CoPrA: A Process Analysis Technique to Investigate Collaboration in Groups

Isabella Seeber
University of Innsbruck
isabella.seeber@uibk.ac.at

Ronald Maier
University of Innsbruck
ronald.maier@uibk.ac.at

Barbara Weber
University of Innsbruck
barbara.weber@uibk.ac.at

Abstract

Studying collaboration and its support by IT in experiments has been limited to collecting and analyzing collaboration products and performance criteria ex post. This paper presents CoPrA, a COllaboration PRocess Analysis technique, which allows investigating behavior of individuals in distributed groups mediated by collaboration technology from a process-oriented perspective. Based on coded communication logs, CoPrA uses information on user participation and actions performed by group members to extract similarities or differences within and between groups. The analysis' results answer questions such as whether groups adopt treatments in experimental settings, whether and when patterns of collaboration change over time, and how comprehensively participants negotiate during collaboration. Thus, CoPrA can enhance our understanding of collaboration processes and additionally provides grounds for richer analysis and increased internal validity of experimental data. Applicability of the technique is demonstrated with a laboratory experiment investigating the impact of process structure on patterns of collaboration.

1. Introduction

In knowledge-intensive organizations, an increasing share of semi- or unstructured knowledge work [12,16,25,31] demands highly skilled, creative and self-organized employees and a high share of collaboration across time, space and the organization's institutional boundaries. Thus, teams are often challenged with the appropriate use of technology or how to coordinate, communicate, and cooperate effectively. The discipline of information systems (IS) has a long tradition in studying phenomena of group decision support systems [9,23], computer supported cooperative work (CSCW) [11] or, more recently, collaboration engineering (CE) [28] which aim at understanding the practices of collaboration and designing organizational and IT-based measures or infrastructures for their support in order to increase productivity of knowledge work [10].

CE can be understood as an approach that combines design, facilitation and training expertise with the goal to create repeatable work practices executed by practitioners and supported by collaboration technology (CT) [17]. ThinkLets represent the core concept used to capture de- and prescriptions of work practices of successful collaboration, made available for reuse and can be regarded as collaboration facilitation [18,28]. ThinkLets strive to support practitioners in creating suitable, transferable, predictable and repeatable collaboration processes [17,18] and hence can be regarded as building blocks for performing collaboration tasks.

Moreover, investigating peoples' behavior during the execution of collaboration processes and supported by CT, is deemed important to inform system design which requires enhanced understanding of how people actually behave and use technology in groups [3]. A number of performance criteria have been proposed for evaluating thinkLets and were (partly) applied in experimental treatments [8,13,15,21]. They include result-oriented criteria, e.g., speed, level of reduction, level of shared understanding, and process-oriented performance criteria, e.g., acceptance by participants, satisfaction with thinkLet by participant, ease of use for facilitator [7].

We argue that in experimental settings it is beneficial to include the analysis of actions performed during the collaboration process in addition to output-oriented data and performance criteria that are collected ex post. This allows us to augment the understanding of how experimental conditions, e.g., collaboration process facilitation, actually affect behavior. By analyzing the actions performed by the different group members while being engaged in the collaboration process, we obtain insights into the collaboration process itself and can analyze how the result or the end product is developed, e.g., a written report. Beyond that, richer analysis, i.e., method triangulation, and a check of internal validity of experimental data are fostered, i.e., whether thinkLets are adopted or not. Goal of this paper thus is to propose CoPrA, an analysis technique using process mining which allows exploiting transcribed and coded communication logs of experimental treatment and control groups. This aims at extending our understanding of peoples' behavior during collaboration. We demonstrate CoPrA's applicability with a case example of an experiment using thinkLets and extended process structure for a collaborative writing task.

The remainder of this paper is structured as follows: Section 2 elaborates on the concept of thinkLets for designing repeatable collaboration processes and criteria for assessing the processes' performance. Section 3 introduces the proposed analysis technique from a conceptual perspective by outlining three phases, as well as possible visualization of results. Subsequently, Section 4 exemplifies its applicability by presenting a laboratory experiment and additional information that was gathered in our adoption of this technique. Section 5 discusses our results and reflects on limitations. Section 6 concludes this paper and gives an outlook on future research.

2. Background

Collaboration describes a joint effort to achieve a group goal and appears to be correlated with an organization's success and sustainability [28]. When people reside in some kind of collaborative setting, expenses for planning, researching, training, and other development activities can be reduced, resulting in cost reduction and efficiency as well as increased quality [22].

In practice, teams are often challenged with, e.g., how to use CT effectively or how to coordinate, communicate, and cooperate productively when working together. Professional facilitators would possess the necessary skills and competencies, e.g., concerning analytics or communication, to guide through or intervene into the collaboration process, but represent a costly option for organizations [28]. In contrast to collaboration supported by facilitators, thinkLets enclose "best facilitation practices" that can be used by collaboration engineers to design a repeatable and transferable collaboration process which in turn is applied and executed by practitioners, the domain experts [18]. In everyday practice, collaboration processes are characterized by recurring activities which are denominated as collaboration patterns in the domain of CE and comprise [4]:

- *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,
- *organize* - move from less to more understanding of the relationships among concepts the group considers,
- *evaluate* - move from less to more understanding of the relative value of the concepts under consideration,
- *build consensus* - move from fewer to more group members who are willing to commit to a proposal.

For each of these patterns there exists a variety of thinkLets (see [5,6,8] for more details and examples). Each thinkLet is documented in a specific way, including identification (describing the name of the thinkLet) script, (describing minimum instructions how to walk through a process), and a selection guideline (describing when to best use the thinkLet) [28].

For measuring the quality of thinkLets on collaboration processes, e.g., Hengst et al. [13] or Kolfschoten [19] collected various performance criteria for repeatable collaboration processes comprising process effectiveness, process efficiency, quality of results, etc. Additionally, empirical studies such as Kamal et al. [15] and de Vreede et al. [27] analyzed the quality of specific collaboration processes that were facilitated with thinkLets by adopting these criteria. In general, there exists a need to investigate group processes and what people actually do during collaboration by analyzing the actions they execute. This goes along with a proposed research agenda for CE, which demands to do "further fundamental research on how groups and organizations accept, adopt, and adapt repeatable collaboration processes. How do groups embrace a given repeatable collaboration process over time? How do they change it over time to better suit their needs?" [28]. When evaluating the effectiveness of the thinkLets, the focus so far was mainly put on the collaboration product or peoples' perceptions of the collaboration process. This paper extends this collection of approaches for evaluating thinkLets by providing a technique that assesses the collaboration process, among other things whether the provided process structure was actually adopted and hence impacted behavior.

Consequently, the following section takes up on this research agenda and proposes a technique which allows investigating collaboration processes over time and is suitable for understanding thinkLet use and for studying implications for design.

3. CoPrA: A Technique for Analyzing Collaboration Processes

Even though existing studies generated interesting insights on the usefulness and purposefulness of process structure (e.g., [21]) in general, or thinkLets (e.g., [8]) in particular, research approaches mostly focus on investigating the end product of collaboration, but not the collaboration process itself. Next to analyzing the product of collaboration, e.g., a document, the collection of questionnaire data relating to, e.g., satisfaction, quality, or effectiveness presents only a snapshot retrospective of the collaboration process. We recommend analyzing factual information on the process to complement the analysis of the product and perceptions put forward by participants.

We argue that CoPrA allows investigating collaboration processes by closely looking at the actions team members performed during the collaboration process which enriches analysis results. Moreover, internal validity of, e.g., laboratory experiments, can be strengthened, since adopting given repeatable collaboration processes, such as fostered by thinkLets, should lead to observable changes of process behavior. For this purpose, process mining has been proposed as a means to monitor and analyze the behavior of people during process execution within the CSCW domain [1]. However, collaborative settings are often characterized by ill-structured problems [14]. This entails challenges for existing process mining algorithms and techniques [29]. With CoPrA (cf. Figure 1), we can extract information and reflect on the context during the collaboration process, as well as state similarities or differences due to occurrences within the logged information, the eventlog. Hence, with the application of the two lenses, distribution and control-flow perspective, the focus of the research problem not only strives to explain the what of collaboration behavior, but also the how [26].

As Figure 1 depicts, the technique is divided into the three phases that are common to approaches for analyzing and interpreting process data: data collection, data preparation, and process analysis. It represents a process model that strives to enhance understanding of collaboration processes by analyzing action and user information such as in the underlying case where the designed collaboration process is facilitated by thinkLets and moderated by the group itself rather than by a facilitator. In the following, we explain these phases before a practical example is given in Section 4.

3.1 Data Collection

CoPrA's first phase deals with gathering data necessary for all further analysis. After research design and research questions have been defined, the researcher needs to consider how data can be gathered, at best automatically. There are a number of data sources, such as exchanged comments that can be automatically exported from the CT tool, but also audio or video logs can be used.

3.2 Data Preparation

In CoPrA's second phase, the method of content analysis is performed to code data with a deductively or inductively created coding schema. The selected coding schema is what drives the results of the analysis and hence all subsequent interpretations about thinkLets and their impact on collaboration processes. ThinkLet scripts are often characterized by negotiation prompts, such as discussing a proposal, finding an agreement, reviewing comments. Hence, the collaboration process will likely be filled with actions such as propose, argue, argue against an idea. Depending on the thinkLet(s) used, peaks of specific actions are anticipated and can be made observable within the eventlog, i.e. communication protocols. In literature a variety of negotiation models exist [2,20,24] that can be used deductively as coding schemas but need to be cautiously selected by researchers depending on the underlying research question.

Since the understanding of the applied coding schema will be of importance for the case example in Section 4, it is shortly outlined in Table 1. For coding communication logs, the framework of generic activities on the pragmatic level for negotiation after Rittgen [24] was selected and action description as well as examples adapted to fit the collaborative activity. In general three generic action types can be differentiated. Action type UL relates to the understanding of text in the case description and describes actions referring to questions and answers. Coordination activities and therefore the structuring of the overall process are coded with setting agenda (SA) actions, which also consider whether and when adaptations to the given timeline are performed. Finally, negotiation actions (N) describe conversation and discussions among participants based on proposals or ideas. Thus, negotiation actions describe whether group members agreed or disagreed to ideas [24].

The collected communication logs were coded manually by the authors identifying negotiation actions according to the coding schema after Rittgen (2007) [24]. Subsequently, a so-called eventlog, complying with the MXML schema for ProM, was created for further analysis.

At its core, the element <AuditTrailEntry> refers to a performed action/event and comprises the following information.

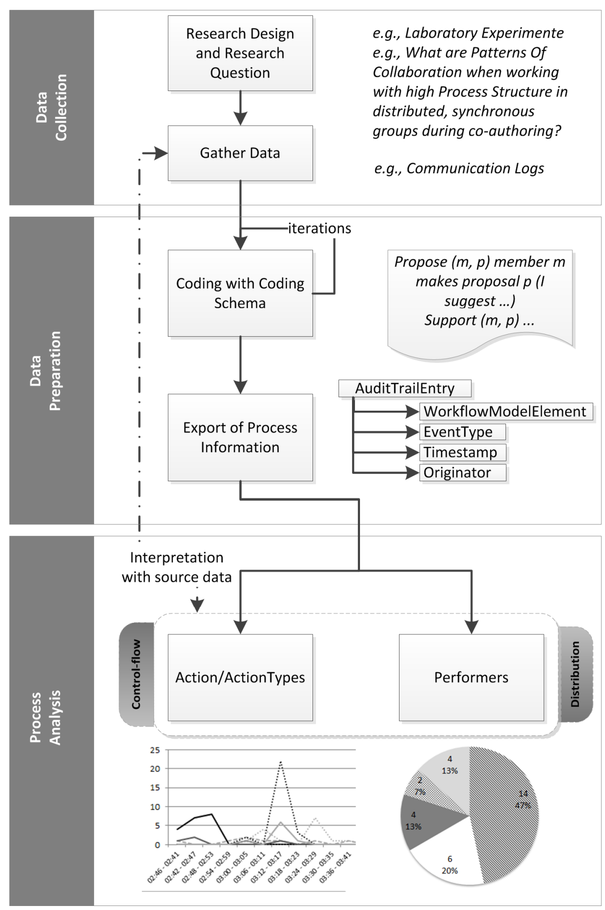

Figure 1: Analysis technique for collaboration processes

Please note that the element EventType, an element required in the MXML standard format, is not relevant for our purpose of analysis:

- *WorkflowModelElement*. This element describes the performed action (e.g., propose)

- *Timestamp*. This element describes the point in time when the action started (e.g., 2011-05-11T03:02:00)
- *Originator*. This element denotes the performer of a statement (e.g., Quinn).

The interpretation of content and meaning of codes is crucial for all further analysis. Each coder needs to be well aware of the rules or semantic meaning a code has. For example, the codes “argue-for” and “support” are semantically closely related. However “argue-for” was only assigned to a text fragment in case that a group member provided an argument for an idea and not only a simple agreement. In most cases these rules develop over time and should be documented. Hence, deep and reflective collaboration among researchers as well as a number of iterations through the communication logs are necessary to define the appropriate level of granularity and to ensure coherent coding.

Table 1: Coding schema for negotiation after Rittgen (2007) [24]

	ACTION	CODING	EXAMPLE
collaborative negotiation (N)	Member m expresses acceptance to proposal p	**accept (m, p)**	“A wiki is a good idea to distribute information and knowledge.”
	Member m delivers argument a to challenge p.	**argue_against (m, p, a)**	“Internet is not enough.”
	Member m delivers argument a to support p.	**argue_for (m, p, a)**	“In order to give new employees a guideline”.
	Member m expresses objection to proposal p.	**challenge (m, p)**	“But how do we get the people to use it?”
	Member m proposes p’ instead of p.	**counter (m, p, p’)**	“I think extra payment does not work but we have to build a kind of teamspirit.”
	Member m makes proposal p.	**propose (m, p)**	“some kind of groupware …”
	Member m expresses rejection to proposal p	**reject (m, p)***	"I think that is no solution."
	Member m expresses consent to proposal p.	**support (m, p)**	“I think groupware is a good idea.”
	Member m withdraws his/her proposal p.	**withdraw (m, p)***	“My idea does not work; let us forget it.”
understanding text (UL)	Member m needs clarification on issue q.	**ask (m, q)**	“”How should we go on?”
	Member m provides a possible answer to question q	**assume (m, q, a)**	“I think we are supposed to give a solution to this.”
	Member m gives a definite answer a to question q	**clarify (m, q, a)**	“We should discuss all together.”
setting agenda (SA)	Add activity a to the agenda as item number n.	**add (a, n)**	“Collect more ideas concerning how to “manage, preserve and distribute knowledge.”
	Perform next activity a from the item number n.	**perform (a, n)**	“Let’s write the summary together.”

* actions marked with (*) did not occur in the underlying experiment and hence were originally taken from the author [24]

3.3 Process Analysis

Up to this point, coding was performed by researchers that are preferably also part of the experimental sessions so that interpretation of text is eased. For the third phase, the coded content is used for analysis from two major perspectives: control-flow and distribution perspective. Likewise these two perspectives can be related to either the performed actions or action types and performers. The control-flow perspective allows interpreting similarities or differences of actions or performers due to changes over time. Consequently, the differences and similarities of a group’s behavior can be observed and studied. A further scenario for the usefulness of investigating the control-flow perspective relates to the changes of actions performed by participants over time. Hence, information in respect to, e.g., social loafing, or domination can be deduced. With the availability of transcribed communication among participants, these mined observations can then be related back to the context they occurred in. Moreover, ProM’s process mining plugin (Flexible)HeuristicMiner, (F)HM, automatically generates likely sequences of actions which can extend our understanding how actions relate to each other [29,30]. A detailed description of this mining technique would go beyond the scope of this paper and therefore an example in the following section will illustrate possible benefits.

The distribution perspective summarizes information about actions and performers. This allows extracting the number of occurrences of actions in respect to the overall collaboration process or selected collaboration patterns (which have been made visible due to the control-flow perspective). Again, distributions based on performers or actions/action types can be analyzed.

With the exception of the generation of the action sequence diagram of the process mining algorithm (F)HM in ProM, all other computations relating to action, action types, and participants can be generated with spreadsheet applications.

4. Case Example and Discussion

In order to illustrate CoPrA's usefulness and purposefulness, we conducted a laboratory experiment investigating the impact of process structure on patterns of collaboration. The purpose here is to demonstrate the technique rather than present results from empirical work with thinkLets.

4.1 Experimental Setup and Design

Our laboratory experiment was largely based on an experiment described in Lowry et al. [21]. The focus of the study was to investigate how groups differ during a synchronously distributed collaborative writing activity when impacted by process structure. The research design for this experiment was a one-way between-groups independent design and tested the factor process structure on two levels, high and low process structure (K=2). Groups were tasked to collaboratively write a report presenting ideas for improvement based on a mini case study in the domain of knowledge management.

Similar to Lowry et al. [21], our quasi-experiment investigated process structure as an experimental condition, which was expressed by the level of explicitness of the description of how to execute the task. The scripts included explicit steps how to walk through the phases of brainstorming, outlining and drafting, as well as group reviewing and finalizing for a collaborative writing task. The authors selected and adapted thinkLets as well as additional prompts for writing the report prior to the experimental session so that participants could moderate the overall process by themselves. The following thinkLets for the respective collaboration patterns were used: generate – DirectedBrainstorm, reduce/clarify/evaluate – FastFocus, BoomWagon, and build consensus – ReviewReflect [8]. At the beginning of the session, a researcher introduced the collaborative writing task and explained that each script includes prompts how to walk through the overall process. Groups were not informed that these prompts differed in explicitness and hence were not trained in the use of specific thinkLets. Instructions prior to the experiment were merely limited to the use of the CT, Google Wave. 24 students participated in the experiment and were randomly assigned into groups of three, resulting in four groups working with high and four groups working with low process structure. Each member was located into one of three rooms equipped with the necessary tool support that allowed simultaneous distributed group work.

Differences between our laboratory experiment and the one described in [21] exist mainly in respect to duration and tool support. Our experiment lasted for three hours and was therefore rather tight on available time for introducing the experiment and tool, reading the task, brainstorming for ideas, outlining, drafting and reviewing the report. In contrast to [21], our experiment used a freely available collaboration tool that supported synchronous communication and writing (Google Wave). Specific functionality allowed automatic export of log data, comprising author, timestamp, and content which was a direct input for coding. Communication logs from three high process structured groups are used to exemplify the purpose of the technique and were coded along the coding schema introduced in Table 1 supported by ATLAS.ti. As mentioned before, the phase of coding communication data builds the basis for further analysis and needs to be well understood by all researchers. Depending on the specific coding schema that is used, researchers need to define rules for coding. For example, "argue_for" and "support" express some kind of agreement to a proposal. However, "argue-for" was only applied to the transcript in case that a group member delivered some kind of argument. If no argument was delivered, the text was coded with "support". Likewise, the assignment of "accept" was dependent on the idea and could only occur if no more comments were related to this idea. Therefore, the tracking of each idea is necessary. Subsequently, the tool's XML export allowed manipulating data in such a way that it corresponded to the defined data structure outlined in Section 3.2.

With the application of CoPrA, we strived to analyze similarities and differences in collaboration processes based on information related to performed action and participants. Specifically, the analysis technique should answer the following questions:

- Can we observe if and when participation of group members changes over time?
- Can we observe if and when phases or patterns of collaboration change over time?
- Can we observe if participants adopt the provided experimental treatment, here the given process structure?
- Can we observe how deeply and comprehensively participants negotiate?
- Can we observe sequences of actions that can be interpreted as candidates for causal dependencies?

4.2 Data Analysis and Discussion of Results

As pointed out in Figure 1 two perspectives lead through the analysis phase: distribution and control-flow perspective.

We start out with the distribution perspective of action types in Figure 2. Action types denominate actions that are grouped into negotiation (N), setting agenda (SA), or understanding task (UL). In all groups the action types N and UL were highly performed. Especially, group II showed with 63 negotiation actions of 94 overall performed actions a high share of negotiation. Hence, this gives a first indication of how deeply groups discussed their ideas, i.e., number of N actions, and whether they required clarification, i.e., number of UL actions, to achieve their goal. When focusing on action type N, all three groups showed consensus-oriented behavior. This means that the action "support" was often performed to agree to a proposal while no argument was given (this would have been coded with “argue-for”). In particular, group I showed 10 “support” actions (17%), group II had 28 (45%) (cf. Figure 3) and group III had 4 “support” actions (16%).

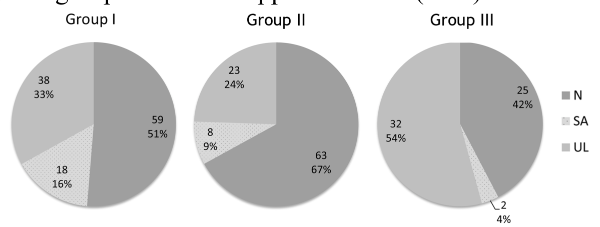

Figure 2: Distribution after action types

This observation can be supported by the fact that none of the actions “withdraw” and “reject” occurred. Moreover, the action “challenge” was not often performed. In particular, group I performed 2 (7%) “challenge” actions, group II had 1 (2%), and group III had 4 (16%). The same low usage holds for “argue-against” (group I had 1(1%), group II had 0, and group III had 2 (8%)).

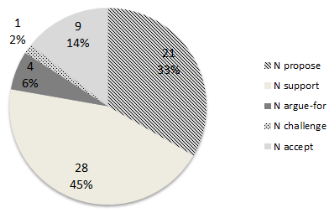

Figure 3: Group II - distribution of action types "Negotiation"

A further observation from the distribution statistics is that not all proposed ideas were accepted or rejected. Based on the script, however, groups were asked to discuss ideas and select the ones that seem to be most useful and should be covered in the report. The ratio between actions “propose” and “accept” (“reject” was not performed) add up to 54% (14/26) in group I, 32% (9/28) in group II, and 18% (2/11) in group III. This implies that proposition were not explicitly rejected or accepted by group members, but rather implicitly (not) taken up during the drafting phase of the writing activity. An extended discussion of this observation will be made when focusing on the control-flow perspective, later in this section. When focusing on action type SA in Figure 2, it is observable that this action type takes up the least share. Here groups agreed on moving on in the script, i.e., SA-perform or adding a new process step to the overall process, i.e., SA-add. SA-add was only performed in group I and relates to the request to brainstorm ideas with respect to four themes that were set by the group.

Information about comments made by performer gives first clues in respect to participation and hence represents a further application example for this kind of perspective (cf. Figure 4). Here results depict that in each group, one person participated relatively a lot to the collaboration process. When analyzing the control-flow perspective in respect to action/action types it is now possible to deduce process behavior over time. Figures 5.1 – 5.3 depict the course of the process for each of the three groups. The x-axis denominates the amount of actions made; the y-axis depicts time sequences in the duration of 5min. SA and UL actions are aggregated to ease interpreting the figure. All other actions are split for action type N.

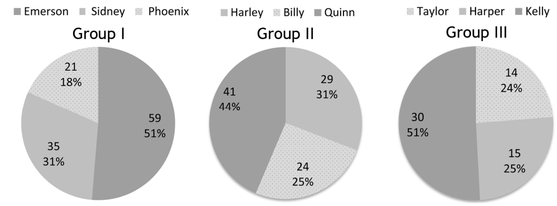

Figure 4: Distribution after performer

By comparison, the three groups differ which would not be observable when just focusing on the counted values that constitute the distribution perspective. As mentioned earlier, the highly process structured script defined three phases and proposed time allowances, brainstorming (app. 30min), outlining and drafting (app. 80min), group reviewing and finalizing (app. 20min) that can be read of the figure.

When focusing on the first phase brainstorming, each group behaved differently. Group III spent the first 15min on questioning and answering (Q&A) that is represented in the graph with a light grey dotted line. By taking the communication logs into account, they first synchronized on what the process script asked them to do, e.g., Taylor 2:49: “Every one of us must write 1500 to 2500 words or all together?”, and clarified questions, e.g., Kelly 2:51:”All together of course”. Next, actions related to negotiation started and lasted for about 25min. The brainstorming phase in group I started out with performing many negotiation actions within the first 35 min. An interesting observation is that during the brainstorming phase many proposals were accepted simultaneously. Then, app. 30min of Q&A followed. The process script, however, asked the participants to first collect ideas (thinkLet: DirectBrainstorm), select and argue for relevant ones (thinkLet: FastFocus/BoomWagon) with a subsequent discussion of collected comments and agreement of the report’s structure (thinkLet: ReviewReflect). Therefore, the peak of accept actions were intended to occur after a peak in proposition actions. Moreover, the communication logs show that time was spent on coordination during the Q&A session, e.g., Sidney 3:19 “Who is going to adapt the template to match the recommended structure?”, effort distribution, e.g., Phoenix 3:23 “I take the 2nd point – is that ok?”, as well as on process script questions, e.g., Phoenix 3:26 “How many pages do we need?”. In contrast to the other groups, group II was characterized by a rather long, app. 50 min, brainstorming phase that was filled with many negotiation actions. When focusing on the action “propose”, black solid line, data showed that the group adhered to the given highly process structured script, which requested to propose many ideas at the beginning of the writing activity. Based on this information we can conclude that this group adopted provided adapted thinkLets for "generate" and "reduce/clarify/evaluate" during the brainstorming phase.

During the second phase, outlining and drafting, the groups I and III behaved rather similar and had at least 30min where no comment was exchanged. Scarce communication was only spotted for group II which was mostly related to Q&A.

During the last phase group III showed rather different behavior than the other groups, since they performed a number of negotiation actions. Here the communication logs showed that the group was stuck and required further means for structuring their brainstorming ideas. For this purpose they brainstormed and searched for models or approaches that were useful for structuring their recommendations, e.g., Taylor 4:42 "I found some models. I think this model is best fit for our case.".

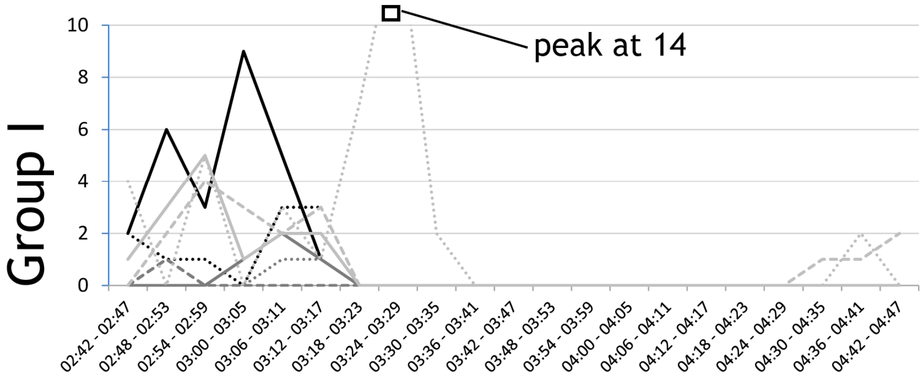

Figure 5.1: Time sequence of action types - split for negotiation (group I)

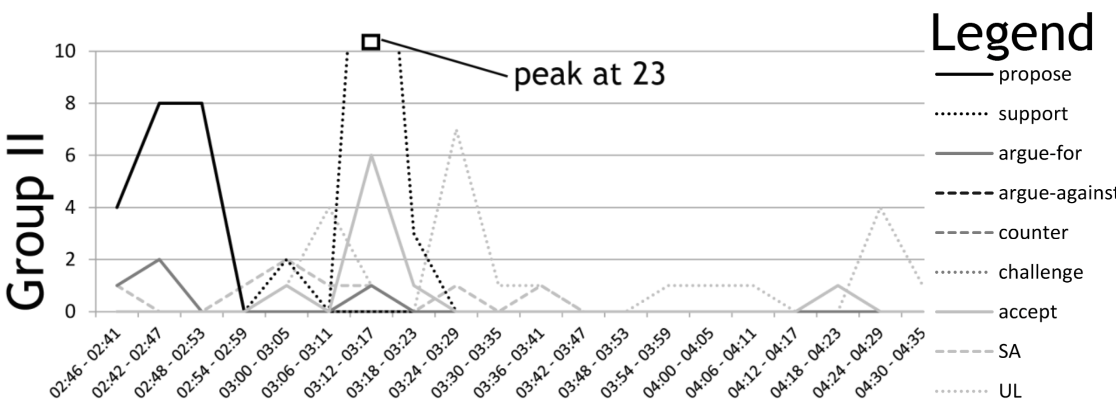

Figure 5.2: Time sequence of action types - split for negotiation (group II)

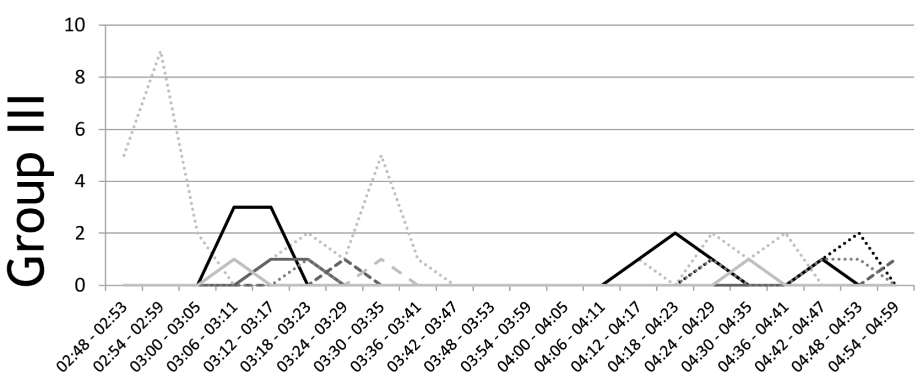

Figure 5.3: Time sequence of action types - split for negotiation (group III)

Besides the control-flow perspective in respect to applied actions and action types, also the participation of performers can be visualized. Figure 6 depicts the ordering of comments made by performers. In the case of group II, participation was mostly equally distributed with a peak at the end of the brainstorming phase. This peak was already visible in Figures 5.1 – 5.3, however, with this participation information we can deduce that all members more or less equally supported ideas. Furthermore, it is observable that at the end of the process participant "Billy" discontinues participating for the last 15-20min.

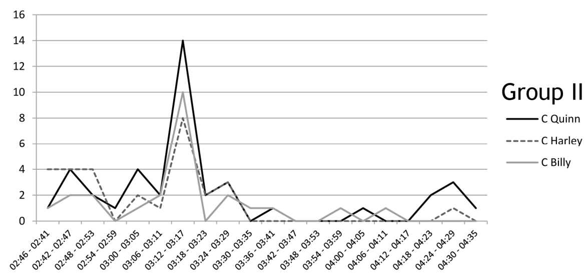

Figure 6: Time sequence of participation Group II

For the other groups, this rather equal distribution across the collaboration process is not observable anymore. In both groups there are two participants that have spikes in comments made at the beginning of the process.

An additional source for information provides the (F)HM result. Within Figures 5.1 – 5.3 it was depicted that Group II adhered to the given process script. However, the most likely sequence of actions is not yet visualized. (F)HM, a process mining plug-in for ProM, allows mining the main behavior by extracting likely relationships between actions based on direct, indirect successors and looping actions. The result is a so-called heuristic net (cf. Figure 7) which depicts the likelihood that an action follows a certain (other) action or loops. Each rectangle depicts an action with the information how often it occurred within the eventlog. The arrow between actions denominates the relation between them and gives information how certain we can be that one action follows another. The figure depicts that in case that a proposal was discussed it was likely that it was either supported or an argument was given, before it was finally accepted. Other rather strong dependencies exist between "ask" followed by "clarify", "clarify" followed by "ask", "support" followed by "accept". To receive this information each single proposal and related other actions needs to be marked during the coding, phase two, with an identifier. For a more detailed explanation regarding the computational steps of the mining algorithm, refer to [29,30].

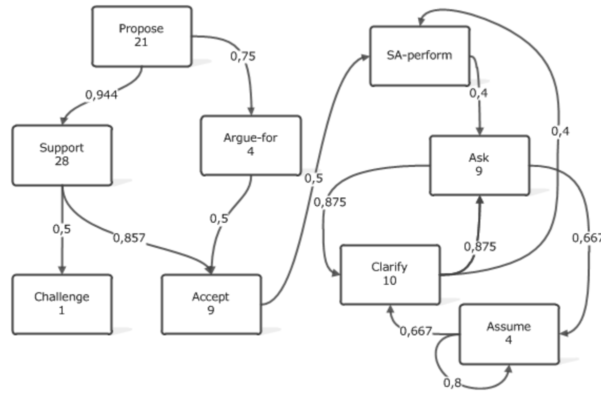

Figure 7: HeuristicNet for group II

This section provided results that were gathered in a laboratory experiment investigating the impact of process structure in distributed, synchronous groups during collaborative writing. The CoPrA technique was exemplified and generated a number of interesting finding which will be discussed within the next section.

5. Discussion and Limitation

In this paper we proposed the process analysis technique CoPrA which offers several valuable insights into groups' processes during the execution of collaborative activities. Within Section 4.1 we listed five questions that CoPrA should strive to answer based on a case example. Each of these points will now be discussed based on the observations that were made in the previous section. Answering these questions depends on the applied coding schema as other coding schemas might not cover actions for negotiation and the thinkLet one chooses as this assumedly strongly influences the behavior of groups.

- *Can we observe if and when participation of group members changes over time?*

The analysis provided information on the extent of participation per individual group members in the overall collaboration process or within phases. It also visualized whether group members dominated within a collaboration process or did not participate at all. Changes of participation can be indicated with respect to, e.g., participants who highly contribute at first and show low or no participation at the end of a process/phase. Causes for this phenomenon can then be investigated by contextualizing this observation with qualitative analysis of communication logs.

- *Can we observe if and when phases or patterns of collaboration change over time?*

The process script explicitly stated three phases which were observable in the control-flow perspective of performed actions. Moreover, it was visualized how much time the groups required for performing the "generate" and "convergence" thinkLets, since the script asked them to propose creative ideas at first and then support those which are worthy for further coverage. This kind of observation seems to be valuable for the pre-test of an experimental setting to inform the researcher how much time should be allocated for a thinkLet. Additionally, when having an experimental design with dependent samples, information could be gathered whether each reuse of a thinkLet reduces required time and hence shortens a specific collaboration phase.

- *Can we observe if participants adopt the provided experimental treatment, here the given process structure?*

The analysis showed that one group fully adopted the given process structure whereas other groups did not. For instance, during the brainstorming phase the group performed first many propositions (Direct Brainstorm) and subsequently supported and agreed on the best ones (FastFocus/Boomwagon) as intended in the highly structured process script. In our case, process structure was described by defining specific steps for negotiation, i.e., that the group should first propose any idea, and in a next step discuss each idea regarding its usefulness. Moreover, the amount of occurred SA-add actions provided further relevant information since it indicates if the provided process structure is appropriate. In the case that participants added intermediate process steps to reach their goal, this could imply that the amount of process structure is not yet suitable and redesign of the thinkLet should be considered. For this interpretation, however, it is useful to revisit communication logs to find the cause of these process additions.

- *Can we observe how deeply and comprehensively participants negotiated?*

The analysis also provided information how deeply groups discussed ideas due to the amount of negotiation action types. Based on the ratio of proposals and positively connoted actions, e.g., argue-for and support, we can infer whether the overall process was consensus-oriented. Additionally, the comparison of the number of propositions made and ideas accepted or rejected informs us whether each idea was discussed until the end or implicitly accepted or rejected during the drafting phase.

- *Can we observe sequences of actions that can be interpreted as candidates for causal dependencies?*

The analysis, conducted with the (Flexible)HeuristicMiner of ProM, supported further interpretation of collaboration processes by visualizing the most likely sequences of actions for a group. Consequently, a more abstract representation of the collaboration process could ease discussion related to similarities and differences among groups.

The laboratory experiment and the analysis technique suffer from a few limitations. Besides the adapted thinkLets, the provided process structure also included process support to use the tool and to perform the collaborative writing task. Hence, the presented development of action and action types is also influenced due to the adaptions that have been made to the process scripts. Even though results of the analysis show interesting findings, the interpretation of communication logs on the basis of a selected coding schema appeared to be demanding and requires researcher triangulation. Comparability of results between studies can only be achieved when all three phases of the analysis technique are known and transparent.

6. Conclusion and Outlook

Previous research in CE has outlined that there is a need to conduct further research on how groups accept, adopt and adapt repeatable collaboration processes. We argued that current performance criteria mainly take a retrospective focus on the end-product of a collaboration process, but not necessarily on the process itself. The proposed process analysis technique CoPrA does not only allow observing collaboration processes and patterns with respect to developments over time, but also allows interpreting observation within their context based on coded communication logs. By applying analyses from a distribution and control-flow perspective, the technique can offer valuable insights on similarities and differences observed between and within groups. In particular, it visualizes changes in participation or phases of collaboration, whether experimental conditions have been adopted, the extent of negotiation and most likely action sequences.

Future research should investigate how internal validity of an experiment can further be augmented by considering multiple sources for data collection. Besides coding of communication data, also a simultaneously developed artifact, e.g., a document, and its gradual development over time could be considered for data analysis. Additionally, the analysis of activity streams of participants, logging all actions across applications used in support of individual activities during collaboration can further extend understanding of collaboration processes.

Acknowledgements

This work was co-funded by the European Commission under the Information and

Communication Technologies theme of the 7th Framework Programme, Integrating Project ARISTOTELE (contract no. FP7-257886).

7. References

[1] Aalst, W.M.P.V.D., Exploring the CSCW spectrum using process mining, Advanced Engineering Informatics 21, 2, 2007, pp. 191-199.

[2] Abedin, B., Investigating the Trend of Non-Task Social Interactions in Online Collaborative Learning Environments, Proceedings of 44th HICSS, 2011

[3] Andriessen, J.H.E., Working with Groupware: Understanding and Evaluating Collaboration Technology, Springer Berlin Heidelberg, 2003

[4] Briggs, R., Kolfschoten, G., de Vreede, G.-J., and Dean, D.L., Defining Key Concepts for Collaboration Engineering, Information Systems, 2006, pp. 121-128

[5] Briggs, R.O., de Vreede, G.-J., Nunamaker, J.F., and Tobey, D., ThinkLets: Achieving Predictable, RepeatablePpatterns of Group Interaction with Group Support Systems (GSS), Proceedings of 34th HICSS, 2000

[6] Briggs, R.O., de Vreede, G.-J., and Nunamaker, J.A., Collaboration Engineering with ThinkLets to Pursue Sustained Success with Group Support Systems, Journal of Management Information Systems 19, 4, 2003, pp. 31-64.

[7] Davis, A., Badura, V., and de Vreede, G.-J., Understanding Methodological Differences to Study Convergence in Group Support System Sessions, In R.O. Briggs, P. Antunes and G.-J. de Vreede, Groupware: Design, Implementation and Use, Springer, 2008, pp. 204-216.

[8] Davis, A., de Vreede, G.J., and Briggs, R., Designing thinkLets for convergence, AMCIS, 2007

[9] DeSanctis, G. and Gallupe, R.B., A Foundation for the Study of Group Decision Support Systems, Management Science 33, 5, 1987, pp. 589-609.

[10] Drucker, P.F., The Age of Social Transformation, The Atlantic Monthly 274, 5, 1994, pp. 53-80.

[11] Grudin, J., Computer-Supported Cooperative Work: History and Focus, Computer 27, 5, 1994, pp. 19-26.

[12] Hayes, N., Boundless and bounded interactions in the knowledge work process: the role of groupware technologies., Information and Organization 11, 2, 2001, pp. 79-101.

[13] den Hengst, M., Dean, D.L., Kolfschoten, G., and Chakrapani, A., Assessing the Quality of Collaborative Processes, Proceedings of 39th HICSS, 2006

[14] Jonassen, D.H. and Kwon, H., Communication patterns in computer mediated versus face-to-face group problem solving, Educational Technology Research and Development 49, 1, 2001, pp. 35-51.

[15] Kamal, M., Davis, A., Nabukenya, J., Schoonover, T., Pietron, L., and de Vreede, G.-J., Collaboration Engineering For Incident Response Planning: Process Development and Validation, Proceedings of 40th HICSS, 2007

[16] Kelloway, E.K. and Barling, J., Knowledge Work As Organizational Behavior, International Journal of Management Review 2, 3, 2000, pp. 287-304.

[17] Kolfschoten, G.L. and de Vreede, G.-J., A Design Approach for Collaboration Processes: A Multimethod Design Science Study in Collaboration Engineering, Journal of Management Information Systems 26, 1, 2009, pp. 225-256.

[18] Kolfschoten, G.L., Briggs, R.O., Appelman, J.H., and de Vreede, G.J., ThinkLets as building blocks for collaboration processes: a further conceptualization, Groupware: Design, Implementation and Use, 2004, pp. 137–152.

[19] Kolfschoten, G.L., Theoretical Foundations for Collaboration Engineering door, Delft University of Technology, 2007

[20] Liu, C. and Tsai, C., An analysis of peer interaction patterns as discoursed by on-line small group problem-solving activity, Computers & Education 50, 3, 2008, pp. 627-639.

[21] Lowry, P.B., Nunamaker Jr, J.F., Curtis, A., and Lowry, M.R., The impact of process structure on novice, virtual collaborative writing teams, IEEE Transactions on Professional Communication, 48, 4, 2005, pp. 341–364.

[22] Mattessich, P.W. and Monsey, B.R., Collaboration: What makes it work. A review of Research Literature on Factors influencing Successful Collaboration, Amherst H. Wilder Foundation, Saint Paul, 1992, pp. 3-56.

[23] Nunamaker Jr., J.F., Briggs, R.O., Mittleman, D.D., Vogel, D.R., and Balthazard, P.A., Lessons from a dozen years of group support systems research: a discussion of lab and field findings, Journal of Management Information Systems 13, 3, 1996, pp. 163-207.

[24] Rittgen, P., Negotiating models, Advanced Information Systems Engineering, Springer, 2007, pp. 561–573.

[25] Schultze, U., A confessional account of an ethnography about knowledge work, Management Information Systems Quarterly 24, 1, 2000, pp. 3-41.

[26] Trauth, E.M. and Jessup, L.M., Understanding computer-mediated discussions: positivist and interpretive analyses of group support system use, Management Information Systems Quarterly 24, 1, 2000, pp. 43–79.

[27] de Vreede, G.-J., Fruhling, A., and Chakrapani, A., A Repeatable Collaboration Process for Usability Testing, Proceedings of 38th HICSS, 2005

[28] de Vreede, G.J., Briggs, R.O., and Massey, A.P., Collaboration engineering: foundations and opportunities, Journal of the Association for Information Systems 10, 3, 2009, pp. 7

[29] Weijters, A. and Ribeiro, J., Flexible Heuristics Miner (FHM), 2010

[30] Weijters, A., van der Aalst, W.M.P., and de Medeiros, A.K.A., Process mining with the heuristics miner-algorithm, Technische Universiteit Eindhoven, Tech. Rep. WP 166, 2006

[31] Wolff, E.N., The growth of information workers in the U.S. economy, Commun. ACM 48, 10, 2005, pp. 37-42.