?
What do you need in order to make the choice among facilitation techniques easier?
Do you have a method or tool that supports you in choosing facilitation techniques?
You support collaboration, what is in your opinion good collaboration?

Appendix 6. Process prescription example

In this appendix we describe an example of a process prescription. The layout of the prescription has been adjusted by removing empty spaces. This prescription was used for case 6 in chapter 5. It is instantiated to brainstorm solutions for a project about a trolley network in a large North African city. Author: Gwendolyn Kolfschoten April 2007.

Facilitation process model



Assumptions document

Goal, deliverables, scope

The goal of this session is to brainstorm solutions for the HTM consultancy case. In the case students have to come up with solutions to upgrade a trolley in Casablanca they need to increase the capacity of the line. One of the key factors that determine the line capacity is the speed of the trolleys on the line. They can also upgrade the trolleys themselves and there are some options to extend the line. They will evaluate the solutions they brainstormed on several criteria and they will elaborate on their implications. The result of the session can be used to further write their reports for the project.

Institutionalized methods in organization when used

The students are familiar with the multi criteria approach. They have learned about brainstorming and GSS but most of them did not yet use the GSS.

Time frame, group size

The session takes place in two regular lecture hours; we did not plan a break in the session because of the limited time frame. We will run 4 groups of 12 students each in parallel.

Content/Domain expertise required
You do not need a lot of content info. The ideas for solutions should come from the group, you need to inspire them, offering suggestions and you offer the criteria for evaluation. If you want you can take a look at the slides of the introduction lecture they got on the case. They will be provided.

Guiding rules for behavior

- Try to make detailed and relevant comments, judgment comes in the voting activity
- Let each other finish talking, listen to each other, read each other's contributions, reading is the equivalent of listening in electronic discussion.
- Try to come up with unique, original solutions, think broad, be open for ideas

Group background, Group experience, Group context, education level, org. culture
The groups are first year bachelor students from this faculty. This is their second project. They have no experience with the GSS and limited experience in group work. The challenge in these sessions is that they have to brainstorm with students from other groups. This might be strange to them but it will increase the amount of perspectives and inspire them with new solutions. Participation in the session is obliged for the students. They can use the results of the brainstorm for their reports.

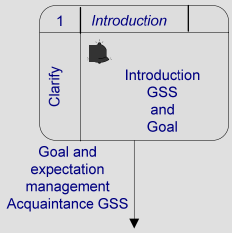
Available technology and tools, required knowledge of technology
You will each run a separate GSS session. You need to prepare your role as a facilitator and as a chauffeur as you will have to do both. To prepare your role as a chauffeur you need to try the software some time. There is also a document with screenshots of the tricky things in Group Systems.

Stakeholders/Actors involved and stakes
Stakes of the students are that they come up with good solutions and that they can explain the implications of those solutions. If they do a good brainstorm they can use the results directly in their report. A stake that is thwarted is that they need to share their ideas with other groups. The best group can present their solution at the HTM office, and therefore there is some competition among the students. However, it can also inspire them to come up with new solutions. Furthermore, HTM is more concerned with the extent to which they thought trough their solution than with the uniqueness of the solution.

The skills and knowledge required for the execution of the process besides those documented in the process prescription
You need to show some authority to the younger students. It is important that you can explain clearly the advantages of the GSS in general and the added value of each step for the students. Furthermore, you need to be able to defend the methods we use in case students question their 'fairness' or supportiveness.

Script

1. Introduction



Explain the goal and explain to the students why it is useful to participate and why the results will be useful to them:

Goal: to come up with solutions for the HTM case, to evaluate the solutions and to elaborate on the implications of those solutions
Added value of participation: to experience a GSS session, to see what it is like.
Added value of results: the results can be used for your project report.

Explain the rules:

Try to make detailed and relevant comments; judgment comes in the voting activity

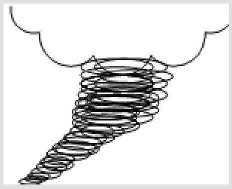
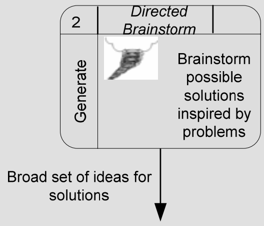
<table>
<tr><td colspan="2">Let each other finish talking, listen to each other, read each other’s contributions, reading is the equivalent of listening in electronic discussion.
Try to come up with unique, original solutions, think broad, and be open for ideas

Explain the advantages of GSS:

Parallel working
Anonymous
Structured and focused
Automatic minutes and automatic voting results

Do a quick exercise with the Group to elicit the key problems in the case study (same as first thinkLet)

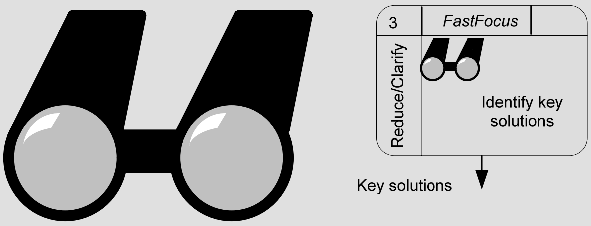
Explain that all groups will see all results, they can learn from each other</td></tr>
<tr><td></td><td>2. DirectedBrainstorm

Overview
Participants will brainstorm solutions on separate pages. When a solution is submitted, the pages swap and the participant gets a different page with contributions of others to add new contributions or to respond to. The facilitator inspires the group by emphasizing different aspects of the problem.
Metaphor
This thinkLet is named DirectedBrainstorm because the input of the participants will be focused by the emphasis on different aspects of the problem</td></tr>
<tr><td></td><td>Script</td></tr>
<tr><td></td><td>Do this
Explain that this step is important to generate a First set of solutions. Explain that we are looking for creative solutions to the problem.
Explain the assignment; brainstorm as many creative solutions to the trolley problem as you can think of, based on the different aspects of the problem.</td></tr>
<tr><td></td><td>Instructions
Click ‘go’
You will get an empty page in front of you
Type in a solution, one solution at a time
Click ‘submit’
You will now see a new page that already has a solution from somebody else.
You can now:
Add a new solution independent of the solution that is already there
Elaborate on the solution or respond to it, in this case refer to the random number behind it. (e.g. for idea , 31: addition)
Be creative and be inspired by the solutions of others, I will also remind you about the different aspects of the problem.</td></tr>
</table>

	After 5 min. give a prompt every 2-3 min to highlight the different aspects of the problem: old rails, old trolleys, difficulty with crossroads, safety problems, accidents, financial issues, lack of capacity, low status of this mode of transportation Limitation to ticket price Try to push the group to go further than the obvious solutions and open doors; e.g. do you think that HTM did not already think of that?, try to think of more unique solutions. Keep in mind the 20 min.
	Rules to maintain One solution a time When you respond or elaborate, refer to the number of the idea you respond to
	What will happen?
	Pattern of Collaboration Key pattern: Generate Participants will generate solutions based on instructions they will read solutions of others and they will listen to your prompts. This will inspire them to come up with new and creative solutions. It might also help to try and push them to go beyond the obvious solutions and open doors.
	Expected result The result of the directed brainstorm is a large list with solutions divided over multiple pages. The list can contain redundancy and double solutions.
	Timeframe We will take 20 minutes for this step. Make sure that you do not reduce the time for this step. If you are behind on schedule, reduce time in other steps, not in this one. The session will be meaning less if it is based on an incomplete list of solutions.
	Challenges After a while the amount of input reduces This is normal, there is a curve in the input of a brainstorm thinkLet, after a while people need more time to read the ideas of others, after that the amount of input will increase again.
	Contribution In this step we will brainstorm the solutions. Everyone has a chance this way to contribute their ideas for solutions, and participants will inspire each other. Because we divide the input on multiple pages, we reduce the information overload in the session. There will be redundant and double solutions, therefore in the next step we need to reduce and clarify the results to converge to a short list of ideas.
	DirectedBrainstorm example We did directed brainstorm to identify possibilities to improve a production process. During the brainstorm different problems with respect to the process were illuminated. Because people read each other's ideas they became inspired and came up with new creative ideas to improve the process. Some of the best ideas were implemented.

3. FastFocus



Overview

In this thinkLet we will ask each participant in turn to name the most important solution on their page. The facilitator will clarify the contributions and make sure there is no overlap in the list. We continue until there is a complete summary of the brainstorm

Metaphor

FastFocus is a relatively fast way to create a good and complete summary, and to clarify the ideas.

Script

Do this

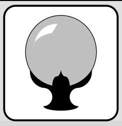
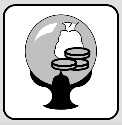
Turn of the “inactive timer” but keep the participants in the EBS tools, with one page in front of them
Open a separate list on the public screen and let the chauffeur get ready to add ideas.
Explain that this summary round is required to reduce and clarify the large amount of ideas from the brainstorm. This will make it easier to vote and to elaborate on the solutions.

Instructions

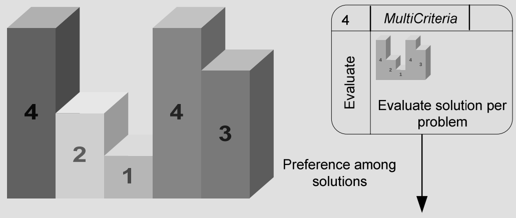
We will now make a summary of this list. I will ask each of you in turn to contribute the key idea on your page to the group summary. We will continue until all key ideas are contributed, but you can only contribute one idea at a time.
Start with the first participant and ask what the key idea on his/her page is.
Check if the solution is clear to everyone
Check if the contribution is a solution, e.g. solving the problem
Check if the solution is correctly captured by the chauffeur on the main screen
Check if the solution does not overlap with another solution
After one round, let participants swap pages
Ask them whether there are important ideas on this page that are not yet on the main list. If there are many, make another round, if there are few; add only those that are left.
End with asking the group to confirm that the summary is complete

Rules to maintain

One solution per person
Let participants contribute in turn, don’t let others add ideas, if you do, the dominant people in the group will determine the list.
Solutions should not appear twice in the list and should not overlap with other solutions
Solutions must be clear and should be accepted by the group
Remember that you are making a summary, not yet a selection, so no discussions whether the idea is important, however, rephrase if the contribution is not a solution to the problem.

	**What will happen?**
	**Pattern of Collaboration** The pattern of collaboration is reducing and clarification. The participants will in turn contribute solutions from their page. You can rephrase and clarify the solutions in a short discussion, and let the chauffeur capture the result. This way you build support for the resulting summary of the brainstorm.
	**Expected result** A small set of solutions without redundancy and overlap and with support of the group
	**Timeframe** We will take 30 minutes for this step
	**Challenges** Too many things to think about Try to make a checklist in your mind or on paper. While some checks seem arbitrary, they are all very important. If solutions are unclear, overlapping or not really solutions, the voting step will be very difficult The discussion takes too long Ask for short and to the point formulation, keep the discussion focused, and ask participants to summarize their comment as a proposal for the formulation of the solution. Participants want to contribute more than one solution Do not allow this; explain to people that there will be a chance at the end to add missing solutions. However, if you let people contribute more than one solution or when it's not their turn; dominant group members will determine the list.
	**Contribution** To get a good overview of the key solutions in the brainstorm we need to remove redundancy, overlap and unclear solutions. Therefore we make this summary. When we skip this step, the voting round will be based on misunderstanding and different interpretations. Most of this problem will be solved by a good FastFocus.
	**FastFocus example** To determine a strategy for a ministry, a variety of experts were invited. After an elaborate brainstorm there was a large set of ideas for this strategy. The result for the minister had to be a short and concise list of ideas. Therefore we did a fast focus. The resulting set was a set of strategies that were supported and understood by all experts.

4. MultiCriteria



Overview
The participants Judge the solutions based on several performance indicators. The results will be discussed in the next thinkLet.

Metaphor
Multi criteria analysis is a known technique in which we compare solutions on several criteria.

Script

Do this
Explain that we perform this step to help the students in their analysis and comparison of the different solutions.
Explain/clarify the different performance indicators, and ask the participants if they agree with these criteria:

- Capacity increase
- Costs
- Sustainability (long term rate of return)
- Robustness (amount of accidents, interruptions, strikes, etc.)
- Feasibility technical
- Feasibility social (organizational, cultural)

Explain the scale: 1 the solution scores very negative on this performance indicator, 5: the solution scores very positive on this performance indicator. Note that high costs should therefore be scored (1) negative, and low costs, positive.
Send the participants the ballot form.
Collect the ballot forms
Explain the voting results
When necessary, remove several solutions that scored very low.

Instructions
Click through the matrix and fill in the scores of the different performance indicators.
When you complete the form, cast your ballot with the small ballot box button.

Rules to maintain
Try to estimate the score for each criterion separately

What will happen?

Pattern of Collaboration
The pattern of collaboration is evaluation. Participants will judge the solutions on the different performance indicators.

Expected result
The result of MultiCriteria is an overview of the average score of each solution per cell. When the scale was interpreted right, the total per row indicates how well the solutions score compared to each other. When you click with the right mouse button on a cell you can show the cell graph, to visualize the spread of votes per cell. You can decide not to continue with some solutions when they score very low.

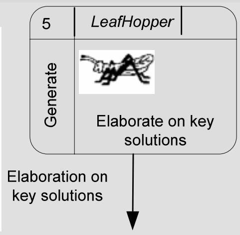
Timeframe
We will take 15 minutes for this step.

Challenges
When you have a very long list of solutions from the FastFocus this step might take too long. In this case you can first vote on the ideas using a checkmark thinkLet, based on feasibility. Select the highest scoring solutions and than do the multi criteria analysis.

Contribution
This step is important to further analyze and compare the solutions. The students can use the results in their report, but they need to add weight factors and they will need to explain the scores and the method used.

Multi Criteria example
This thinkLet is often used to judge strategies or trends on their impact and feasibility or impact and certainty. This helps to gain insight in different implications of these trends/strategies.

5. LeafHopper



Overview
The participants will now view the final set of solutions and they can add ideas about the implications of the solution.
Metaphor
A leafhopper jumps from leaf to leaf to eat and go on, like in this brainstorm participant scan add suggestions to the ideas they choose.

Script

Do this
Explain that this last step is important to help the students in further

elaborating on their solutions to explore implications.
Explain that the students can add ideas about the implications of the solutions and about problems that might arise during the implementation of this solution
The students can choose for which solution they begin with their brainstorm.

Instructions
Double click the solution you want to work on.
Add your ideas on implications and click on submit.
You can also react on other ideas or elaborate on contributions of others.

Rules to maintain
Ensure that all solutions are addressed; bring unaddressed solutions to the attention of the participants.
Ask participants to refer to ideas of others using the reference numbers as in the free brainstorm.

What will happen?

Pattern of Collaboration
Generate, participants will elaborate on solutions and think about their implication and implementation.

Expected result
The result of this step is a set of solutions with elaboration on their implications and implementation. The students can use this result as the basis for their reports.

Timeframe
We will take 15 minutes for this step, or more if you have time left.

Challenges
All participants start working on the first solution, therefore the bottom of the list is not addressed.
Divide participants in two groups; ask half to start at the top, and half to start at the bottom.

Contribution
This last step will offer a first step to further elaborate on the solutions they proposed.

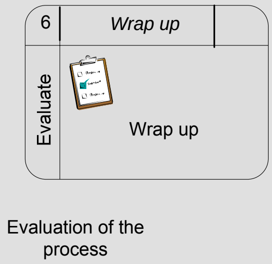
LeafHopper example
We used LeafHopper in an organization where twelve difficult problems had to be solved with experts from various backgrounds. Using this technique we could all together work on the problems and add to each other’s ideas. When we would have used on each problem separately, some experts would not have been able to contribute to some of the problems.

6. Wrap-up

Reflect on the results, and compare them with the goal explained in the introduction: Goal: to come up with solutions for the HTM case, to evaluate the solutions and to elaborate on the implications of those solutions

If time allows it, make a quick round among the students (verbally) and ask the how they liked the brainstorm and what they liked about working with the GSS.

Ask students to fill out the questionnaire.
Ask students to shut down their computer.



Appendix 7. Lessons learned from the process prescription evaluation

In this appendix we describe the lessons learned from the cases in which we evaluated the process prescription to further improve its completeness and transferability.

Case 1: Practitioners case consultants

In this case study two consultants from a medium-size consultancy firm focused on construction projects became practitioner to run a risk assessment session with the use of GSS. The purpose of this project was to add this skill (GSS based risk assessment) to their portfolio, and to use it for different groups and different organizations. The consultants both had experience in running risk assessment sessions with pen and paper, but did not have GSS experience. To run the process with GSS the activities in the process did not change, but the facilitation techniques used were altered to take full benefit from the GSS functionalities.

Preparation and transfer

The process prescription and thinkLet selection was done in collaboration with the practitioners, as they also functioned as project managers. After this participative design effort, the process was documented according to a preliminary version of the process prescription template and the thinkLet script in which the following elements were included:

- The facilitation process model
- The thinkLet scripts containing identification, a script with slightly different elements but conveying rules, do this, say this and more detailed explanation of actions, parameters and capabilities, and the what-will-happen document
- A script for the introduction and wrap up
- Cue cards that they modified themselves, adding memory aids that where important to them.
- Sheets to present the introduction and the different instructions to the group

There was thus no assumption document and the icons for the thinkLet script were not used in this preliminary version of the process prescription.

In the transfer first the general process was discussed based on the facilitation process model and the process slides. Next, the script for each thinkLet was discussed, with focus on the essential script elements and challenges. Furthermore, the GSS tool for each activity was demonstrated and collaboration with the chauffeur (experienced technology operator) was briefly addressed when needed. The practitioners re-read the script and altered the cue cards customizing them as their personal memory aids to further prepare themselves. In total the practitioners spend 12 hours each to prepare for the facilitation task.

Results

The resulting collaboration process went mainly as designed. While the practitioners thought they executed the script as instructed, the collaboration engineer observed several instances in which important instructions or script elements were forgotten. Most important were the purpose of different activities and the explanation of the rationale behind the process prescription. The process prescription and the transfer were considered complete and information was offered in a useful order. The mental effort of the practitioner task was rated 6.5 on a scale from 1-9, slightly higher than a routine task in their work as consultants. They indicated that they expected this mental effort to decrease rapidly in a next iteration of the task. The mental effort/difficulty was not higher than expected, maybe slightly lower. The process did not offer much added value compared to their manual process, but the GSS did offer added value compared to their original manual process.

The following lessons learned and opportunities for improvement were concluded from this first case study:

Lessons learned about the transfer

1. It is very important that the practitioner can argue why the goal, the approach and the tools used are the appropriate method and what their added value and advantages are. Participants in the meeting were dissatisfied with one of the functionalities of the system that was used. When they complained about this feature the practitioner admitted its weakness, rather than explaining the reason for choosing this feature and the added value of using it. The added value of the tools used can be documented in the assumption document and can be rehearsed in the transfer training.
2. The practitioners indicated after the transfer that they wanted to see what the activities would look like in the GSS. For this purpose we did a walk-through with the GSS with one of the practitioners. The other practitioner indicated that he felt that he really missed this part of the transfer. We therefore need to alter the transfer training approach; we need to first run though the process in general and then address the details per activity. Furthermore, a try out of the practitioner task would have been nice, such a try out with the GSS is essential when practitioners have no experience with such system.
3. One of the participants objected to perform a critical activity in the process and asked why this activity was included. The practitioner had difficulty answering this question and therefore lost some of his credibility as process leader. The purpose and contribution of the different activities should be explained to the group, this was indicated in the script, but the importance of this activity should be better emphasized in the transfer training

Lessons learned about the process prescription

4. Due to some logistic problems with the set-up of the GSS, and a delay in the introduction, the first brainstorming activity was rushed. Since this activity was used to create the basic set of risks for the assessment, it affected the quality and completeness of the entire assessment in a negative way. In the wrap up, participants complained both about the lack of time for brainstorming and about the completeness of the risk assessment. Although this is a known pitfall, it was not reported in the script or addressed in the transfer: we therefore need to improve the completeness of the challenges section.
5. After the transfer the practitioners complained that during the discussion of the script they could not follow at which activity they were in the process. Also they complained that the script was too extensively documented and did not offer much structure. We therefore need to improve the thinkLet scripts; the layout can be improved, to support better overview and recognition of thinkLet activities.
6. A last incident happened when the practitioners tried to build consensus about the quantitative risk assessment. When they tried to decide for the group what the value at risk should be, the group did not accept this; they wanted to stick to the conflicting results they created, as they displayed the uncertainty about some of the risks. In manual risk assessment sessions the consultants were used to determine the value at risk with the group, and did not experience this problem. We learned that we need to explain the practitioners the contrast of making decisions for the group as a consultant, compared to the democratic rules builds in the GSS.

Case 2: Practitioners case undergraduate students

In this case study we let undergraduate students, who participated in a course on facilitation of group meetings, execute a session for peers in a course on system design. The session concerned a process for requirements negotiation according to the Easy Win Win approach. (Briggs and Gruenbacher, 2002) In the course on facilitation the background on the task of a facilitator and the function of collaboration support in general was discussed and tested in an exam.

Preparation and transfer

The students got a full day workshop in which they experienced and tried to facilitate several thinkLets. There were a few months between the workshop and the process transfer. One of the two practitioners had considerable experience as a technical assistant, had observed a large number of facilitated GSS sessions, and had a one time experience in facilitation. Given the facilitation course experience we used only one hour to transfer the process prescription. In this hour we explained the topic of the meeting and the assignment the students got. Next, we discussed the process in general based on the process model and the slides of the facilitation process. Last, we discussed the thinkLet scripts, the important instructions, challenges and questions from the practitioners The transfer was very short and especially the content of the session was discussed only marginally. In the Collaboration Engineering approach it is assumed that the practitioners are domain experts, but in this case the practitioners had no more expertise on the topic than the participants. Furthermore, the practitioners did not study the documentation very extensively after the transfer (1-1,5 hours) as they already knew the thinkLets.

The students got a similar process prescription as the practitioners in case 1. Again, in this preliminary version of the process prescription template, we did not yet offer them an assumption document, and the icons were not used. Furthermore, there was no improvement made to the lay-out of the script. Due to sickness of one practitioner, only two practitioners ran the process. One ran the process twice in a row. This practitioner had chauffeur experience.

Results

The process went less well than the first case study. Student participants in the process had to play a role, to simulate a process for functional requirements engineering. However, participants had difficulty playing these roles and thus the 'simulation' did not become very realistic. This had two specific reasons. First the participating students were not very highly motivated, did not have a clear overview of the learning goals and added value of participation and felt that they were used as 'guinea pigs' for research. This resulted in low response rates in discussions and a slow process in which practitioners needed to spend much effort on motivating the participants. Second, the practitioners had difficulty to run the session in the time allocated, which increased the pressure to motivate and encourage participants. Therefore this became a large drain on the attention of the practitioner. As a result the practitioners started to run the session more as a demonstration. For this purpose they made significant alteration to the script. Furthermore, the added value for the participants, with respect to the content became considerably less. A last challenge was that the chauffeurs were relatively un-experienced this time and therefore the teamwork between chauffeur and practitioner was challenging. In the results we see that commitment of the participants was rather low, and consequently, execution of the process becomes more difficult as participants are less willing to make effort and share knowledge.

The practitioners did not spend much time preparing themselves after the transfer. After the sessions they indicated that this was a mistake and that they should have rehearsed the smaller details and challenges. In the transfer these should be better emphasized and the importance of critical script elements should be illustrated with examples on how things can go wrong. The cue cards and process model were used. When the practitioners work with a chauffeur the information for the chauffeur should not be included on the cue cards, instead special cue cards for the chauffeurs could be made. We learned in this case study that the facilitation process model should be instantiated with the specific timeframe of the meeting so it can be used to monitor whether the process runs on schedule.

Lessons learned about the execution

1. A demo session, one where the stakeholders are role-played and no real stakes are present brings challenges on its own and is thus not very representative for a real session. This is an important insight for the transfer training, as it was suggested to perform a role-play session as an exercise, but a role-played session seems to be more difficult because the practitioner needs to motivate participants to play their role, which poses a significant additional task. Therefore a role-played session might

work counterproductive as an exercise (harm self-efficacy). However, participants in an organization might be more serious and committed in a pilot/simulation session than students.

Lessons learned about the transfer training

2. The importance of some aspects of the scripts is underestimated. An example of this is the introduction and verification of the goal and deliverables with the group to gain support and to motivate the group. This activity was not performed, while it could have helped to increase motivation. Another aspect for quick and efficient transfer training could be to ask the practitioners to introduce the different activities or write a short introduction speech for each activity and compare these. The script elements that are important in stimulating participation should not only be explained, their importance should be emphasized, and some of them should be practiced in advance.
3. Part of the reason that it was difficult for the practitioners to motivate participation and role-playing was the fact that they were unable to answer questions about the different stakes and responsibilities for the different roles. The practitioners, if not domain experts should be encouraged to prepare the introduction by themselves, to alter the slides and to understand the topic. Especially they should be able to clarify the content instantiated in the script (categories, criteria, brainstorm questions, etc.)
4. Some practitioners did not use their cue cards, and some made different memory aids for themselves. Practitioners should be encouraged to make their own cue cards, or alter them, so it contains only the information they find useful, therewith reducing the cognitive load of using them.

Lessons learned about the process prescription

5. The timeframe of the session should be written out for each session separately to enable the practitioner to track their progress. Practitioners should be instructed where and how to cut in the time for activities when they are behind schedule, often they made the wrong decisions here, cutting short the brainstorm, so no good session input is generated, as we also saw in case 1, is such pitfall, and also in this case, important activities were rushed, while other activities were better candidates to increase efficiency. The time management and the activities that can be shortened in case other activities are delayed should be addressed.

In table 5.4 we can see the results from the questionnaire. Practitioners scored between 3.5 and 4.3 on average on the various quality dimensions, which equals slightly insufficient to neutral. The practitioner that ran the session twice did not perform much better in the second session, except on commitment which slightly increased in the second session. Given the challenges listed above this result is not very surprising. The practitioners seemed surprised about the challenges they encountered, and while they did remember the instructions they had to give, they did not know what to do when these instructions were not followed. Even the challenges that were documented with possible solutions did not seem to offer much support during the execution. A key lesson from this case study is therefore that a walk-trough of the script is not sufficient practitioners need to be challenged in the transfer to really think about the different problems that can occur during the process, so they can think in advance about ways to deal with a lack of motivation and other challenges.

The mental effort of the transfer was rated 6 on average (9pnt scale). The mental effort of execution was considered much higher than indicated by the practitioners from case 1. 8,3 on average. Causes of this high mental effort were not only the lack of motivation, but also the large group size (n = 27-30). Both did not think it would be easy but found the task more difficult than expected.

Case 3: Practitioners case undergraduate students

The set up of this case study was similar to case 2, but in this situation the practitioner students where master students while the participating students were bachelor students.

Furthermore, the participating students needed the results of the session for a project that was part of the bachelor curriculum. The session supported the bachelor students in finding solutions for their project and comparing these solutions with a multi criteria analysis. In this case study the assumption document was introduced to discuss the background of the collaboration process with the practitioners. In this case six practitioners participated, and the groups they supported were considerably smaller. A challenge was that the practitioners had to offer technical support to their peers and could not use experienced chauffeurs. For this purpose we added a GSS walk-through to the transfer and we offered an overview of the tricky settings and data manipulations with screenshots and instructions. Again the practitioners learned a generic set of thinkLets during a one day workshop as part of a facilitation course, in which techniques were discussed, experienced and tried by the practitioners. In this case, the time between the thinkLet workshop and the process transfer was shorter. The practitioners had no facilitation or chauffeur experience, and only participated in a few GSS sessions as part of their study. Besides the workshop, the facilitation course offered some background on the task of a facilitator and the function of collaboration support in general which was tested in an exam.

Preparation and transfer

The transfer lasted 1 hour in which the process was first discussed in general based on the assumptions, the process model and the slides and next we ran through the thinkLet scripts to discuss critical instructions and challenges. Practitioners prepared very differently. In reaction to the material offered practitioners mentioned that the cue cards and script were double information. However, some practitioners used both, some used only one of both and some developed their own memory aid for execution. Others just re-read the documentation, and some of the practitioner's rehearsed in pairs especially focusing on the use of the GSS. In the evaluation practitioners mentioned that they felt not sufficiently equipped to explain the contribution of each thinkLet. The information about the contribution of each thinkLet was provided but should be emphasized, In general, practitioners would have liked to have more opportunity to test and try the process activities, for instance in a simulation session.

Results

The process ran as prescribed, but different practitioners made different variations to the script. Some of these variations had a positive effect, but many involved forgetting script elements or less precise instructions. One of the thinkLets (FastFocus) was considered particularly difficult, as it required them to both lead a discussion and to capture and clarify the key issues rising from that discussion. The practitioners indicated they would have liked to exercise with this activity. In several activities (voting, final brainstorm) it was difficult for the practitioner to present the results of an activity to the group, and the script did not offer much support in this task. The rooms in which they had to facilitate were small, which caused inconvenience.

The results for each practitioner are listed again in table 5.4. The researcher observed that the practitioners that scored lower did show less self-esteem and made more deviations from the script. However, still most of the challenges and problems that emerged were documented, but not sufficiently rehearsed. In this case study, where practitioners were less experienced than in the previous cases we found that the rehearsal and discussion elements of the transfer training became more important.

Lessons learned were:

1) Confirmation of previous lessons were:
    a. We need to add a challenge of the generate thinkLets to the script, explaining the risk of reducing the time for brainstorming.
    b. The contribution and specific advantages of the thinkLets should be explained better, to help the practitioner when (s)he needs to defend the use of the thinkLet.
    c. When using technology, the practitioners should practice with the tools.
    d. For voting thinkLets an example of a result display and the meaning of the different numbers should be explained

e. Content of the cue cards should be customizable.
f. Introducing the goal and gaining commitment remains a hurdle, its importance should be re-emphasized and illustrated during the transfer training.

Lessons learned about the execution

2) The practitioner should be able to explain his role as process leader. Also, the role separation and collaboration with the chauffeur should be better prepared. In this case practitioners had to alternate roles. Working in pairs, in one session they did the process execution and in one session they did the technical support. However, in several cases, the student with the task of technical support gave the group instructions on how to use the technology, which was the task of the process leader.

Lessons learned about the process prescription

3) The script should be as concise as possible; with all the improvements that were implemented in the revised versions of the process prescription template, its size and therewith its intrinsic cognitive load increases. We need to deliberate for each information element whether it will offer support in executing the task, and whether the support outweighs the added complexity.

Lessons learned about the transfer training

4) The difficult thinkLets should be practiced during the transfer, to enable the practitioners to experience and anticipate these difficulties. While we can describe the challenges involved, the experience might be different from the anticipated challenge.

The mental effort of the transfer and the execution effort were rated by four practitioners:

Practitioner	1	2	3	4
Transfer effort scale 1-9	4	6	6	4
Execution effort scale 1-9	6	7	8	5

Table 1 practitioner ratings of cognitive effort

In this case some practitioners rated the effort of transfer significantly lower than before. While practitioners rated the effort of execution higher than the effort of transfer, its variation is striking. Practitioner 1 and 4 scored both efforts lower and had the best and most consistent results on their questionnaire. While in this group of practitioners the experience was equal this might indicate that some practitioners simply had more feeling for the task than others, as was also observed by the researcher.

Case 4: Practitioner case consultants (follow-up case 1)

In this case study we continued with case 1, but in this case a more experienced facilitator/ consultant became practitioner. This practitioner worked normally with pen and paper tools to support groups, so the techniques and GSS where mostly new to him. The transfer had to be fast due to time constraints, and thus was fit in 1,5-2 hours. In the transfer first the process model was discussed. Then we discussed the characteristics of GSS and each activity of the script in detail. The importance of activities and the emphasis discussed above were improved in the transfer. Furthermore, the icons in the template were included for the first time.

Preparation and transfer

The practitioner did not prepare himself in addition to the transfer described above, and did not use the cue cards. Also the icons in the script were not used in any way. During execution of the process, some reminders were offered to the practitioner by the researcher, and the researcher introduced the GSS. There were only two large deviations from the script, the instructions for voting were too limited and the interventions to guide the discussion about the voting results were not as scripted.

Results
The practitioner found especially the voting different than expected and expected more from the brainstorm. The convergence activity that he anticipated as difficult ran smooth.

The information provided was quite much; especially given the timeframe of the transfer. There was no information missing, but it would have been nice to have some more experience with the GSS, especially to see how participants would contribute and how results would be presented. Furthermore, it was difficult to link the script to the FPM and to follow along during the transfer. It would help to better link these (FPM and script) with numbers and pictures and to put page numbers in the script. Also the names and terminology were difficult to follow.

Overall the facilitation effort was not considered much more difficult than with the use of manual tools, maybe even easier. However, the transfer was considered more effort intensive.

Lessons learned about the transfer training

1. As in the previous cases we found that the transfer requires more focus on showing how things work in the GSS, especially explain what voting results will look like and how to explain them to the group. When results need to be presented, they must be instantly understood by the practitioner.
2. This practitioner found the used of cue cards a sign of 'lack of preparation' and did not want to use them. This led to many deviations from the script. We think it is therefore important to stimulate practitioners to prepare and create some other kind of reminder if they do not want to use cue cards.

Lessons learned about the process prescription

3. We further found that the script still did not offer sufficient structure and overview. We should create a better link between the FPM and the script through pictures, FPM blocks, numbers that match, and the use of page numbers
4. Also, we found that the memory aids and structure that was provided was not self-explaining. In the transfer training we need to explain more about the structure of the process prescription, and the thinkLets.

The mental effort (scale 1-9) was scored 4,5 for transfer and 2,5 for execution. The scores from the participants where rather high, which is partly because of the experience of the practitioner, however, the template seems to become more complete, and cognitive load seems to be moved more to the transfer than to the execution. Despite the experience of the facilitator the metal effort is rather low, also in comparison to his colleagues in case 1. As an experienced facilitator the practitioner should have been able to identify information that was missing in the process prescription and transfer. As this was not the case we feel that we have achieved some level of completeness. We also feel that the cognitive load reduction is beginning to establish, and that it can be further increased by creating a proper transfer training.

Case 5: Practitioners case undergraduate students (follow-up case 2)

The fifth case was identical to the second case but we used the final version of the process prescription template. The practitioners had either followed a course on facilitation and group processes in which they facilitated the session from case 3, or they were experienced technical assistants for the GSS system, and observed many sessions in that role. The manual in this case study was the full manual as described in chapter 5. We improved the link between the facilitation process model and the thinkLet descriptions by adding the FPM block in the right upper corner of the script pages that concerned the specific step.

Preparation and transfer
The transfer again consisted of a one hour briefing, and the practitioners got the process prescription, slides to present the introduction and assignments in the session and

background about the session content. Practitioners read this information before the transfer. During the transfer we specifically addressed the challenges for each thinkLet and the issues with respect to the 'demonstration-character' of the session.

The practitioners prepared themselves by reading the materials, on which they spent 1-3 hours. One practitioner practiced in explaining things to the group. Some practitioners indicated that they would have read the prescription better if they could do it again, and some indicated that they would prepare more about the content of the meeting.

Results

The researcher observed the session and added input in the brainstorm phase of the process to ensure that the content of the session was elaborate and meaningful. The process for all practitioners went as intended, and surprises were limited. Most surprising was the need to explain instructions in much more detail than expected and the results (how it appeared on the main screen in the GSS) and feedback from the group. Deviations from the script were limited, and most were caused by lack of time. Some practitioners shortened a step and altered the labels of categories to fit the input from the group. These deviations are good as they show the practitioner's ability to be flexible based on the input of the group and the progress in the process. One deviation was not anticipated, a practitioner re-did a sorting task with the group because the purpose was not clear and therefore the results were not useful. Again motivating the students in the demonstration setting was difficult, combined with the relative large group sizes, and they still felt insufficiently prepared with respect to the content of the session. One practitioner also indicated that (s)he was much focused on running the process prescription, while during the session and the reflection (s)he indicated that next time (s)he would focus more on the participants. All practitioners indicated that they would be able to train other students to run the same process, based on the manual.

Lessons learned about the transfer training

1. One of the key pitfalls remained that the practitioners were not explicit enough about the purpose of specific steps. This causes many questions from participants and occasionally it results in outcomes that are not useful. In the training approach we need to ensure that practitioners understand the importance of explaining the purpose of each step and that they have sufficient background to do so.
2. One of the thinkLets (FastFocus) required the practitioner to summarize the input of short discussions and to rephrase this. This was difficult for some practitioners and should be practiced during training

Lessons learned about the process prescription

3. The amount of icons in the manual is considered rather extensive and might be confusing. Furthermore, their meaning was not explained in the briefing. A short explanation can be added, this can also be addressed in the training.
4. The manual does not contain instructions to handle a system failure or problem. In case GSS is used it would be useful to add a short section on what to when the technology breaks down. This can also be addressed in the training.
5. Another suggestion was to add possible questions from participants to the what-will-happen section. We believe however, that these issues are addressed in the contribution and the challenges section.
6. One practitioner wanted some information about the thinkLet concept in the manual. We agree that some basic explanation of the concept would be useful. This can also be addressed in the training.

The practitioners rated the completeness of the manual, the usefulness of each part of the manual and whether they felt equipped to execute the process. The results are presented in table 2 (two practitioners also participated in case 3)

Practitioner	1	2	3	4	Average
Completeness	6	7	6	7	6.50
Usefulness assumptions	6	7	6	5	6.00
Usefulness thinkLets	7	6	5		6.00
Usefulness identification	6		4	4	4.67
Usefulness script	7	7	6	6	6.50
Usefulness what-will-happen	7	6	6	6	6.25
Usefulness FPM	7	6	5	3	5.25
Usefulness cue cards	6	6	6	6	6.00
Felt equipped to execute?	5	7	5	5	5.50

Table 2 Evaluation of the manual completeness,
usefulness, and 'felt equipped'
on 1-7 scale 1 very un - 7 very (useful, complete, equipped)

The results support the lessons learned; only minor additions to the manual were suggested. We interpret that the relatively low score for identification is caused by the fact that this is considered a commodity by the users.

Lessons learned about the execution

7. Two practitioners indicated that they would have liked more information about the group and their reasons for participation. This should be addressed in the assumptions document
8. One practitioner would have liked some general facilitation tips and tricks. Such tips could be part of the training.

The practitioners indicated the mental effort, difficulty and how tiring they found the execution. The results are presented in table 3.

Practitioner	1	2	3	4
Mental effort facilitation	4	7	5	5
Difficulty facilitation	2	4	5	3
Tiring facilitation	3	6	4	4

Table 3 mental effort of facilitation on a scale 1-7

All practitioners further indicated that they thought they would be able to train peer students (with some leadership skills) based on the manual to run the process.

Case 6 Practitioners case undergraduate students (follow-up case 3)

The sixth case was identical to the third case but we used the final version of the process prescription template. The practitioners had all followed a course on facilitation and group processes. The manual in this case was the full manual as described above. A key problem in this case was that 4 of 6 practitioners did not speak the mother language of the participating students. This made it very difficult to support the process. Some sessions were (partly) held in English, others were supported in English while content was generated in Dutch. This made it very challenging to execute the session.

Preparation and transfer

Again, the transfer was very limited. Students got a workshop on the use of thinkLets as part of the facilitation course, after which a walk thought of the process manual was the only preparation. Students spend an average 2.75 hours on additional preparation after the walk through.

Results

The setting of this case was a bit chaotic. Due to limited resources the practitioners had to execute in parallel with 2 other groups and facilitators in one very tight room. This made observation relative difficult. Furthermore, due to the amount of facilitation students and the amount of participating students the group sizes were large and the researcher had to step in facilitating one session, which made observation of the students difficult. However, the evaluations revealed some final insights:

Lessons learned about the transfer training
Students indicated that they would have liked to do a demo session in class to practice and to see how the process prescription was intended. Furthermore, they would have liked to have some more practice with the technology. As indicated in the table below only one student did not feel equipped to execute the process.

Lessons learned about the process prescription
The process prescription was evaluated rather positive (see table 4). While some elements were more useful than others the practitioners indicated that they did not find any information superfluous. Students found the manual clear and liked the structure it offered. Some would have liked more explanation on the tool or screenshots, and one student indicated that more information about the group would be useful. Last, more examples from practice for specific interventions (facilitation tips) were suggested.

Practitioner	1	2	3	4	5	6	Average
Completeness	4	7	3	6	6	6	5.33
Usefulness assumptions	5	5	4	5	6	5	5.00
Usefulness thinkLets	7	6	4	7	4	6	5.67
Usefulness identification	4	6	4	4		5	4.60
Usefulness script	6	7	4	5	6	6	5.67
Usefulness what-will-happen	3	7		5	4	4	4.60
Usefulness FPM	3	3	5	7	6	5	4.83
Usefulness cue cards	2	5	7	5	1	7	4.50
Felt equipped to execute?	6	7	2	6	4	4	4.83

Table 4. Evaluation of the process prescription

Lessons learned about the execution
Practitioners mostly indicated that the session went as expected. One practitioner found the session more stressful than expected, one found the session more easy than expected. Students were satisfied with the results, but most indicated that the results could have been better. However, this is also due to some problems in the projects of the students. Changes in the process were made, but these were smart deliberate changes, mostly to reduce the amount of alternatives in two rounds to increase efficiency. One practitioner indicated that (s)he forgot to mention some rules, mostly this involved insufficiently explanation of the rationale of steps. In the table below (see table 5) the mental effort, difficulty and assessment of how tiring it was to facilitate are indicated.

Practitioner	1	2	3	4	5	6
Mental effort facilitation	5	4	6	6	6	5
Difficulty facilitation	4	4	6	5	5	5
Tiring facilitation	5	5	7	5	4	7

Table 5. Evaluation of the execution

5 practitioners indicated that they felt able to train others to run the process, one indicated that (s)he could not, because he found it difficult to assess his own mistakes.

Appendix 8. Design evaluation

In this appendix we describe the assignment and case description used to evaluate the design approach and design support booklet.

Design Exercise

Assignment:

- Design a collaboration process that will achieve the goal and create the deliverables described in the case study. Take the nature of the group into account. Prepare your design according to the supporting thinkLet-based design booklet. Use the modeling and description techniques as indicated in the design booklet under step 5: Design documentation. Include problem and process description, detailed agenda, and Facilitation Process Model. Add further descriptions and explanations if you think this is necessary.
- Document your design in a way that makes it possible for one of your colleagues to facilitate this collaboration process.
- If you need more information about the case, you can ask me questions.
- Please record the time you spend on creating your design.

Case Description

Note: This is a fictitious case – there is no organization like SIRS in the US.

Proposal Workshop SIRS
Date: April 2007
Time: 12.30-17.30 including lunch
Number of participants: 20
Problem owner SIRS (Social Intelligence- and Research Service)

Introduction

The SIRS was founded on January 1st, 2003; its primary goal is to track down large-scale complex labor related fraud. They focus on large-scale swindlers. There are several ways in which the SIRS can get involved in fraud research. The two most important are:

1. Colleague fraud investigators, such as FBI, IRS and labor inspection discover a large case where social services are a victim.
2. On their own initiative, for instance after a risk analysis or based on signals from the criminal intelligence unit.

Why a workshop?

When the SIRS was founded, it was expected that it would be able to provide more than enough cases for investigation via the first way described above. In practice, however communication is problematic and the number of cases as a result are too low.
The following reasons for this problem were indicated:

- The executive organizations want to solve large cases themselves because they are fun, exiting, interesting and when successful, increase their status.
- The SIRS is seen as a threat, other fraud investigators do not like to loose their responsibilities.
- The SIRS is relatively unknown and does not yet have a certain profile or reputation.
- All organizations that should provide cases are going through a large reorganization and are therefore more focused internally.

The Central problem is therefore: **Compared to expectations, the provision of cases is disappointing, both in number and in size.**

The consequences for the SIRS include that they don’t meet the high expectations set by the government and politicians.

The direction of the solution is to be found in a reduction of the dependence of input by external organizations. Independence can be reached when the SIRS develops her own proposals for research and investigation. In order to develop such proposals the management wants to use the available expertise on research and investigation in the team.

Goal of the session
The goal of the session is therefore: **To increase the supply of cases.**
In the workshop, the SIRS will get insight in the fraud problems that should be dealt with in the research.

The **central question** that will be posed to the participants will be:

What fraud problems should be addresses by the SIRS?

The answer of this question will render a list of fraud problems that match the SIRS demarcation. This means that the participants should have a clear idea of the SIRS demarcation. The list of fraud problems should be elaborated on. We expect the participants to:

- Describe the fraud problem
- Give insight to the fraud problem
- Propose solutions to the fraud problem

The results of the workshop will be a list of research-worth, fraud problems
We want to focus on 5 to 10 problems, the group should gain consensus about this top 5-10.

Rough session agenda
Introduction by problem owner concerning occasion and purpose of the session.
Introduction to the GSS by facilitator.

1. Brainstorming fraud problems that could be approached by the SIRS.
2. Reducing to max 10 problems
3. Per problem brainstorming about:
    - The kind of fraud and the scale. (Potential financial and social damage, actors involved, spreading of the problem over the country.) The "why" question.
    - The kind of activities that should be performed in order to commit the fraud. How is it done? The "what" question.
    - What are possible solution from the SIRS perspective (solutions tactical)
    - What policy measurements should be taken to avoid this kind of fraud? (Solutions policy)

Wrap-up

Agenda

11.00-12.30	Building up the system
12.30-13.30	Lunch and welcome participants
13.30-13.40	Introduction by problem owner
13.40-	Start workshop

....

Questionnaire

This questionnaire is used for the research of Gwendolyn Kolfschoten about the quality of collaboration process design.

Please answer all questions. Be careful to watch the scale of the answer.

Time-spent on the design:	…… hours/minutes				
Please 'grade' or evaluate your own design:	…… grade				
Explain shortly why you think you should receive this grade/evaluation:					
Please shortly describe your approach to this design exercise: (Which steps did you take after reading the exercise?)					
Please offer suggestions for improving the design support (thinkLet design booklet and thinkLet book) you got:					
Please answer the following questions:					
Question	1	2	3	4	5
1.) I found the design exercise difficult (1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
Why (not):					
2.) I found the design exercise took me a lot of effort (1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
Why (not):					
3.) I think my design had the elements of a high quality design displayed at page 4 of the thinkLet design support booklet: (1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
Why (not):					
4.) I agree with the elements of high quality design at p 4. (1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
Why (not):					
5.) I strictly followed the proposed design approach at p 8. (1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
Why (not):					
6.) I think that my design is easy to understand for my peers/colleagues (1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
Why (not):					
7.) I feel confident that I can execute this design (1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
Why (not):					
8.) I did not deviate from the suggested design documentation at p 22/27. (description, agenda, FPM) (1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
Why (not):					
9.) I think the suggested design documentation is comprehensive (1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
Why (not):					
10.) The design guidelines at p 30 helped me to make a better design than I could have without them. (1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
Why (not):					
11.) Without the design support booklet I could not have made a high quality design (1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
Why (not):					
(1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
12.) Elements of High quality design p 4	1	2	3	4	5
I used this information					
I found this information useful					
This information saved me time					

I found this information easy to use					
I fully understood this information					
This information helped me to improve my design					
Without this information I could not make a good design					
This information should be improved					
How should it be improved:					
(1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
13.) Design approach p 8	1	2	3	4	5
I used this information					
I found this information useful					
This information saved me time					
I found this information easy to use					
I fully understood this information					
This information helped me to improve my design					
Without this information I could not make a good design					
This information should be improved					
How should it be improved:					
(1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
14.) ThinkLets (thinkLet book)	1	2	3	4	5
I used this information					
I found this information useful					
This information saved me time					
I found this information easy to use					
I fully understood this information					
This information helped me to improve my design					
Without this information I could not make a good design					
This information should be improved					
How should it be improved:					
(1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
15.) Documentation: Facilitation Process Model p 23	1	2	3	4	5
I used this information					
I found this information useful					
This information saved me time					
I found this information easy to use					
I fully understood of information					
This information helped me to improve my design					
Without this information I could not make a good design					
This information should be improved					
How should it be improved:					
(1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
16.) Documentation Agenda Format p 22	1	2	3	4	5
I used this information					
I found this information useful					
This information saved me time					
I found this information easy to use					
I fully understood this information					

This information helped me to improve my design					
Without this information I could not make a good design					
This information should be improved					
How should it be improved:					
(1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
17.) Design guidelines p 30	1	2	3	4	5
I used this information					
I found this information useful					
This information saved me time					
I found this information easy to use					
I fully understood this information					
This information helped me to improve my design					
Without this information I could not make a good design					
This information should be improved					
How should it be improved:					
(1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
18.) Choice map p 20					
I used this information					
I found this information useful					
This information saved me time					
I found this information easy to use					
I fully understood this information					
This information helped me to improve my design					
Without this information I could not make a good design					
This information should be improved					
How should it be improved:					
(1) strong disagree, (2) disagree, (3) neutral, (4) agree, (5) strong agree					
19.) Classifications p 19					
I used this information					
I found this information useful					
This information saved me time					
I found this information easy to use					
I fully understood this information					
This information helped me to improve my design					
Without this information I could not make a good design					
This information should be improved					
How should it be improved:					
Please rank the support elements from 1-8, 1 = contributing most, 8 = contributing least based on: **20.) Which element contributed most to the quality of your design:**					

Elements of High quality design p 4	
Design approach p 8	
ThinkLets (thinkLet book)	
Documentation Facilitation Process Model p 23	
Documentation Agenda Format p 22	
Design guidelines p 30	

Choice map p 20	
Classifications p 19	
How should it be improved:	
Please rank the support elements from 1-8, 1 = contributing most, 8 = contributing least based on: **19.) Which element contributed most to learning how to design:**	
Elements of High quality design p 4	
Design approach p 8	
ThinkLets (thinkLet book)	
Documentation Facilitation Process Model p 23	
Documentation Design Format p 22	
Design guidelines p 30	
Choice map p 20	
Classifications p 19	
How should it be improved:	
Please rank the support elements from 1-8, 1 = contributing most, 8 = contributing least based on: **20.) Which element contributed most to designing efficiently:**	
Elements of High quality design p 4	
Design approach p 8	
ThinkLets (thinkLet book)	
Documentation Facilitation Process Model p 23	
Documentation Design Format p 22	
Design guidelines p 30	
Choice map p 20	
Classifications p 19	
How should it be improved:	

Format design assignment

Description

Problem and background, motive for the session
Task analysis: goal, deliverables, and objectives
Stakeholder analysis: group, stakes, roles and needs
Resource analysis: time, knowledge, effort and physical resources
Self or facilitator/ practitioner analysis: skills, experience, personality, domain expertise
Process and result decomposition
ThinkLet choice
Agenda building (agenda and FPM form are below)
Process description
Validation

Agenda

#	Activity	Question/ Assignment	Result	ThinkLet & Pattern	Time
1					

Facilitation Process Model

Appendix 9. Master ThinkLet

In this appendix we describe all elements of the master thinkLet and its instantiation for design and transfer.

ThinkLet Example

1. IdentificationClass

ThinkLetname
LeafHopper

Picture

Overview
All participants view a set of pages, one for each of several categories. Each participant hops among the categories to make contributions as dictated by interest and expertise.

Metaphor
A LeafHopper is a small insect that is something like a grasshopper or a cricket. It hops from leaf to leaf eating what it wants, then moving on. We named this thinkLet LeafHopper because the team members can jump from category-to-category, contributing as they are inspired, then moving on to new topics.

2. RuleClass

Constraints
Categories, Contribution specification

Action, condition, effect
Participant: Add any number of contributions to any category in parallel, effect generate, efficiency
Participant: Add only contributions that are relevant to the category in which they are placed, effect structure
Participant: Add only contributions that match to the contribution specification, effect quality of contributions
Participant: Shift focus from category to category as interest and inspiration dictate, effect completeness, inspiration
Participant: Read the contributions of others for inspiration, effect completeness, inspiration
Participant: Respect anonymity of other participants, effect openness, less barriers to contribute

3. RoleClass

RoleName
Participant

Participant selection
For this thinkLet participants need to have experience or interest on related and overlapping topics but not necessarily the exact same topic. Participants should have expertise that provides synergy.

4. CapabilityClass

Requirement and rights
One page for each category (public)
Participants must able to add contributions

Participants must be able to contribute anonymously
A separate communication channel for the brainstorm instruction

Instantiation guidance
In Group Systems face to face: Categorizer: categories are buckets; contributions are list items, work anonymous, present brainstorm instruction verbally/supported with slide.

In Group Systems distributed: Categorizer: categories are buckets; contributions are list items, work anonymous, present brainstorm instruction through participant instruction.

Manual face to face: set up flip-over sheets with categories as headings, offer participants sticky notes and markers(in one color) to make contributions, offer enough room to allow participants to work anonymous, present brainstorm instruction verbally/supported with slide.

5. ScriptClass

DoThis:
Explain the LeafHopper goal contribution
Explain the categories
Ensure that participants understand the categories
Specify the type of contributions required
Ensure that participants understand the type of contributions required
Explain how to add contributions
Explain how to move between categories
Give the brainstorm instruction as scripted below:

SayThis:
"Each of you may have different interests and different expertise."
"Start making contributions in the category in which you have the most interest or the most expertise."
"Then, if you have time, move to the other categories to read to make contributions."
"You may not have time to work on every category, so work first on the categories that are most important to you."
"Read the contributions of others for inspiration."
"Respect each others anonymity."

MaintainRule:
Make sure that you communicate and guard the following rules:
Allow participants to add any number of contributions to any category in parallel
Allow participants to add only contributions that are relevant to the categories in which they are placed
Allow participants to add only contributions that match to the contribution specification
Let participants shift focus from category to category as interest and inspiration dictate
Ensure that participants read the contributions of others for inspiration
Ensure that participants respect anonymity of other participants

6. WhatWillHappenClass

Expected patterns of collaboration:
Primary Pattern: Generate
Participants will generate contributions; this generation thinkLet can have an elaborative, gathering, or creative character.
Secondary Pattern: Organize
Participants will categorize each contribution
Secondary Pattern: Evaluate
Participants can make reflective contributions based on the criteria for contribution specified in the contribution specification and categories.

Expected result
The result of the Leafhopper thinkLet is a categorized set of contributions. The result can contain double contributions, redundant contributions and overlapping contributions. There might be contributions that are categorized in the wrong category.

Timeframe
The time estimation for a LeafHopper activity made by experts is 30 minutes. The deviations based on group size and task sizes are displayed in the table below.

	Small	Large
Group size (normal is 20 people)	-2	-
Task size (200 ideas is normal)	-10	+7

Challenges
The categories are misunderstood
You can identify this challenge when people start talking about the categories after you explained the task. Shortly discuss each category, and check whether everyone understands. Consider re-naming categories to clarify their label. Prevent this challenge by discussing the categories in advance, and by using labels that are familiar to the group.
The categories are not distinguishing
You can identify this challenge when people start complaining that their contributions apply to all categories. To solve this either re-label the categories to make them more distinguishable, or make a category “other” or “general” where overlapping contributions can be placed.
Contributions appear mostly in the first category
It is natural to people start in the first category and to work their way through. However, in a small timeframe this can result in too little contributions in the last category. Therefore emphasize that participants can start with all categories, and that they should start with the categories they find most important/ have most experience with. When this does not help, ask half of the group to start with the last category.

Contribution
The result of this activity is set of contributions for a set of categories. The thinkLet offers an approach to get input that covers a pre-defined scope of topics (categories) The thinkLet allows participants to share related ideas. The thinkLet can be used for elaboration, evaluation, gathering, or creativity:
Elaboration: the categories represent topics that are detailed and further explored by the participants
Evaluation: the contribution specification contains one or more specific criteria, contributions specify how the category answers to this criterion.
Gathering: the participants contribute known information in each category
Creativity: the participants contribute new ideas in each category

SuccessStory
We once worked with a commercial software development team that had 12 tricky issues to resolve. They needed input from engineers, customers, product managers, developers, users, and several other success-critical stakeholder groups. They discovered a rare opportunity when all the key stakeholders were to be in the same place at the same time, and managed to schedule a meeting. Then they realized that although they needed input from all the stakeholders, any given stakeholders only had an interest in about 1/3 of the issues. This meant that no matter what topic was being discussed, 2/3 of these high-powered participants might be sitting around bored. They felt it was impolitic to bore high-powered participants, but unfortunately, the mix of issues and interests was such that they could not simply schedule sub-sessions around each topic. We chose a LeafHopper to resolve this dilemma. The development team posted the issues on topic pages in view of the team. They asked the participants to work first on the topics in which they had the most at stake, and on which they had the most expertise. The participants proposed options for resolving each issue, and then argued the pros and cons of the proposals. The whole

discussion of 12 topics took just over an hour and a half.

7.SelectionGuidanceClass

PatternClassification:
Generate, Organize

ResultClassification:
Input structure

RequiredInput:
Categories: < >
Instantiation guidance
Categories: Identify categories. You can prepare these categories in advance, or you can use the results of a previous step in the process. Make sure that the category names are as short as possible. Make sure that categories cover the scope of the brainstorm. Make sure that categories do not overlap. Consider the use of an “other” category. Make sure that the categories match the expertise and interest of group members, however, not all group members should have interest and expertise on all categories. For a discussion on local transportation problems categories can be: Train, Trolley, Bus, Taxi, Car, Bike, Pedestrian, Other.

Contribution specification: <>
Instantiation guidance
Contribution specification: Make specific what type of contributions you expect from participants; e.g. solutions, problems, elaborations, etc. you can also specify additional criteria for these contributions e.g. feasible solutions, short-term problems, detailed-elaborations.

Insights:
Participant: Add any number of contributions to any category in parallel
Participants focus first on their interest or expertise; this allows participants with different backgrounds to work in parallel. The number of contributions is not limited in any way. Though parallel contribution, information is generated faster than in a sequential pattern. Depending on the level of detail of the contribution specification one can expect contribution rates of 1/min-1/10min
Participant: Add only contributions that are relevant to the category in which they are placed
Participants need to add their contributions in a specific category. This offers them a structure to explore possible contributions in a gathering/evaluation/elaboration situation and it offers inspiration in a creativity situation.
Participant: Add only contributions that match to the contribution specification
The contribution specification can range from a generic concept that only specifies the sub-pattern of collaboration such as for instance “solutions” specifies that the pattern will be creativity where “comments” specifies that is will be generation and “problems” that it will be gathering. Very detailed contribution specification can be for instance a “SMART” specified requirement
Participant: Shift focus from category to category as interest and inspiration dictate
Since you offer multiple topics at a time, the thinkLet allows you to involve participants with different backgrounds, and yet let them work in parallel without losing their attention. The categories offer the participants a framework for thinking. They will either use this framework to explore the scope of the brainstorm or as inspiration.
Participant: Read the contributions of others for inspiration
During the brainstorm participants can read each others contributions this can inspire them to think of additional contributions, not only in a creativity pattern, but also in the other sub patterns.
Participant: Respect anonymity of other participants
Anonymity allows participants to make more critical or controversial contributions.

Combinations:

Preceding thinkLets

	To identify topics	ThemeSeeker OnePage StrawPoll FastFocus
Following thinkLets		
	To further elaborate	BranchBuilder Second LeafHopper
	To evaluate	StrawPoll/MultiCriteria/CheckMark on topics StrawPoll/MultiCriteria/CheckMark on contributions BucketWalk-evaluate classification only BucketShuffle
	To reduce	Concentration (BucketWalk) GoldMiner per Category CheckMark/BroomWagon
Modifiers		Add comments One minute madness Specify additional roles

ChoiceGuidance:

Task fit:

Choose this thinkLet for:

Elaboration: to explore the full scope of a topic, offering categories that cover this scope a framework for elaboration in parallel.

Choose this thinkLet for:

Evaluation: to gain contributions specifying the value based on one or more criteria, for each category in parallel.

Gathering: to explore the full scope of a topic, offering categories that cover this scope a framework to gather all relevant aspects of this topic in parallel.

Creativity: to stimulate thinking about creative ideas from different angles or perspectives in parallel.

Group fit:

For a group with diverse background and partly overlapping expertise/experience or interests

Alternatives:

For Generate:

OnePage: If there is only one category

FreeBrainstorm: If you need a more interactive brainstorm

BranchBuilder: If you need more levels of detail

For Organize:

OnePage-ThemeSeeker-PopcornSort: If you don't know the categories in advance; to create support for the categorization

For Evaluate:

StrawPoll: If you want a quantitative evaluation rather than a qualitative evaluation

DiscussionIndex:

Low: participants will mostly work from their own experience and interest, towards the end of the brainstorm they might be inspired by each other, but no discussion will emerge.

ComplexityIndex:

Participants: Low: Depends on the complexity of the contribution specification

Low information interdependency (parent-child relation category-contribution and contribution-specification)

Low dynamics/uncertainty through categories and contribution specification the thinkLet will offer (when accepted) a predictable specific result

Medium amount of information through amount of categories, contributions and contribution criteria multiplied by time and amount of participants.

Facilitator:

Low: good overview of type, scope and quality of input; limited challenges.

Appendix 10. Research instruments for transfer evaluation

In this appendix we describe the research instruments used to evaluate the transfer of collaboration process design to practitioners.

10a Questionnaire quality of collaboration

Quality of collaboration							
session, date, facilitator							
Please consider each question separately.							
	1=Strongly Disagree			**4=Neutral**			**7=Strongly Agree**
1. I feel satisfied with the way in which today's meeting was conducted.	1	2	3	4	5	6	7
2. I feel good about today's meeting process.	1	2	3	4	5	6	7
3. I liked the way the meeting progressed today.	1	2	3	4	5	6	7
4. I feel satisfied with the procedures used in today's meeting.	1	2	3	4	5	6	7
5. I feel satisfied about the way we carried out the activities in today's meeting.	1	2	3	4	5	6	7
	1=Strongly Disagree			**4=Neutral**			**7=Strongly Agree**
6. I liked the outcome of today's meeting.	1	2	3	4	5	6	7
7. I feel satisfied with the things we achieved in today's meeting.	1	2	3	4	5	6	7
8. When the meeting was over, I felt satisfied with the results.	1	2	3	4	5	6	7
9. Our accomplishments today give me a feeling of satisfaction.	1	2	3	4	5	6	7
10. I am happy with the results of today's meeting.	1	2	3	4	5	6	7
	1=Strongly Disagree			**4=Neutral**			**7=Strongly Agree**
11. I support the goal of this meeting as it was presented in the introduction.	1	2	3	4	5	6	7
12. I had a stake in achieving the goal of this meeting as it was presented in the introduction.	1	2	3	4	5	6	7
13. I was motivated to contribute in this meeting.	1	2	3	4	5	6	7
14. I was willing to put my time and effort in this meeting.	1	2	3	4	5	6	7
15. I found this meeting important.	1	2	3	4	5	6	7

	1=Strongly Disagree			4=Neutral		7=Strongly Agree	
16. I found the meeting worth the time and effort.	1	2	3	4	5	6	7
17. The time and effort requested from me was reasonable.	1	2	3	4	5	6	7
18. I was able to contribute relevant knowledge and experience I had for the meeting.	1	2	3	4	5	6	7
19. The time and effort I spend in the meeting was what I expected	1	2	3	4	5	6	7
20. My input was justified.	1	2	3	4	5	6	7
	1=Strongly Disagree			4=Neutral		7=Strongly Agree	
21. The result of the meeting had the quality I expected.	1	2	3	4	5	6	7
22. What we achieved today met my expectations.	1	2	3	4	5	6	7
23. We achieved what we intended.	1	2	3	4	5	6	7
24. The result has the quality intended.	1	2	3	4	5	6	7
25. The result was as I hoped	1	2	3	4	5	6	7
	1=Strongly Disagree			4=Neutral		7=Strongly Agree	
26. The input asked from me was in balance with the results	1	2	3	4	5	6	7
27. The result was not a waste of my time and effort	1	2	3	4	5	6	7
28. What we achieved was worth the time and effort	1	2	3	4	5	6	7
29. The quality of the results is in balance with the time and effort asked from me	1	2	3	4	5	6	7
30. The quality of the results justifies my input	1	2	3	4	5	6	7

31. In how many workshops have you used tools and methods similar to the ones we used today?

0	1 or 2	3 to 5	6 to 10	more than 10

32. How many years of full-time work experience do you have? ____________

33. How old are you? ____________ 34. Sex: Male Female

35. What is your city and country of birth? _________________

36. What is your city and country of residence? _________________

37. How many years have you lived in your country of residence? ____________

38. What is your first language? ______________________

Remarks:

THANK YOU!

10b Pre-training practitioner profile

Practitioner profile	
Individual characteristics: Current situation!	
What is your job title?	Job title:
How many reporting levels are there between you and the head of your organization?	Nr. Levels:
Do you supervise other employees in your organization or do other employees report to you?	Description position in hierarchy:
What is your experience in supporting groups	
What is your age?	
What is your gender?	
If you have experience in supporting group, please answer the following questions	
How many years have you worked in group support?	# Years ___________________________
Please estimate the number of workshops/meetings you have supported in these years.	# Workshops _______________________
Please estimate the number of different groups you worked with in these years?	# Different groups/clients ____________ (There can be different groups within an organization)
Please estimate the number of different organizations you worked with in these years?	# Different organizations ____________ (There can be different groups within an organization)
Please estimate the extent to which your current professional work consists of group support How was this in the past?	Current % group support role ______________________ Past % group support role ______________________
Can you give a few examples of different types of group processes that you supported over the years? (e.g. strategy building, requirement negotiation, etc.)	Examples: type of group processes:
How many different types of group processes did you support over the years?	# different types of group processes
Over the years, how many times (number/percentage) have you used interactive group support **technology** for any portion of the meeting?	# times:
If you use **technology** in the meeting, do you operate it yourself? (if not by whom?) If you operate the technology, do you combine this with process support or are these roles separated? Which role do you perform most?	Role separation
Do you design **processes** (agenda's) for the groups you support in advance?	Design

Do you use **process designs** or agenda's made by others? How are these transferred to you?	Design support
Do you use known process **methods/ facilitation techniques** (agenda items)? Where do you find/learn these?	Use existing methods
Do you adjust these **methods** to the specific process? How and when do you make these adjustments?	Adjustment Before, during the session, both
Do you also design new process **methods**? How and when do you design these methods?	always, frequently, sometimes, never
Do you teach/share those new **methods** to/with others? How do you teach/share these?	always, frequently, sometimes, never (teach/share)
	1=very unskilled 7=very skilled
How skilled do you consider yourself with respect to group support?	1 2 3 4 5 6 7

Please tell us about your formal educational background;
What is the highest diploma you have?
Did you follow other education that does not generally precede obtaining this diploma?
How did you learn to support groups as you do?
For process support?
For technology support?

THANK YOU

10 c Post training questionnaire

Post training questionnaire							
Name:							
1. What did you think of the manual? 2. What kind of additional information would you like to have? Why? 3. How should this information be presented? Why? 4. Do you find the manual (too) extensive? Why (not)? 5. Is there information that you think is superfluous? Why? 6. What else did you do/ are you going to do to prepare yourself for the facilitation of the session?							
Was the manual complete?	Not at all complete		complete			very complete	
	1	2	3	4	5	6	7
What did you think of the following aspects of the manual?	not at all useful		useful			very useful	
Assumption document	1	2	3	4	5	6	7
ThinkLets	1	2	3	4	5	6	7
Identification	1	2	3	4	5	6	7
Script	1	2	3	4	5	6	7
What-will-happen	1	2	3	4	5	6	7
Facilitation process model	1	2	3	4	5	6	7
Cue cards	1	2	3	4	5	6	7
Please estimate the following:	low			average		high	
How do you estimate the mental effort of preparation and training?	1	2	3	4	5	6	7
How difficult was it to prepare for the session?	1	2	3	4	5	6	7
How tiring was the preparation and training?	1	2	3	4	5	6	7
How difficult was the preparation and training compared to other trainings you did within the context of your study?	1	2	3	4	5	6	7
Do you feel equipped to facilitate the session?	Insufficient			Sufficient		Very equipped	
	1	2	3	4	5	6	7
Thanks!							

10d Post session Interview

Post session interview

Name:

1. Did the session go as expected?
2. Was the result as you expected?
3. Where did you deviate from the script? Why? What was the result?
4. What information did you miss in advance?
5. What training did you miss?
6. What kind of support did you miss?
7. Did you find the facilitation difficult?
8. What would you do different next time?

Please estimate the following	low		average			high	
What was the mental effort of facilitation?	1	2	3	4	5	6	7
How hard was it to facilitate?	1	2	3	4	5	6	7
How tiring was it to facilitate?	1	2	3	4	5	6	7

9. Was the task of facilitation as you expected?
10. How many hours did you spend on preparation after the training?
11. What did you do to prepare yourself?
12. What else would you do next time to prepare yourself?
13. What did you think of the quality of the results?
14. Was the quality of your session equal to the session you chauffeured?
15. Do you think you could train others to facilitate this session?

References

Abell, A. and Oxbrow, N. (2001) *Competing with Knowledge,* MPG Books Ltd., Bodmin, Cornwall.

Ackermann, F. (1996) "Participants Perceptions on the Role of Facilitators using Group Decision Support Systems", *Group Decision and Negotiation,* 5**,** 93-519.

Ackermann, F. and Eden, C. (2005) "Using Causal Mapping with Group Support Systems to Elicit an Understanding of Failure in Complex Projects: Some Implications for Organizational Research", *Group Decision and Negotiation,* 14**,** 355-37.

Ackoff, R.L. (1978) *The Art of Problem Solving,* John Wiley & Sons.

Acosta, C.E. and Guerrero, L.A. (2006) *Supporting the Collaborative Collection of User's Requirements* In *International Conference on Group Decision and Negotiation*, Karlsruhe, Universtatsverlag Karlsruhe.

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**,** 256-266.

Alaa, G., Appelman, J.H. and Fitzgerald, G. (2006) *Process Support for Agile Requirements Modeling and Maintenance of E-Projects* In *the Americas Conference on Information Systems*, Acapulco, Mexico, AIS.

Alexander, C. (1979) *The Timeless Way of Building,* Oxford University Press, New York.

Alexander, C. (1980) *The Nature of Order: An Essay on the Art of Building and the Nature of the Universe,* The center for environmental structure, Berkeley.

Alexander, C. (1996) *Key note at the ACM conference 1996*, http://www.patternlanguage.com/archive /ieee/ieeetext.htm.

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.

Antunes, P., Ho, T. and Carriço, L. (1999) *A GDSS Agenda Builder for Inexperienced Facilitators* In *10th Euro GDSS Workshop*, Copenhagen, Denmark, Delft University of Technology.

Appelman, J.H. and Driel, J. van (2005) *Crisis-Response in the Port of Rotterdam: Can we do Without a Facilitator in Distributed Settings?* In *Hawaii International Conference on System Science*, Los Alamitos, IEEE Computer Society Press.

Bacharach, S.B. (1989) "Organizational Theories: Some Criteria for Evaluation", *Academy of Management Review,* 14**,** 496-515.

Badler, N., Bindiganavale, R., Bourne, J., Palmer, M., Shi, J. and Schuler, W. (1998) *A Parameterized Action Representation for Virtual Human Agents* In *Workshop on Embodied Conversational Characters*, Lake Tahoe, California.

Badler, N.I., Palmer, M.S. and Bindiganavale, R. (1999) "Animation Control for Real-Time Virtual Humans", *Communications of the ACM,* 42**,** 65-73.

Baltes, B.B., Dickson, M.W., Sherman, M.P., Bauer, C.C. and LaGanke, J.S. (2002) "Computer-Mediated Communication and Group Decision Making: A Meta-Analysis", *Organizational Behavior and Human Decision Processes,* 87**,** 156–179.

Balthazard, P., Ferrell, W.R. and Aguilar, D.L. (1998) "Influence Allocation Methods in Group Decision Support Systems", *Group Decision and Negotiation,* 7**,** 347-362.

Beckman, K., Khajenoori, S., Coulter, N. and Mead, N.R. (1997) "Collaborations: Closing the Industy-Academia Gap", *IEEE Software***,** 49-57.

Belkin, N.J. and Croft, W.B. (1992) "Information Filtering and Information Retrieval: Two Sides of the Same Coin?" *Communications of the ACM,* 35**,** 29-38.

Beranek, P.M., Beise, C.M. and Niederman, F. (1993) *Facilitation and group support systems* In *Hawaii International Conference on System Sciences*, Wailea, IEEE Computer Society Press.

Bishop, J.W., Scott, K.D., Goldsby, M.G. and Cropanzano, R. (2005) "A Construct Validity Study of Commitment and Perceived Support Variables: A Multifoci Approach across Different Team Environments", *Group and Organization Management,* 30**,** 153-180.

Bjork- Ligon, E. and Bjork, R.A. (Eds.) (1996) *Memory Handbook of Perception and Cognition,* Academic Press, San Diego.

Blumer, H. (1969) *Symbolic Interactionism: Perspective and Method,* Prentice hall, Englewood Cliffs.

Boehm, B. (1988) "A Spiral Model of Software Development and Enhancement", *IEEE Computer,* 21**,** 61-72.

Boehm, B. and Ross, R. (1989) "Theory-W software project management: Principles and examples", *IEEE Transactions on Software Engineering,* 15**,** 902-916.
Boehm, B.W. (1981) *Software Engineering Economics,* Prentice-Hall, New-York.
Bostrom, R., Anson, R. and Clawson, V.K. (1993) In *Group Support Systems: New Perspectives*(Eds, Jessup, L.M. and Valacich, J.S.) Macmillan.
Bostrom, R.P. and Anson, R. (1992) In *Computer Augmented Teamwork, A Guided Tour*(Ed, Bostrom, R.P., R.T. Watson And S.T. Kinney) Van Nostrand Reinhold, New York, pp. 16-33.
Bostrom, R.P., Watson, R.T. and Kinney, S.T. (Eds.) (1992) *Computer Augmented Teamwork, A Guided Tour,* Van Nostrand Reinhold, New York.
Brady, R.H. (1967) "Computers in Top-Level Decision Making", *Harvard Business Review***,** 67-76.
Bragge, J. and Merisalo-Rantanen, H. (2005) *Insights on Developing a Collaboration Process for Gathering: Innovative End-User Feedback on Information Systems* In *11th Group Decision and Negotiation conference*, Vienna.
Bragge, J., Merisalo-Rantanen, H. and Hallikainen, P. (2005) "Gathering Innovative End-User Feedback for Continuous Development of Information Systems: A Repeatable and Transferable E-Collaboration Process", *IEEE Transactions on Professional Communication,* 48**,** 55-67.
Briggs, R.O. (1994) *The Focus Theory of Team Productivity and its Application to Development and Testing of Electronic Group Support Systems*, Tucson.
Briggs, R.O. (2006) *The Value Frequency Model: Towards a Theoretical Understanding of Organizational Change* In *International Conference on Group Decision and Negotiation*, Karlsruhe, Universtatsverlag Karlsruhe.
Briggs, R.O. and Grunbacher, P. (2001) *Surfacing Tacit Knowledge in Requirements Negotiation: Experiences using EasyWinWin* In *The 34th Hawaii International Conference On System Sciences*, Los Alamitos., IEEE Computer Society Press.
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**,** 151-196.
Briggs, R.O. and Gruenbacher, P. (2002) *EasyWinWin: Managing Complexity in Requirements Negotiation with GSS* In *Hawaii International Conference on System Sciences.*, Waikoloa, IEEE Computer Society Press.
Briggs, R.O., Kolfschoten, G.L. and Vreede, G.J. de (2005) *Toward a Theoretical Model of Consensus Building* In *Americas Conference on Information Systems*, Omaha,
Briggs, R.O., Kolfschoten, G.L. and Vreede, G.J., de (2006a) *Instrumentality Theory of Consensus* In *First HICSS Symposium on Case and Field Studies of Collaboration*, Kauai.
Briggs, R.O., Kolfschoten, G.L., Vreede, G.J. de and Dean, D.L. (2006b) *Defining Key Concepts for Collaboration Engineering* In *Americas Conference on Information Systems*, Acapulco, Mexico, AIS.
Briggs, R.O., Qureshi, S. and Reinig, B. (2004) *Satisfaction Attainment Theory as a Model for Value Creation* In *Hawaii International Conference On System Sciences*, Los Alamitos, IEEE Computer Society Press.
Briggs, R.O. and Reinig, B.A. (2007) *Bounded Ideation Theory: A New Model of the Relationship between Idea quantity and Idea-quality during Ideation* In *Hawaii International Conference on System Science*, Waikoloa, IEEE Computer Society Press.
Briggs, R.O. and Vreede, G.J., de (2001) *ThinkLets, Building Blocks for Concerted Collaboration,* Delft University of Technology, Delft.
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**,** 31-63.
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* In *Hawaii International Conference on System Sciences*, Los Alamitos, IEEE Computer Society Press.
Briggs, R.O., Vreede, G.J. de and Reinig, B. (2003b) *A Theory and Measurement of Meeting Satisfaction* In *Hawaii International Conference on System Sciences*, Los Alamitos, IEEE Computer Society Press.
Brunken, R., Plass, J.L. and Leutner, D. (2003) "Direct Measurement of Cognitive Load in Multimedia Learning", *Educational Psychologist,* 38**,** 53-61.
Buzan, T. (1974) *Use your Head,* British broadcasting organization, London.

Chan, W.T., Chau, D.K.H. and Lang, X. (1999) "Collaborative Scheduling over the internet", *Computer-Aided Civil and Infrastructure Engineering,* 14**,** 15-24.
Checkland, P.B. (1981) *Systems Thinking, Systems Practice,* John Wiley & Sons, Chichester.
Chen, H. , Hsu, P. , Orwig, R. , Hoopes, L. and Nunamaker, J. F. jr (1994) "Automatic Concept Classification of Text from Electronic Meetings", *Communications of the ACM,* 37**,** 56-73.
Cheng, K.E. and Deek, F.P. (2007) *A Framework for Studying Voting in Group Support Systems* In *Hawaii International Conference on System Sciences*, Waikoloa, IEEE Computer Society Press.
Churchman, C.W. (1971) *The Design of Inquiring Systems, Basic Concepts of Systems and Organizations,* Basic books, Inc publishers, New York.
Clawson, V.K., Bostrom, R. and Anson, R. (1993) "The Role of the Facilitator in Computer-Supported Meetings", *Small Group Research,* 24**,** 547-565.
Clawson, V.K. and Bostrom, R.P. (1995) *The Importance of Facilitator Role Behaviors in Different Face to Face Group Support Systems Environments* In *Hawaii International Conference on System Sciences*, Los Alamitos, IEEE Computer Society Press.
Cohen, J. (1988) *Statistical Power Analysis for the Behavioral Sciences,* Erlbaum, Hillsdale NJ.
Cooper, G., Tindall-Ford, S., Chandler, P. and Sweller, J. (2001) "Learning by Imagining", *Journal of Experimental Psychology: Applied,* 7**,** 68-82.
Couger, J. D. (1995) *Creative Problem Solving and Opportunity Finding,* Danvers, Mass: Boyd And Fraser.
Courtney, J.F. (2001) "Decision Making and Knowledge Management in Inquiring Organizations: toward a new Decision Making Paradigm for DSS", *Decision Support Systems,* 31**,** 17-38.
Craig, R.T. (1999) "Communication Theory as a Field", *Communication Theory,* 9**,** 119-161.
Creswell, J.W. (1994) *Research Design: Qualitative and Quantitative Approaches,* Sage Publications, Inc., Thousand Oaks, California.
Davenport, T.H. (1993) *Process Innovation: Reengineering Work through Information Technology,* Harvard Business School Press, Boston.
Davison, R.M. and Briggs, R.O. (2000) "GSS for Presentation Support: Supercharging the Audience through Simultaneous Discussions during Presentations", *Communications of the ACM,* 43**,** 91-97.
Dennis, A.R. and Wixom, B.H. (2002) "Investigating the Moderators of the Group Support Systems Use with Meta-Analysis", *Journal of Management Information Systems,* 18**,** 235-257.
Dennis, A.R., Wixom, B.H. and Vandenberg, R.J. (2001) "Understanding Fit and Appropriation Effects in Group Support Systems via Meta-Analysis", *Management Information Systems Quarterly,* 25**,** 167-183.
DeSanctis, G. and Gallupe, R.B. (1987) "A Foundation for the Study of Group Decision Support Systems", *Management Science,* 33**,** 589-609.
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**,** 51-72.
Dictionary.com (2007) *design. (n.d.).* http://dictionary.reference.com/browse/design.
Drucker, P.F. (1967) *The Effective Executive*, London.
Drucker, P.F. (1969) *The Age of Discontinuity,* William Heinemann Ltd, London.
Eden, C. (1992) "A Framework for Thinking about Group Decision Support Systems (GDSS)", *Group Decision and Negotiation,* 1**,** 199-218.
Education Encyclopedia (2002) In *Encyclopedia of Education* the Gale Group, Inc.
Ellis, C.A., Gibbs, S.J. and Rein, G.L. (1991) "Groupware: Some Issues and Experiences", *Communications of the ACM,* 34.
Enserink, B. (2003) *Creating a Scenario logic - Design and Application of a Repeatable Methodology* In *Hawaii International Conference on System Sciences*, Los Alamitos, IEEE Computer Society Press.
Ericsson, K.A. and Simon, H.A. (1993) *Protocol Analysis: Verbal Reports as Data,* MIT Press, Cambridge.
Evans, P. and Wolf, B. (2005) "Collaboration Rules", *Harvard Business Review,* 7**,** 1-11.
Facilitators, International Association of (2004) "Statement of Values and Code of Ethics for Facilitators".
Fjermestad, J. and Hiltz, S.R. (1999) "An Assessment of Group Support Systems Experimental Research: Methodology and Results", *Journal Of Management Information Systems,* 15**,** 7-149.

Fjermestad, J. and Hiltz, S.R. (2001) "A Descriptive Evaluation of Group Support Systems Case and Field Studies", *Journal Of Management Information Systems,* 17**,** 115-159.
Fruhling, A. and Vreede, G.J. de (2005) In *Value Based Software Engineering* (Eds, Biffl, S., Aurum, A., Boehm, B., Erdogmus, H. and Gruenbacher, P.) Springer-Verlag, Berlin, pp. 201-223.
Gamma, E., Helm, R., Johnson, R. and Vlissides, J. (1995) *Elements of Reusable Object-Oriented Software,* Addison-Wesley Publishing Company.
Gardner, H. (1983) *Frames of Mind: The Theory of Multiple Intelligences,* Basic, New York.
Garfield, M.J., Taylor, N.J., Dennis, A.R. and Satzinger, J.W. (2001) "Research Report: Modifying Paradigms- Individual Differences, Creativity Techniques, and Exposure to Ideas in Group Idea Generation", *Information Systems Research,* 12**,** 322-333.
Gavish, B. and Gerdes, J.H. jr (1997) "Voting Mechanisms and their implications in a GDSS environment", *Annals of Operation Research,* 71**,** 41-74.
Glaser, B. and Strauss, A. (1967) *The Discovery of Grounded Theory,* Aldine, Chicago.
Gog, T. van, Paas, F. and Merrienboer, J.J.G. van (2004) "Process-Oriented Worked Examples: Improving Transfer Performance Trough Enhanced Understanding", *Instructional Science,* 32**,** 83-98.
Griffith, T.L., Fuller, M.A. and Northcraft, G.B. (1998) "Facilitator Influence in Group Support Systems", *Information Systems Research,* 9**,** 20-36.
Gronemann, B., Joeris, G., Scheil, S., Steinfort, M. and Wache, H. (1999) *Supporting Cross-Organizational Engineering Processes by Distributed Collaborative Workflow Management - The MOKASSIN approach* In *International Conference on Concurrent Multidisciplinary Engineering*, Bremen, Germany.
Grossman, G.M. and Helpman, E. (1991) *Innovation and Growth in the Global Economy,* The MIT Press, Cambridge.
Grover, V. (1999) "From Business Reengineering to Business Process Change Management: A Longitudinal Study of Trends and Practices", *IEEE Transactions on Engineering Management,* 46**,** 36-46.
Grover, V. and Kettinger, W.J. (1995) *Business Process Change; Reengineering Concepts, Methods and Technologies,* Idea group Publishing, Harrisburg.
Grudin, J. (1994) "Computer Supported Cooperative Work: History and focus", *Computer,* May**,** 19-26.
Harder, R.J. and Higley, H. (2004) *Application of ThinkLets to Team Cognitive Task Analysis* In *Hawaii International Conference on System Sciences*, Los Alamitos, IEEE Computer Society Press.
Harder, R.J., Keeter, J.M., Woodcock, B.W., Ferguson, J.W. and Wills, F.W. (2005) *Insights in Implementing Collaboration Engineering* In *Hawaii International Conference on System Science*, Los Alamitos, IEEE Computer Society Press.
Harper, D (2001) *Online Etymology Dictionary*, http://www.etymonline.com/.
Harrison, B.N. and Coplien, J.O. (1996) "Patterns of Productive Software Organizations", *Bell Labs Technical Journal*.
Hayne, S.C. (1999) "The Facilitator's Perspective on Meetings and Implications for Group Support Systems Design", *DataBase,* 30**,** 72-91.
Hender, J.M., Dean, D.L., Rodgers, T.L. and Nunamaker, J.F. jr (2002) "An Examination of the Impact of Stimuli Type and GSS Structure on Creativity: Brainstorming Versus Non-Brainstorming Techniques in a GSS Environment", *Journal of Management Information Systems,* 18**,** 59-85.
Hengst, M. den, Adkins, M., Keeken, S. J. van and Lim, A.S.C. (2005) *Which facilitation functions are most challenging:A global survey of facilitators* In *Group Decision and Negotiation conference*, Vienna.
Heninger, W.G., Dennis, A.R. and McNamara Hilmer, K. (2006) "Individual Cognition and Dual-Task Interference in Group Support Systems", *Information Systems Research,* 17**,** 415-424.
Huber, G.P. (1984) "The Nature and Design of Post-Industrial Organizations", *Management Science,* 30**,** 928-951.
Hwang, M.I. (1998) "Did Task Type Matter in the Use of Decision Room GSS? A Critical Review and a Meta-analysis", *International Journal of Management Science,* 26**,** 1-15.
IEEE (1990) "IEEE Std 610.12 Glossary of Software Engineering Terminology", *IEEE Standard*.
INCOSE (2007) *What is Systems Engineering?* http://www.incose.org/practice/whatissystemseng.aspx.
International Association of Facilitators (2004) *IAF Website*, http://www.IAF-world.org.
Jackson, M.A. (1983) *System Development,* Prentice-Hall, Englewood Cliffs.

Janis, I.L. (1972) *Victims of Groupthink: A Psychological Study of Foreign-Policy Decisions and Fiascoes,* Houghton Mifflin Company, Boston.
Jarvenpaa, S.L. and Stoddard, D.B. (1998) "Business Process Redesign: Radical and Evolutionary Change", *Journal of Business Research,* 41**,** 15-27.
Jassawalla, A.R. and Sashittal, H.C. (1998) "An Examination of Collaboration in High-Technology New Product Development Processes", *Journal of Product Innovation Management,* 15**,** 237-254.
Jenkins, J. (2005) *IAF Methods Database*, www.iaf-methods.org.
Jung, D.I. and Sosik, J.J. (2003) "Group Potency", *Group and Organization Management,* 28**,** 366-391.
Kalyuga, S., Ayres, P., Chandler, P. and Sweller, J. (2003) "The Expertise Reversal Effect", *Educational Psychologist,* 38**,** 23-31.
Kamal, M., Davis, A.J., Nabukenya, J., Schoonover, T.V., Pietron, L.R. and Vreede, G.J. de (2007) *Collaboration Engineering for Incident Response Planning: Process Development and Validation* In *Hawaii International Conference on System Science*, Waikoloa, IEEE Computer Society Press.
Kempenaers, J., Pinte, J., Detand, J. and Kruth, J.P. (1996) "A Collaborative Process Planning and Scheduling System", *Advances in Engineering Software,* 25**,** 3-8.
Kester, L., Lehnen, C, Gerven, P.W.M. van and Kirschner, P.A. (2006) "Just-in-Time, Schematic Supportive Information Presentation During Cognitive Skill Acquisition", *Computers in Human Behavior,* 22**,** 93-112.
Kettinger, W.J. and Teng, J.T.C. (1997) "Business process change: a study of methodologies, techniques and tools", *Management Information Systems Quarterly,* 21**,** 55-80.
Kirschner, P.A. (2002) "Cognitive load theory: Implications of cognitive load theory on the design of learning", *Learning and Instruction,* 12**,** 1-10.
Klein, H.K. and Myers, M.D. (1999) "A Set of Principles for Conducting and Evaluating Interpretive Field Studies in Information Systems", *Management Information Systems Quarterly,* 23**,** 67-94.
Kolfschoten, G.L. (2007) *A Toolbox for the Facilitation of E-Democracy* In *Group Decision and Negotiation Conference*, Mt. Tremblant, Concordia University.
Kolfschoten, G.L., Appelman, J.H., Briggs, R.O. and Vreede, G.J. de (2004a) *Recurring Patterns of Facilitation Interventions in GSS Sessions* In *Hawaii International Conference On System Sciences*, Los Alamitos, IEEE Computer Society Press.
Kolfschoten, G.L., Briggs, R.O., Appelman, J.H. and Vreede, G.J. de (2004b) *ThinkLets as Building Blocks for Collaboration Processes: A Further Conceptualization* In *CRIWG*, San Carlos, Costa Rica, Springer-Verlag.
Kolfschoten, G.L., Briggs, R.O. and Vreede, G.J. de (2007) "Computer Aided Pattern-Based Collaboration Process Design: A Computer Aided Collaboration Engineering Tool", *Lecture Notes in Computer Science,* 4715.
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**,** 611-621.
Kolfschoten, G.L., Hengst, M. den and Vreede, G.J. de (2005) *Issues in the Design of Facilitated Collaboration Processes* In *Group Decision and Negotiation Conference*, Vienna,
Kolfschoten, G.L. and Houten, S.P.A. van (2007) *Predictable Patterns in Group Settings through the use of Rule Based Facilitation Interventions* In *Group Decision and Negotiation conference*, Mt Tremblant, Concordia University.
Kolfschoten, G.L., Lowry, P.B., Dean, D. L. and Kamal, M. (working paper) "A Measurement Framework for Patterns of Collaboration".
Kolfschoten, G.L., Niederman, F., Vreede G.J., de and Briggs, R.O. (2006b) *Understanding the Job Requirements and Roles for Group Support Systems Facilitators* In *SIGMIS Computer Personnel Research Conference*.
Kolfschoten, G.L. and Rouwette, E. (2006a) *Choice Criteria for Facilitation Techniques* In *First HICSS Symposium on Case and Field Studies of Collaboration*, Kauai,
Kolfschoten, G.L. and Rouwette, E. (2006b) *Choice Criteria for Facilitation Techniques: A Preliminary Classification* In *International Conference on Group Decision and Negotiation*, Karlsruhe, Universtatsverlag Karlsruhe.
Kolfschoten, G.L. and Santanen, E.L. (2007) *Reconceptualizing Generate ThinkLets: the Role of the Modifier* In *Hawaii International Conference on System Science*, Waikoloa, IEEE Computer Society Press.

Kolfschoten, G.L. and Veen, W. (2005) *Tool Support for GSS Session Design* In *Hawaii International Conference on System Sciences*, Los Alamitos, IEEE Computer Society Press.
Krone, K.J., Jablin, F.M. and Putnam, L.L. (1987) In *Handbook of Organizational Communication: an Interdisciplinary Perspective* (Eds, Jablin, F.M., Putnam, L.L., Roberts, K.H. and Porter, L. W.) Sage Publications, Newbury Park.
Lehman, M.M. (1989) "Uncertainty in Computer Applications and its Control trough the Engineering of Software", *Journal of Software Maintenance,* 1**,** 3-28.
Levin, J. and Nalebuff, B. (1995) "An Introduction to Vote-Counting Schemes", *Journal of Economic Perspectives,* 9**,** 3-26.
Lin, A.C. (1998) "Bridging Positivist and Interpretivists Approaches to Qualitative Methods", *Policy Studies Journal,* 26**,** 162-180.
Locke, E.A. and Latham, G.P. (1990) *A Theory of Goal Setting and Task Performance,* Prentice Hall, Englewood Cliffs.
Lowry, P.B., Curtis, A. and Lowry, M.R. (2004) "Building a Taxonomy and Nomenclature of Collaborative Writing to Improve Interdisciplinary Research and Practice", *Journal Of Business Communication,* 41**,** 66-99.
Lukosch, S. and Schümmer, T. (2006) "Groupware Development Support with Technology Patterns", *International Journal of Human Computer Systems,* 64.
Malhotra, Y. and Galletta, D. (2005) "A Multidimensional Commitment Model of Volitional Systems Adoption and Usage Behavior", *Journal of Management Information Systems,* 22**,** 117-151.
Malone, T.W. and Crowston, K. (1994) "The Interdisciplinary Study of Coordination", *ACM Computing Surveys,* 26**,** 87-119.
Massey, A.P., Montoya-Weiss, M.M. and Hung, Y-T. (2003) "Because Time Matters: Temporal Coordination in Global Virtual Project Teams", *Journal of Management Information Systems,* 19**,** 129-155.
May, D. and Taylor, P. (2003) "Knowledge Management with Patterns: Developing techniques to improve the process of converting information to knowledge", *Communications of the ACM,* 44**,** 94-99.
Mayer, R.E. and Moreno, R. (2002) "Aids to Computer-based Multimedia Learning", *Learning and Instruction,* 12**,** 107-119.
Mayer, R.E. and Moreno, R. (2003) "Nine ways to reduce Cognitive Load in Multimedia Learning", *Educational Psychologist,* 28**,** 43-52.
McConnell (1993) *The Trainer's Dictionary,* HRD Press, Amherst, Mass.
McGrath, J.E. (1991) "Time, Interaction and Performance (TIP): A Theory of Groups", *Small Group Research,* 22**,** 147-174.
McLeod, P.L. (1992) "An Assessment of the Experimental Literature on Electronic Support of Group Work: Results of a Meta-Analysis", *HUMAN-COMPUTER INTERACTION,,* 7**,** 257-280.
Meijers, K. (2005) *Mini Turbines in Beweging*, www.tbm.tudelft.nl/live/binaries/e857ec2b-ca87-4360-aa70-d8a068d6bba7/doc/K.Meijers.pdf.
Merrienboer, J.G. van, Kirschner, P.A. and Kester, L. (2003) "Taking the Load off a Learner's Mind: Instructional Design for Complex Learning", *Educational Psychologist,* 38**,** 5-13.
Merrienboer, J.J.G. van, Schuurman, J.G., Croock, M.B.M. de and Paas, F.G.W.C. (2002) "Redirecting Learners' Attention during Training: Effects on Cognitive Load, Transfer Test Performance and Training Efficiency", *Learning and Instruction,* 12**,** 11-37.
Meyer, J.P. and Allen, N.J. (1991) "A Thee-Component Conceptualization of Organizational Commitment: Some Methodological Considerations", *Human Resource Management Review,* 1**,** 61-98.
Meyer, J.P. and Herscovitch, L. (2001) "Commitment in the Workplace toward a General Model", *Human Resource Management Review,* 11**,** 299-326.
Miller, G.A. (1956) "The magical number seven plus or minus two: some limits on our capacity for processing information", *Psychological Review,* 63**,** 81-97.
Mintzberg, H. (1983) *Structure in Fives: Designing Effective Organizations,* Prentice-Hall Inc., Englewood Cliffs.
Mitroff, I.I., Betz, F., Pondly, L.R. and Sagasty, F. (1974) "On Managing Science In The Systems Age: Two Schemas For The Study Of Science As A Whole Systems Phenomenon", *TIMS Interfaces,* 4**,** 46-58.
Moreno, R. (2004) "Decreasing Cognitive Load for Novice Students: Effects of Explanatory versus Corrective Feedback in Discovery-Based Multimedia", *Instructional Science,* 32**,** 99-113.

Moreno, R. and Valdez, A. (2005) "Cognitive Load and Learning Effects of Having Students Organize Pictures and Words in Multimedia Environments: the Role of Student Interactivity and Feedback", *Educational Technology Research and Development,* 53**,** 35-45.
Mulder, I., Swaak, J. and Kessels, J. (2002) "Assessing learning and shared understanding in technology-mediated interaction", *Educational Technology & Society,* 5**,** 35-47.
Munkvold, B.E. and Anson, R. (2001) *Organizational Adoption and Diffusion of Electronic Meeting Systems: A Case Study* In *Proceedings of ACM GROUP01*, New York, ACM.
Newell, A. and Simon, H. A. (1972) *Human problem solving,* Prentice-Hall, Englewood Cliffs.
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**,** 1-22.
Niegemann, H.M. and Domagk, S. (2005) *ELEN Project Evaluation Report*, http://www2tisip.no/E-LEN.
Nonaka, I. (1994) "A Dynamic Theory of Organizational Knowledge Creation", *Organizational Science,* 5**,** 14-37.
Nonaka, I. and Takeuchi, H. (1995) *The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation,* Oxford University Press, New York.
Ntuen, C.A., Leedom, D. and Schmeisser, E. (2005) "Performance of Individuals and Groups in Sensemaking of Information from Computer Decision Aiding Systems".
Nunamaker, J.F. Jr., Briggs, R.O., Mittleman, D.D., Vogel, D. and Balthazard, P.A. (1997) "Lessons from a Dozen Years of Group Support Systems Research: A Discussion of Lab and Field Findings", *Journal of Management Information Systems,* 13**,** 163-207.
Nunamaker, J.F., jr., Dennis, A., Valacich, J., Vogel, D. and George, J.F. (1991) "Electronic Meeting Systems to Support Group Work", *Communications of the ACM,* 34**,** 40-61.
Osborn, A.F. (1953) *Applied Imagination,* Scribners, New York.
Paas, F. and Merrienboer, J.J.G. van (1994) "Variability of Worked Examples and Transfer of Geometrical Problem-Solving skills: A Cognitive Load Approach", *Journal of Educational Psychology,* 86**,** 122-133.
Paas, F., Renkl, A. and Sweller, J. (2003a) "Cognitive Load Theory and Instructional Design, Recent Developments", *Educational Psychologist,* 38**,** 1-4.
Paas, F.G.W.C., Tuovinen, J.E., Tabbers, H. and Gerven, P.W.M. van (2003b) "Cognitive Load Measurement as a Means to Advance Cognitive Load Theory", *Educational Psychologist,* 38**,** 63-71.
Pollock, E., Chandler, P. and Sweller, J. (2002) "Assimilating Complex Information", *Learning and Instruction,* 12**,** 61-86.
Popper, K.R. (1959) *The Logic of Scientific Discovery,* Hutchinson & Co, London.
Post, B.Q. (1993) "A Business Case Framework for Group Support Technology", *Journal of Management Information Systems,* 9**,** 7-26.
Qureshi, S. and Keen, P. (2004) "Activating Knowledge through Electronic Collaboration: Vanquishing the Knowledge Paradox", *IEEE Transactions in Professional Communication*.
Renkl, A. and Atkinson, R.K. (2003) "Structuring the Transition from Example Study to Problem Solving in Cognitive Skill Acquisition: a Cognitive Load Perspective", *Educational Psychologist,* 38**,** 15-22.
Rich, E. (1983) *Artificial Intelligence,* McGraw-Hill, Singapore.
Rikers, R.M.J.P., Gerven, P.W.M. van and Schmidt, H.G. (2004) "Cognitive Load Theory as a Tool for Expertise Development", *Instructional Science,* 32**,** 173-182.
Rising, L. (2001) *Design Patterns in Communication Software,* Cambridge University Press, Cambridge.
Romano, N.C. Jr., Chen, F. and Nunamaker, J.F. jr (2002) *Collaborative Project Management Software* In *Hawaii International Conference on System Sciences*, Wikoloa, IEEE Computer Society Press.
Roschelle, J. and Teasley, S.D. (1995) In *Computer Supported Collaborative Learning*(Ed, O'Malley, C.) Springer-Verlag, Berlin.
Rouwette, E.A.J.A., Vennix, J.A.M. and Mullekom, T. van (2002) "Group Model Building Effectiveness: a Review of Assessment Studies", *System Dynamics Review,* 18**,** 5-45.
Santanen, E.L. (2005) *Resolving Ideation Paradoxes: Seeing Apples as Oranges through the Clarity of ThinkLets* In *Hawaii International Conference on System Sciences*, Los Alamitos, IEEE Computer Society Press.

Santanen, E.L., G.L., Kolfschoten and Golla, K. (2006) *The Collaboration Engineering Maturity Model* In *Hawaiian Internal Conference on System Sciences*, Los Alamitos, IEEE Computer Society Press.
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**,** 167 -197.
Schank, R.C., Fano, B., Bell, L. and Jonea, M.Y. (1993) "The Design of Goal Based Scenario's", *The Journal of the Learning Sciences,* 3**,** 305-345.
Schenk, K.D., Vitalari, N.P. and Davis, K.S. (1998) "Differences Between Novice and Expert Systems Analysts: What Do We Know and What Do We Do?" *Journal of Management Information Systems,* 15**,** 9-50.
Schoemaker, P. (1982) "The Expected Utility Model: Its Variants, Purposes, Evidence and Limitations", *Journal of Economic Literature,* 20**,** 529–63.
Schrage, M. (1990) *Shared Minds: The New Technologies of Collaboration,* Random House, New York.
Schuman, S. (2006) *Grp-Facl*, http://www.albany.edu/cpr/gf/.
Schwarz, R.M. (1994) *The Skilled Facilitator,* Jossey-Bass Publishers, San Francisco.
Shah, J.J. and Vargas-Hernandez, N. (2003) "Metrics for Measuring Ideation", *Design Studies,* 24**,** 111-134.
Simon, H.A. (1960) *The New Science of Management Decision,* Prentice Hall, New York.
Simon, H.A. (1973) "The Structure Of Ill Structured Problems", *Artificial Intelligence,* 4**,** 181-201.
Smith, J.M., Smith, J.M. and Smith, D.C.P. (1977) "Database Abstractions: Aggregation and Generalization", *ACM Transactions on Database Systems,* 2**,** 105-133.
Sol, H.G. (1982) *Simulation in information systems development,* Rijksuniversiteit Groningen, Groningen, the Netherlands.
Strauss, A.L. and Corbin, J. (1998) *Basics of Qualitative Research, Techniques and Procedures for Developing Grounded Theory,* Sage, London.
Sweller, J. (1988) "Cognitive load during problem solving: Effects on learning", *Cognitive Science,* 12**,** 257-285.
Sweller, J. (2004) "Instructional Design Consequences of an Analogy between Evolution by Natural Selection and Human Cognitive Architecture", *Instructional Science,* 32**,** 9-31.
Sweller, J., Merrienboer, J.G. van and Paas, F.G.W.C. (1998a) "Cognitive Architecture and Instructional Design", *Educational Psychology Review,* 10**,** 251-296.
Sweller, J., Merrienboer, J.J.G. van and Paas, F.G.W.C. (1998b) "Cognitive Architecture and Instructional Design", *Educational Psychology Review,* 10**,** 251-296.
Terveen, L.G. (1994) "Overview of human-computer collaboration", *Knowledge-Based Systems,* 8**,** 67-81.
Thompson, J.D. (1967) *Organizations in Action,* McGraw-Hill, New York.
Trauth, E.M. and Jessup, L.M. (2000) "Understanding Computer-Mediated Discussions: Positivist and Interpretive Analysis of Group Support System Use", *Management Information Systems Quarterly,* 24**,** 43-79.
Trochim, W.M. (2000) *The Research Methods Knowledge Base*, http://trochim.human.cornell. edu/kb/index.htm.
Tyran, C.K. and Sheperherd, M. (1998) *GSS and Learning Research: A Review and Assessment of the Early Studies* In *Hawaii International Conference on System Science*, IEEE Computer Society Press.
Veld, J. In 't (1987) *Analyse van Organisatie Problemen,* Stenfert Kroese, Leiden.
Vreede, G.J. de (1997) "Collaborative Business Engineering with Animated Electronic Meetings", *Journal of Management Information Systems,* 14**,** 141-164.
Vreede, G.J. de , Fruhling, A. and Chakrapani, A. (2005) *A Repeatable Collaboration Process for Usability Testing* In *Hawaii International Conference on System Sciences*, Los Alamitos, IEEE Computer Society Press.
Vreede, G.J. de, Boonstra, J. and Niederman, F.A. (2002) *What is Effective GSS Facilitation? A Qualitative Inquiry into Participants' Perceptions* In *Hawaiian International Conference on System Science*, Los Alamitos, IEEE Computer Society Press.
Vreede, G.J. de and Briggs, R.O. (2001) *ThinkLets: Five Examples Of Creating Patterns Of Group Interaction* In *Group Decision & Negotiation*, La Rochelle, France.

Vreede, G.J. de and Briggs, R.O. (2005) *Collaboration Engineering: Designing Repeatable Processes for High-Value Collaborative Tasks* In *Hawaii International Conference on System Science*, Los Alamitos, IEEE Computer Society Press.
Vreede, G.J., de, Briggs, R.O. and Kolfschoten, G.L. (2006a) "ThinkLets: A Pattern Language for Facilitated and Practitioner-Guided Collaboration Processes", *International Journal of Computer Applications in Technology,* 25**,** 140-154.
Vreede, G.J. de and Bruijn, J.A. de (1999) "Exploring the Boundaries of Successful GSS Application: Supporting Inter-Organizational Policy Networks." *DataBase,* 30**,** 111-131.
Vreede, G.J. de, Davison, R. and Briggs, R.O. (2003a) "How a Silver Bullet May Lose its Shine - Learning from Failures with Group Support Systems", *Communications of the ACM,* 46**,** 96-101.
Vreede, G.J. de, Koneri, P.G., Dean, D.L., Fruhling, A.L. and Wolcott, P. (2006b) "Collaborative Software Code Inspection: The Design and Evaluation of a Repeatable Collaborative Process in the Field", *International Journal of Cooperative Information Systems,* 15**,** 205-228.
Vreede, G.J. de, Vogel, D.R. , Kolfschoten, G.L. and Wien, J.S. (2003b) *Fifteen Years of in-situ GSS Use: A Comparison Across Time and National Boundaries* In *Hawaii International Conference on System Science*, Los Alamitos, IEEE Computer Society Press.
Weber, R. (2004) "The Rhetoric of PositivismVersus Interpretivism a Personal View, Editor's Comments", *Management Information Systems Quarterly,* 28**,** iii-xii.
Weick, K.E. (1995) *Sensemaking in Organizations,* Sage Publications Inc., Thousand Oaks, CA.
Weiss, J. and Hughes, J. (2005) "Want Collaboration? Accept - and Actively Manage -Conflict", *Harvard Business Review,* 3**,** 1-12.
Wester, F. and Peters, V. (2004) *Kwalitatieve Analyse. Uitgangspunten en Procedures.,* Coutinho, Bussum.
Wheeler, B.C. and Valacich, J.S. (1996) "Facilitation, GSS and Training as Sources of Process Restrictiveness and Guidance for Structured Group Decision Making an Empirical Assessment", *Information Systems Research,* 7**,** 429-450.
Winston, P. H. (1984) *Artificial Intelligence,* Addison-Wesley, Reading.
Wood, R.E. (1986) "Task Complexity: Definition of the Construct", *Organizational behavior and Human Decision Processes,* 37**,** 60-82.
Yoong, P. (1995) "Assessing Competency in GSS Skills: A Pilot Study in the Certification of GSS Facilitators", *SIGCPR.*
Zhang, J. (1997) "The Nature of External Representations in Problem Solving", *Cognitive Science,* 21**,** 179-217.
Zigurs, I. and Buckland, B. (1998) "A Theory of Task/Technology Fit and Group Support Systems Effectiveness", *Management Information Systems Quarterly,* 22**,** 313-334.

Author index

Abell	45
Ackermann	12,19,21,70,91,161,193-195
Ackoff	12,27,106
Acosta	166,190
Agres	6,13,16
Alaa	141
Alexander	45,142-145,149-151
Antunes	107
Appelman	141,166
Bacharach	48
Badler	45,146-148
Baltes	32,196
Balthazard	161
Beckman	3
Belkin	160
Beranek	12
Bishop	39
Bjork- Ligon	45,63
Blumer	56
Boehm	27,45,53,109
Bostrom	9-11,31,45,111,138
Brady	106
Bragge	1,141,166,190
Briggs	1-269
Brunken	77
Buzan	63,81
Chan	161
Checkland	19,27,45,106,161
Chen	161
Cheng	161
Churchman	2,48
Clawson	9,11,19,21,31,70,193-195
Cohen	175,182
Cooper	93
Couger	19,106
Courtney	2
Craig	145
Creswell	49,50
Davenport	22,45,106
Davison	10
Dennis	5,11-13,32,107,196
DeSanctis	5,13,45
Dickson	19-21,70,193-195
Drucker	1,12,106
Eden	161
Ellis	5,13,45
Enserink	141
Ericsson	56
Evans	2
Fjermestad	5,6,10,11,32,45,196
Fruhling	166
Gamma	45,142,144,150,151
Gardner	67,75
Garfield	11
Gavish	161
Glaser	56
Gog	93
Griffith	45
Gronemann	161
Grossman	1,13
Grover	22,45
Grudin	45
Harder	141,166
Harper	4,13
Harrison	142
Hayne	9,19,21,31,70,107,193-195
Hender	11
Hengst	66
Heninger	192
Huber	1
Hwang	32,196
Jackson	45
Janis	5,39,71
Jarvenpaa	22
Jassawalla	5
Jenkins	12,116
Jung	39
Kalyuga	91,137
Kamal	159,190
Kempenaers	161
Kester	82,92,93
Kettinger	22,45
Kirschner	64,70
Klein	47
Krone	45,145
Lehman	109
Levin	161
Lin	58
Locke	3,34,38,39,45,121
Lowry	159,161
Lukosch	142
Malhotra	38
Malone	4
Massey	5
May	142
Mayer	36,38,74,80-82
McConnell	91
McGrath	5,13
McLeod	32,196
Meijers	1
Merrienboer	65,66,75,76,90,92,93
Meyer	36,38,45
Miller	63
Mintzberg	2,12

Mitroff	19,27,106	Schrage	3
Moreno	74,75,80-82,92,94	Schuman	55
Mulder	160	Schwarz	5,6,13,45
Munkvold	6	Shah	161
Newell	147	Simon	19,27,56,406,147
Niederman	7,19,20,21,52,70,194,195	Smith	160
Nonaka	1,2,13,45	Sol	19
Ntuen	160	Strauss	50,56
Nunamaker	5-20,28,31,45,107,110	Sweller	45,63,64,74,75,93,189
Osborn	159	Terveen	3
Paas	63,64,90,98	Thompson	3
Pollock	70,73,74,80,92	Trauth	47,48
Popper	48	Trochim	49,50
Post	6,12,91	Tyran	32,196
Qureshi	1,12	Veld	33,34,45
Renkl	93	Vreede	1-269
Rich	147	Weber	47,48
Rikers	94	Weick	160
Rising	142	Weiss	3,13
Romano	161	Wester	56
Roschelle	2,3	Wheeler	107
Rouwette	55,161	Winston	147
Santanen	11,15,141,155,-166,189-192	Wood	70
Schank	166	Yoong	12
Schoemaker	39	Zhang	147
		Zigurs	107

Subject Index

Acceptance	42,43,54,105,109,119,138,150,165,187
Agenda	20,57,108,117,129,134-137
Anonymity	10
Apprenticeship	12
Appropriation	11
CACE tool	190,191
Choice map	114,115,128,129,133,137
Clarify	24,125,160,163
Cognitive load	63-77,89,90-94,98-101,126,170,192
Collaboration	1-5,32-40
Collaboration engineer	8,9,19,20
Collaboration Engineering	7-10,15-29,186-189
Collaboration Engineering approach	7-10,22-29
Collaboration process design	78-82,106-138,207-216
Collaboration process prescription	72-89, 207-216, 216-226
Collaboration process prescription template	82-89, 207-216
Collaboration support	5-7
Commitment	35-40, 167,169,175,181,238,239
Consensus	24,125,162-164
Cooperation	4,5
Coordination	4,5
Cue card	82,157,158
Deployment	22,23,27-29
Design	19,20,22,23,25,105-138,186-188
Design approach	19,20,22,23,25,105-138
Design pattern	142-145
Design support	120-138
Effectiveness	33,34,40,167,169,175,181,238,239
Efficaciousness	42,43,54,105,109,117,118,138,150,164,165,187
Efficiency	34,35,40,167,169,175,181,238,239
Engineerability	29,186-188
Evaluate	24,125,162,163
Execution load	67
Extraneous cognitive load	64,65,75-77
Facilitation	6-8,17-21,193-195
Facilitation methods	11,12
Facilitation process model	79,134-137,157,207
Facilitator	6-8,17-21,193-195
Generate	24,124,159,160,162
Germane cognitive load	64,65,76,77
Group	5,110
Group Support System	5,10
Groupware	5
Innovation	2
Intervention	145-149
Intrinsic cognitive load	64,65,70-75,77,92-94
Knowledge creation	2
Master thinkLet	152-157,233-237
Memory schema	63-65,76
Mental effort	63
Organize	24,125,161,163
Parallelism	10

Pattern classification	124,125,137
Pattern language	142-145,151
Patterns of collaboration	24,124,125,159-164
Practitioner	8,9,17-19,82-90,171-175,177-182,217-226
Predictability	58,119,120,141,142,150,166,188
Productivity	35,40,167,169,175,181,238,239
Quality of collaboration	32-41,167,169,175,181,238,239
Quality of design	41-43, 117-120
Reduce	24,124,125,160,163
Result classification	125-127,137
Reusability	58,118,119, 141,142,150,165,187
Rule based interventions	145-149,155,233
Satisfaction	37,40,41,167,169,175,181,238,239
Script	79-81,207-216,233-235
Sustained	15,28
Task of the collaboration engineer	8,9,17-20
Task of the practitioner	8,9,17-21,70-72,82-90,171-175,177-182,217-226
Theory on collaboration	37-41
Theory on collaboration process design	41-43
ThinkLet	9,77-82,149-166
ThinkLet design pattern	149-152,158,159,232-237
ThinkLet instantiation template	77-82,157,158,208-215,232-237
Training	90-101
Training approach	94-101
Training load	66
Transfer	27,28,102,103
Transfer approach	27,102,103
Transferability	53,54,65-103,119,150,166,188
Validation	117-120,130

Summary

Collaboration is important for knowledge creation and innovation and therefore for the competitiveness of organizations in a knowledge economy (Drucker, 1969, Mintzberg, 1983, Qureshi and Keen, 2004, Grossman and Helpman, 1991, Nonaka, 1994). We define collaboration as joint effort towards a goal (Harper, 2001). Collaboration is challenging and groups cannot overcome the challenges of collaboration by themselves (Ellis et al., 1991, Nunamaker et al., 1997, Weiss and Hughes, 2005, DeSanctis and Gallupe, 1987, McGrath, 1991, Schwarz, 1994). In order to increase the quality of collaboration, collaboration support is required (Nunamaker et al., 1997, Schwarz, 1994, Dennis et al., 2001).

Collaboration support can be offered by technology and by process support such as facilitation. For recurring tasks, it is difficult for organizations to implement sustained collaboration support (Agres et al., 2005, Briggs et al., 2003a). Collaboration Engineering is an approach to create sustained collaboration support by designing collaborative work practices for high-value recurring tasks, and deploying those as collaboration process prescriptions for practitioners to execute for themselves without ongoing support from professionals (Briggs et al., 2006b). In Collaboration Engineering the traditional facilitation role is split up in a design and execution role, which is easier to sustain in organizations (Briggs et al., 2003a). Collaboration Engineering can only be successful if collaboration engineers can design high quality collaboration processes that are transferable to practitioners.

The Collaboration Engineering approach has the intention to overcome some critical challenges in the organizational implementation of collaboration support, and therewith enable a more sustained approach to collaboration support. The Collaboration Engineering approach has two key challenges:

- The design of an efficacious, acceptable, reusable, transferable and predictable collaboration process prescription.
- The transfer of the collaboration process design to practitioners in organizations.

This research offers theoretical foundations to support the design and transfer of collaboration process designs. First we defined 'quality of collaboration' and determined the dimensions of a collaboration process design. Quality of collaboration can be described by the satisfaction with respect to process and outcome, the commitment of participants to the goal of the collaborative effort, and the effectiveness, efficiency and productivity of the collaborative effort. Based on these constructs we presented a theory to explain the relation between goal achievement and the commitment of resources by the participants in the collaboration process.

Quality of collaboration process design has five dimensions that pose some tradeoffs. The quality of the collaboration process design depends on its efficaciousness with respect to the goal, its acceptance by the participants and stakeholders in the process, its reusability in different instances of the collaborative task, its transferability to practitioners and the predictability of each of these 'fits'.

Next, we offer support to design and transfer a collaboration process design, based on existing theories and interviews/surveys among facilitators and practitioners in a

variety of organizations. To create transfer support we used a theory that describes the cognitive load of a learning task and techniques to reduce the cognitive load of this task (Sweller 1988). To support the design of a collaboration process prescription we used different design and engineering approaches in literature (Davenport, 1993). Finally we present a further conceptualization of the thinkLet concept, to support the design and transfer of facilitation techniques.

The thinkLet concept was first proposed in 2001 by Briggs and de Vreede (Briggs and Vreede, 2001, Briggs et al., 2001). In this research we further developed the thinkLet concept. ThinkLets are design patterns; carefully described interventions in group processes with a predictable pattern of collaboration and predictable characteristics of its result. The thinkLet concept offers a conceptual framework for the documentation of facilitation interventions and can be instantiated in two ways. First it can be instantiated to serve the collaboration engineer in choosing among thinkLets, and offers information to make deliberate choices with respect to each of the 'quality of collaboration process design' dimensions (design support). Second, it can be instantiated to support the transfer of the collaboration process prescription (transfer support).

The contribution of our research is a design and transfer approach, with support for both from the thinkLet concept, design support (models and classifications) and a collaboration process prescription template. Each of the support concepts has been developed in several iterations, where each iteration was evaluated to further improve the concept.

The design and transfer support were finally used to design two collaboration processes for government organizations, and to train practitioners in those organizations. The practitioners received a two-day training and then executed the collaboration process prescription for a group. We evaluated the quality of collaboration from a participant perspective and found encouraging results. The quality of collaboration reported in the sessions supported by practitioners was not significantly different than the quality of collaboration of the same session supported by professional facilitators. Limitations with respect to the number of practitioners and facilitators and the difficulties in controlling the experiment make this an encouraging, but tentative conclusion.

Samenvatting

Samenwerking is vereist voor het creëren van kennis en voor innovatie. Daarom is samenwerking van belang voor de concurrentie-positie van organisaties in de kenniseconomie (Drucker 1969; Grossman and Helpman 1991; Mintzberg 1983; Nonaka 1994; Qureshi and Keen 2004). We definiëren samenwerking als 'gezamenlijke inspanning om een doel te bereiken' (Harper 2001). Samenwerken is ingewikkeld, en groepen zijn vaak niet in staat om zelf de valkuilen van samenwerking te vermijden (Ellis et al., 1991, Nunamaker et al., 1997, Weiss and Hughes, 2005, DeSanctis and Gallupe, 1987, McGrath, 1991, Schwarz, 1994). Om de kwaliteit van samenwerking te verbeteren is ondersteuning nodig (Dennis et al. 2001; Nunamaker et al. 1997; Schwarz 1994).

Ondersteuning voor samenwerking kan geboden worden in de vorm van technologie en procesondersteuning zoals facilitatie. Het is voor organisaties moeilijk om op duurzame wijze ondersteuning te bieden voor veel voorkomende samenwerkingstaken (Agres et al. 2005; Briggs et al. 2003a). Collaboration Engineering is een aanpak om op duurzame wijze ondersteuning te bieden door het ontwerpen van samenwerkingsmethoden voor veel voorkomende taken die samenwerking vereisen en een belangrijke toegevoegde waarde bieden aan de organisatie, en door deze procesontwerpen te implementeren door middel van het opleiden van practitioners met de bedoeling dat zij op basis van het procesontwerp, zelfstandig, zonder inmenging van professionals, groepen kunnen ondersteunen in deze taak (Briggs et al. 2006b). In Collaboration Engineering wordt de traditionele facilitatierol opgesplitst in een ontwerpgedeelte en een uitvoerend gedeelte, waardoor deze aanpak eenvoudiger op duurzame wijze in een organisatie geïmplementeerd kan worden (Briggs et al. 2003a). Collaboration Engineering als aanpak vereist dat collaboration engineers samenwerkingsprocessen kunnen ontwerpen die van hoogwaardige kwaliteit zijn en die bovendien overdraagbaar zijn aan practitioners.

De Collaboration Engineering aanpak heeft de intentie om kritische obstakels in de organisatorische implementatie van de ondersteuning van samenwerking te overbruggen, om zodoende een meer duurzame vorm van samenwerkingsondersteuning mogelijk te maken. De Collaboration Engineering aanpak kent twee belangrijke uitdagingen:

- Het ontwerp van een geschikt, acceptabel, herbruikbaar, overdraagbaar en voorspelbaar procesontwerp
- Het overdragen van een proces ontwerp voor samenwerking aan practitioners in organisaties.

Dit onderzoek biedt een theoretische onderbouwing voor het ondersteunen van het ontwerpen en de overdracht van procesontwerpen voor samenwerking. Eerst hebben we het begrip 'kwaliteit van samenwerking' nader gedefinieerd. Vervolgens hebben we de dimensies van 'kwaliteit van een procesontwerp voor samenwerking' geïdentificeerd. Kwaliteit van samenwerken kan beschreven worden aan de hand van de volgende factoren; tevredenheid met betrekking tot het proces en de uitkomst, de mate waarin deelnemers bereid zijn zich te commiteren aan het doel van de samenwerking en de effectiviteit, efficiency en productiviteit van de samenwerking. Op basis van deze factoren presenteren we een theorie over de relatie tussen het bereiken van een doel en het commiteren van middelen door de deelnemers aan het samenwerkingsproces.

Kwaliteit van het procesontwerp voor samenwerking heeft vijf dimensies, die niet individueel geoptimaliseerd kunnen worden maar in balans gebracht moeten worden. De kwaliteit van het proces ontwerp van samenwerking hangt af van in hoeverre het proces ontwerp geschikt is voor het doel van het proces, de acceptatie van het proces door deelnemers en belanghebbenden, de herbruikbaarheid van het procesontwerp in verschillende situaties, de overdraagbaarheid van het proces aan practitioners en de voorspelbaarheid van ieder van deze dimensies.

Vervolgens bieden we ondersteuning bij het ontwerpen en overdragen van het procesontwerp voor samenwerking, op basis van bestaande theorieën en interviews/ enquêtes gehouden onder facilitatoren en practitioners in verschillende organisaties. Ter ondersteuning van de overdracht hebben we gebruik gemaakt van een theorie die de cognitieve belasting van de leertaak beschrijft en technieken aanbiedt om deze cognitieve last te reduceren (Sweller 1988). Ter ondersteuning van het ontwerpen van samenwerkingsprocessen hebben we gebruik gemaakt van verschillende aanpakken voor ontwerpen uit de literatuur (Davenport 1993). Verder presenteren we een uitbreiding op de conceptualisatie van het thinkLet concept, om het ontwerpen en overdragen van facilitatie technieken verder te ondersteunen.

Het thinkLet concept is geintroduceerd in 2001 door Briggs en de Vreede (Briggs and Vreede 2001; Briggs et al. 2001). In dit onderzoek hebben we het thinkLet concept verder ontwikkeld. ThinkLets zijn ontwerppatronen; zorgvuldig voorgeschreven interventies in groepsprocessen met een voorspelbaar samenwerkingspatroon als effect en met resultaten die specifieke karakteristieken vertonen als effect. Het thinkLet concept biedt een conceptueel raamwerk voor de documentatie van faciltiatie-interventies en kan op twee verschillende manieren gebruikt worden. Ten eerste kan het gebruikt worden om de collaboration engineer te ondersteunen in het kiezen tussen thinkLets en biedt het informatie om overwegingen met betrekking tot ieder van de dimensies van ‘kwaliteit van het procesontwerp voor samenwerking’ te maken (ontwerp ondersteuning). Ten tweede kan het gebruikt worden om de overdracht van de documentatie van het procesontwerp voor samenwerking te ondersteunen (overdracht ondersteuning).

De bijdrage van dit onderzoek is een ontwerp- en overdrachtsaanpak met ondersteuning voor beide door middel van het thinkLet concept, ontwerpondersteuning (modellen en classificaties) en een blauwdruk voor de documentatie van het procesontwerp voor samenwerking. Ieder van deze ondersteunende concepten is ontwikkeld in enkele cycli waarbij iedere cyclus geëvalueerd is om verbeteringsmogelijkheden te identificeren.

De ondersteuning voor het ontwerpen en overdragen zijn uiteindelijk gebruikt om twee samenwerkingsprocessen te ontwerpen voor overheidsorganisaties, en om practitioners op te leiden voor die organisaties. De practitioners hebben een tweedaagse opleiding gehad en hebben vervolgens het proces met een groep uitgevoerd op basis van de documentatie van het procesontwerp. We hebben de kwaliteit van samenwerking geëvalueerd vanuit het perspectief van de deelnemers en hebben bemoedigende resultaten gevonden. De kwaliteit van samenwerking gerapporteerd door de deelnemers van groepsprocessen die door practitioners begeleid waren, was niet significant verschillend van de kwaliteit van samenwerking gerapporteerd door de deelnemers van groepsprocessen die door professionele facilitatoren begeleid waren. Beperkingen met betrekking tot het aantal practitioners en facilitators en enkele problemen met betrekking tot het controleren van de experimenten maakt dit tot een bemoedigende maar voorlopige conclusie.

Curriculum Vitae

Gwendolyn Laetitia Kolfschoten was born in Amsterdam on October 7th, 1980. After Graduating from 'Haags Montessori Lyceum' in The Hague in 1998 she studied, Systems Engineering Policy Analysis and Management at Delft University of Technology and specialized in GSS and thinkLet research. She did an internship at EADS where she deployed a GSS and worked as a chauffeur for the GSS at the Faculty of Technology, Policy and Management. As a part-time job she thought mathematics at a Secondary School in Delft. Her master project resulted in a first prototype of a tool to support the choice between thinkLets. After graduating in February 2004 she became a research and teaching assistant at the department of Systems Engineering at the Faculty of Technology, Policy and Management. During her research she presented her work at several international conferences, workshops and consortia. Also, she co-organized several mini tracks, workshops and sessions at international conferences, and reviewed for several conferences and international journals. Furthermore, she facilitated a number of GSS supported sessions for a variety of organizations. Last she served as a course manager and teacher for both bachelor and master courses at the faculty, and supervised several students in the master thesis project.

Publications

Refereed journal publications, in English:

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), pp. 140-154.

Kolfschoten, G. L., Briggs, R. O., Vreede, G. J., de, Jacobs, P. H. M. and Appelman, J. H. (2006) "Conceptual Foundation of the ThinkLet Concept for Collaboration Engineering", *International Journal of Human Computer Science* 64(7), pp. 611-621.

Kolfschoten G.L., Hengst M. den, Vreede G.J. de, (2007) "Issues in the Design of Facilitated Collaboration Processes", *Group Decision and Negotiation* 16(4), pp. 347-361.

Refereed conference papers, in English:

Vreede, G.J. De, Vogel, D.R., Kolfschoten, G.L., Wien, J.S., (2003) "*Fifteen years of in-situ GSS use: a comparison across time and national boundaries*." Proceedings of the Hawaii Internal Conference on System Science. Los Alamitos: IEEE Computer Society Press.

Kolfschoten G.L., Appelman J.H., Briggs, R.O., Vreede G.J., de, (2004) "*Recurring Patterns of Facilitation Interventions in GSS Sessions"*, Proceedings of the Hawaiian Internal Conference on System Sciences, Waikoloa: IEEE Computer Society Press.

Kolfschoten G.L., Briggs, R.O., Appelman J.H., Vreede G.J., de, (2004) *"ThinkLets as Building Blocks for Collaboration Processes: A Further Conceptualization"*, Proceedings of CRIWG, Berlin, LNCS, Springer-Verlag.

Kolfschoten G.L., Veen W., (2005) *"Tool Support for GSS Session Design"* Proceedings of the Hawaii Internal Conference on System Sciences, Waikoloa: IEEE Computer Society Press.

Kolfschoten G.L., Hengst M. den, Vreede G.J. de, (2005) "*Issues in the Design of Facilitated Collaboration Processes"*, Proceedings of the International Conference on Group Decision and Negotiation, Vienna, Austria.

Briggs R.O., Kolfschoten G.L., Vreede G.J. de, (2005) *"Toward a Theoretical Model of Consensus Building"*, Proceedings of Americas Conference on Information Systems, Omaha, NE. AIS.

Santanen, E. L., Kolfschoten G.L., Golla K., (2006) *"The Collaboration Engineering Maturity Model"* Proceedings of the Hawaii Internal Conference on System Sciences, Kauai: IEEE Computer Society Press.

Hengst M. den, Kolfschoten G.L., Dean D.L., Chakrapani A., (2006) *"Assessing the Quality of Collaborative Processes"*, Proceedings of the Hawaii Internal Conference on System Sciences, Kauai: IEEE Computer Society Press.

Kolfschoten G.L., Rouwette, E.A., (2006) *"Choice Criteria for Facilitation Techniques"*, Proceedings of the First HICSS Symposium on Case and Field Studies of Collaboration, Kauai.

Kolfschoten, G.L., G.J. de Vreede, A. Chakrapani, & P. Koneri. (2006): *"A design approach for collaboration engineering"*. Proceedings of the First HICSS Symposium on Case and Field Studies of Collaboration, Kauai.

Macaulay, L.A., A. Alabdulkarim, & G.L. Kolfschoten. (2006): *"An analysis of the role of the facilitator and alternative scenarios for collaboration support."* Proceedings of the First HICSS Symposium on Case and Field Studies of Collaboration, Kauai.

Briggs, R.O., G.L. Kolfschoten, & G.J. Vreede, de. (2006): *"Instrumentality theory of consensus.* Proceedings of the First HICSS Symposium on Case and Field Studies of Collaboration, Kauai.

Kolfschoten, G.L., Briggs, R.O., and Vreede, G.J. de. (2006). *"Definitions in Collaboration Engineering".* Proceedings of the First HICSS Symposium on Case and Field Studies of Collaboration. Kauai.

Kolfschoten, G.L., F. Niederman, de Vreede G.J., & R.O. Briggs. (2006): *"Understanding the job requirements and roles for group support systems facilitators".* Proceedings of the SIGMIS Computer Personnel Research Conference.

Briggs, R.O., G.L. Kolfschoten, G.J. de Vreede, & D.L. Dean. (2006): *"Defining key concepts for collaboration engineering."* Proceedings of the Americas Conference on Information Systems, Acapulco, Mexico. AIS.

Kolfschoten, G.L., E. Valentin, de Vreede G.J., & A. Verbraeck. (2006): *"Cognitive load reduction through the use of building blocks in the design of decision support systems,"* Proceedings of the Americas Conference on Information Systems, Acapulco, Mexico. AIS.

Kolfschoten, G.L., and Rouwette, E. (2006). "Choice Criteria for Facilitation Techniques: A Preliminary Classification". Proceedings of the International Conference on Group Decision and Negotiation. Karlsruhe, Germany. Universtatsverlag Karlsruhe.

Kolfschoten, G.L., Pietron, L., and Vreede, G.J., de. (2006). "A training approach for the transition of repeatable collaboration processes to practitioners". Proceedings of the International Conference on Group Decision and Negotiation. Karlsruhe, Germany. Universtatsverlag Karlsruhe

Kolfschoten, G.L., and Hulst, S. van der. (2006). "Collaboration Process Design Transition to Practitioners: Requirements from a Cognitive Load Perspective". Proceedings of the International Conference on Group Decision and Negotiation. Karlsruhe, Germany. Universtatsverlag Karlsruhe.

Kolfschoten, G.L., and Santanen, E.L. (2007). "Reconceptualizing Generate ThinkLets: the Role of the Modifier". Proceedings of the Hawaii International Conference on System Science, Waikoloa, IEEE Computer Society Press.

Kolfschoten, G.L. (2007). "A Toolbox for the Facilitation of E-Democracy". in: Proceedings of the Group Decision and Negotiation Conference. Mt. Tremblant, Canada, Concordia University.

Kolfschoten, G.L., Vreede, G.J. de, Briggs, R.O., Sol, H.G. (2007). "Collaboration Engineerability". in: Proceedings of the International Conference on Group Decision and Negotiation, Mt. Tremblant, Canada, Concordia University.

Kolfschoten, G.L., and Houten, S.P.A. van. (2007). "Predictable Patterns in Group Settings through the use of Rule Based Facilitation Interventions". in: Proceedings of the International Conference on Group Decision and Negotiation, Mt. Tremblant, Canada, Concordia University.

Kolfschoten, G.L., Briggs, R.O., and Vreede, G.J. de. (2007). "Computer Aided Pattern-Based Collaboration Process Design: A Computer Aided Collaboration Engineering Tool". Accepted for: Proceedings of the CRIWG, Bariloche, Argentina, LNCS Springer-Verlag.

Kolfschoten, G.L., and Vreede, G.J. de. (2007). "The Collaboration Engineering Approach for Designing Collaboration Processes". Accepted for: Proceedings of the CRIWG, Bariloche, Argentina, LNCS Springer-Verlag.

Kolfschoten, G.L., Niederman, F., Vreede, G.J. de and Briggs, R.O. (2008) "Roles in Collaboration Support and the Effect on Sustained Collaboration Support". Accepted for Proceedings of the Hawaii International Conference on System Science, Waikoloa, IEEE Computer Society Press.

Book Chapters, in English:

Briggs, R.O., G.J. Vreede, de, & G.L. Kolfschoten. (in press). ThinkLets for e-Collaboration. In N. Kock (Ed.), *Encyclopedia of e-collaboration*: Idea Group Inc. *in press*

Kolfschoten, G.L., R.O. Briggs, & G.J. de Vreede. (in press). Patterns of Facilitation Techniques and Interventions in e-Collaboration. In N. Kock (Ed.), *Encyclopedia of e-collaboration*: Idea Group Inc. *in press*

Vreede G.J., de, R.O. Briggs, & G.L. Kolfschoten. (in press). Collaboration Engineering. In N. Kock (Ed.), *Encyclopedia of e-collaboration*: Idea Group Inc. *in press.*