Trust and Distrust in Inter-Organizational Software Development Networks

Jarkko Pyysiäinen
Jarkko.Pyysiainen@hut.fi
Helsinki University of Technology, Software Business and Engineering Institute
P.O. Box 9600
FIN-02015 HUT Finland

ABSTRACT
This paper examines trust and distrust in inter-organizational software development networks in the light of thematic interviews collected from seven software development networks. According to the theoretical perspective adopted, the choice between trust and distrust in uncertain conditions depends on the relational resources available to the interactants. Using qualitative analysis, trust building is examined from the perspective of relational resources available for members of the network in the definition of their relationships. The results indicate that the most important relational resource for trust building in projects in stable environments was an early establishment of mediating link persons, equipped with salient role expectations. In projects with unstable environments the most important relational resources for trust building were well-internalized, reliable delivery cycles and inter-organizational processes. The results suggest that the work process itself may contribute significantly to the development of trust and distrust in networked conditions, e.g., by facilitating or preventing visibility. Results also support Luhmann’s idea about trust and distrust as functionally equivalent but opposing strategies to reduce high uncertainty.

1 INTRODUCTION
Software development is an inherently complex activity and software development projects are always faced with some uncertainty. In the case of inter-organizational software development networks complexity and uncertainty pose even bigger challenges than in colocated development. The needs of teams and companies to react and respond to changes in networks vary greatly depending on, e.g., the nature of the task and the causes of uncertainty in question. In the most challenging cases the development work requires active negotiation, iterative improvements and assistance between the members from different teams. In such cases unpredictability among the members of the network is high and consequently also the need to show cooperativeness and readiness to communicate is very great. In order to succeed in the project, the members surrounded by such challenging circumstances face a strong need to trust each other.

A growing body of literature demonstrates the important benefits of trust for organizations and their members (Kramer 1999, Mishra 1996). Even more importantly, from the perspective of this study, trust is seen as a necessary element in facilitating the functioning of networked organizations (van der Smagt 2000, Jarvenpaa & al. 1998, Miles & Snow 1992) and software outsourcing relationships (Heeks & al. 2001, Sabherwal 1999). While on the one hand trust building seems to be a promising mechanism for overcoming many difficulties related to networked software development, it may on the other hand be precisely the virtual and distributed contexts that constrain the development of trust between companies and teams (e.g., Jarvenpaa & al. 1998). Lack of face-to-face interaction
and informal communication may be especially troubling. Hence, the possibility of distrust seems to be lurking in distributed, inter-organizational networks as well. Luhmann (1979, 1988) has provided an interpretation of the dynamics of trust and distrust, which might be helpful in the analysis of micro-social processes of trust building in such conditions.

This paper summarizes the initial findings of a study that aims at gaining a thorough understanding of the nature of and needs for trust in challenging inter-organizational and geographically distributed software development projects. This study aims at shedding new light on such characteristics in networks, which on the one hand demand existence of trust and on the other hand facilitate or prevent the development of trust. Since trust is considered as one of the most difficult concepts to handle in empirical research due to its ambiguous nature (e.g., Misztal 1996, 95), another central objective of this paper is to present an approach for the analysis of trust in empirical, textual data, and to generate discussion about the potentials of such directions of inquiry.

2 THEORETICAL BACKGROUND

For the purposes of this study, the point of departure in the conceptualisation of trust is communicative behavior in social interaction. When analysing communication between interactants, one can always make a distinction between two separate dimensions of communicative behavior: content and relationship dimensions. The content dimension refers to the topic of conversation, the specific subject matter. It may also be compared with the information contained in a message. The relationship dimension refers to properties of communicative behavior, which guide interactants in the interpretation of the content. The relationship dimension defines the nature of the relationship and comments on the content exchanged in interaction. It may be compared with the style in which a certain message is expressed. (Fisher & Ellis 1990, 93-94.) If we look at trust from the perspective of this distinction, it seems suitable to consider trust as a central element of the relationship dimension.

Trust is typically a relationship between interactants, and it entails a certain relational logic that guides the interpretation of messages exchanged between interactants who trust each other. Trust always defines relationships between interacting parties in one form or another, e.g., as trust between friends, as distrust between enemies, or as suspicion between competitors. In this sense trust can be seen as a general, relational, communicative message. This approach is also consonant with the conceptualisations of trust developed by Luhmann (1979, 1988).

The theoretical background summarizes the elements I have utilized in efforts to conceptualise the challenges and potentials of trust building in inter-organizational networks from an interactionist perspective, where focus is on the actual communicative relationships between parties.

2.1 RELATIONAL RESOURCES FOR TRUST BUILDING

A common feature in many definitions of trust is that trust implies a readiness to be vulnerable to the actions of others, in spite of the acknowledged possibility of being disappointed (Deutsch 1973, Luhmann 1988, Mishra 1996). However, it seems that the phenomenon of trust is multiply determined in the sense that several factors can influence its development and disappearance (cf. Luhmann 1988). Hence, it may be useful to approach the factors influencing its development as relational resources. The concept of relational resources refers to such messages, which introduce a shared meaning of some kind to the interaction and provide the interactants with some definition of their relationship. By identifying the central components of trust in various definitions presented in the literature, we can form a general class of relational resources, which are shown to be effective and critical in the
development of trust. My intention is to summarize here briefly such elements and antecedents of trust, the existence of which could in turn be examined in the light of the empirical data. Thus, the following elements are presented as possible resources to be used in the relational framing of the interactions between members of software development networks.

Mishra (1996) has presented an extensive summary of the central components included in influential definitions of trust: according to his summary, four distinctive components capture the content domain of trust literature, namely competence, openness, concern and reliability. These components can be seen as the central dimensions that are called into question when assessing the trustworthiness of other persons.

Another interesting finding is that without the accumulation of experiences and testing of the relationship in actual interaction, trust is not likely to deepen and last between parties. In this vein, e.g., van der Smagt (2000) argues that trust prevails if each of the interacting parties acknowledges the right of the other parties to assess the intentions of their acts. Here a critical point seems to be that a readiness to question one’s own assumptions concerning the topic of interaction is a central source of trust; a rigid adherence to one possible solution does not display a tendency to be concerned of the interests of the other, thus undermining seriously the potentials for the development of trust. In this sense, a reciprocal dialogue is likely to be a strong resource for the development of trust between interacting parties (van der Smagt 2000).

Further, Kramer (1999) has lined out traditional bases of trust within organizations, here supplemented with some ideas to extend these notions to the context of networked organizations:

**Dispositional trust:** Predispositions to trust or distrust others tends to be correlated with other personal orientations and past experiences. Certain features in the behavior of others become associated with stable expectations, and persons may develop a disposition to extrapolate from earlier trust-related experiences when encountered with novel social situations. In this sense the threshold to trust may vary significantly between persons and also between members from different cultures (see e.g. Hofstede 1997). The implication seems to be that familiarity with the specific interaction-styles of others may prove to be crucial in decisions to either trust or distrust. Such may be the case in heterogeneous conditions of a network, where there is likely to be great variability in the criteria of trustworthiness that people have come to expect from the behavior of others. Thus in the analysis of the data this dispositional trust is viewed from the perspective of interaction orientations.

**History-based trust:** Trust thickens or thins as a function of cumulative interaction. Individuals’ judgments about others’ trustworthiness are partly anchored in a priori expectations about others’ behavior, and these expectations change as subsequent experience either validates or discredits them. Reciprocity in exchange relations enhances trust while violation of reciprocity erodes it.

**Third parties as distributors:** Third parties are important because of their ability to diffuse trust-relevant information via informal communication and gossip. On the basis of this kind of mediated information it becomes possible to transfer expectations of existing embedded relationships to newly formed ones.

**Category-based trust:** Common categories function as vehicles for perceiving common identities and common goals. A shared membership in a salient category (woman, researcher) can provide basis for presumptive trust and a sense of familiarity. Membership in a category is associated with a tendency to attribute positive characteristics to other ingroup members. In addition, the phenomenon of self-categorization (e.g., Tajfel & al. 1971) suggests that there is a symmetrical relationship between goal-perception and identification: studies have shown that allocation of trust seems to follow group boundaries even in situations where the identification is based on perception of such minimal common denominators as similar taste preferences or opinions concerning a trivial topic. Thus, it can be argued
that in situations characterised by geographical dispersion and salient group boundaries – as in the case of distributed software development projects – the accentuation of common goals and interests becomes extremely critical.

**Role-based trust:** Often it is not so much the person in the role that is trusted but the system of expertise that produces and maintains role-appropriate behavior. In this sense trust can be seen also as depending on the system that is represented in the role – roles lessen the need for repeatedly negotiating trust when interacting with others.

**Rule-based trust:** Trust based on internalized rules rests not on an explicit contract but on socialization into the structure and practices of the organization. If socialization is high and common principles are well internalized, mutual trust can acquire a taken-for-granted quality.

These traditional bases of trust within organizations can be taken as a point of reference when charting the conditions of trust development in networked, inter-organizational contexts. Common to all aforementioned resources of trust is that all promote reciprocity between interacting parties.

### 2.2 SOCIAL COMPLEXITY, TRUST AND DISTRUST

The concept of social complexity is based on Luhmann’s (1979, 1988, 1989) writings and his theory of trust as a strategy to reduce social complexity. Luhmann’s theory deals with societal phenomena at a very abstract level, and for the purposes of this study my intention is to apply principles of his theory to the organizational level. Social complexity is conceived as the relation between a social system and its environment. No system can achieve a point-to-point adaptation with its environment, but the more there are possibilities to represent and communicate consequences of and alternatives for action within a social system, the higher the ability of the system to react to events in its environment. This representational and communicative ability to react to events in environment is the level of social complexity of the given social system. On the other hand, reduction of social complexity can only be performed within the system, by means of structurally restricting its reactions to the environment. Accordingly, greater complexity of the social system allows better possibilities to react to events in the environment. (Luhmann 1989, 9-13.) In other words social complexity refers to the uncertainty and predictability of events and the communicative acts used to reduce uncertainty and risk within a social system. In this sense an organization whose members are able to reflect and communicate a wide variety of possible consequences of their actions in relation to their environment (e.g., other organizations), manifests a high level of social complexity.

In the case of distributed software projects the needs of social systems (e.g. customer) to establish communicative point-to-point adaptations with their environment (e.g. subcontractor) vary greatly depending on, e.g., the nature of uncertainty of the task in question. Social complexity seems to be a useful concept in understanding the meaning of these differences in project environments.

Luhmann (1979, 1988) has presented a useful distinction of the basic modes of responding to social complexity. The conditions of familiarity, confidence, trust and distrust all represent qualitatively different modes of asserting expectations and of reducing social complexity. Familiarity is a necessary condition for one’s capability to form stable orientations towards the environment; familiarity is also a necessary condition for the development of trust or distrust, but alone it will only allow one to cope only with fairly low levels of social complexity. Familiarity resembles closely such relational resources of trust as common history and shared category membership. The second condition for the development of stable trust is the mode of confidence, which depends in turn upon a certain amount of familiarity of the target. Confidence is based upon expectations of certain normal practices and norms; in state of confidence no alternative ways of doing things are actively thought
about, instead a certain scenario is taken for granted. Confidence resembles closely such relational resources of trust as internalised common rules and role expectations. Together familiarity and confidence form a basis for trust development and maintenance. Trust allows most alternatives for action in the face of risk and uncertainty; trust makes social complexity tolerable by making the one who trusts indifferent to many conflicting and disappointing event sequences, preserving a certain positive “as if” expectation. However, in order to be persistent, trusting behaviour presupposes that the previous stages of familiarity and confidence do not lapse into serious disappointments. Should the expectations lapse into disappointments, reduction of social complexity would happen in the form of distrust. Trust and distrust are seen as functionally equivalent mechanisms to reduce social complexity, but while trust makes social complexity liveable and allows one to utilize the possibilities thus generated, distrust eliminates social complexity altogether. Distrust means refraining from reciprocating actions, which eliminates the possibilities of both disappointment and opportunity. (Luhmann 1979, 1988.) Luhmann’s ideas support the interpretation of development of trust as depending on suitable relational resources, which function to provide the interactants with sufficient familiarity and confidence. These limits of social complexity reduction are depicted in Figure 1.

![Figure 1 Limits of social complexity reduction (Adapted from Luhmann 1979, 1988.)](image)

To conclude the theoretical discussion, the concept of trust adopted in this study can be described as a relational process, which is activated between interacting parties in situations provoking social complexity, i.e., uncertainty and perceptions of both risk and opportunity. According to this view, conditions characterized by increasing social complexity set forth the pressures to reduce this complexity by opting either to trust or to distrust (see Figure 1). The choice between trust and distrust is likely to grow from accumulated interaction, but more importantly, the development of either trust or distrust is seen as depending on the relational framing resources available to the interactants (described in chapter 2.1). The organizational context of the network may or may not provide some facilitating bases of trust (e.g., mediating links, shared rules, role expectations) upon which to draw in the framing of the interactions. However, if the organizational context does not provide any support for the framing of interactions in terms of trust (e.g., absent organizational bases of trust), it is probable that the reduction of social complexity will be accomplished rather by distrust than by trust.

The interesting question to be answered is to which extent the members of a distributed software development network can draw on these relational resources of trust when interacting with each other and to which extent supportive practices must be utilized to facilitate trust building in networked conditions. These questions and theoretical insights will be examined further in the light of the empirical data.
3 METHODOLOGY AND EMPIRICAL DATA

The study follows the qualitative case study approach (Yin 1994). The data was collected from seven inter-organizational software development networks of six Finnish software companies. The focus of the study was on networked projects that involve at least two companies: a customer and a subcontractor. Most networks studied involved more than two organizational parties. Characteristics of the studied networks are presented in Table 1. All selected companies except one were large and nationally well known. The data consists of transcribed thematic interviews of the project personnel and managers (N=36).

Table 1 Characteristics of the selected case networks and interviewees

<table>
<thead>
<tr>
<th>Network</th>
<th>Software type</th>
<th>Customer interviewees</th>
<th>Subcontractor interviewees</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Software product</td>
<td>Partnership responsible</td>
<td>Project manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Project manager</td>
<td>Liaison person</td>
</tr>
<tr>
<td>B</td>
<td>Software product</td>
<td>Product director</td>
<td>Project manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Process developer</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>Bespoke system</td>
<td>Partnership responsible</td>
<td>Project manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Process developer</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 project managers</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Liaison person</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>Software product</td>
<td>Marketing director</td>
<td>Project manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 project managers</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 developers</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>Embedded software</td>
<td>2 process developers</td>
<td>Project manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 partnership responsibles</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>Embedded software</td>
<td>Process developer</td>
<td>Project manager</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2 partnership responsibles</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Project manager</td>
<td></td>
</tr>
<tr>
<td>G</td>
<td>Software product</td>
<td>Marketing director</td>
<td>Managing director</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Project manager</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Developer</td>
<td></td>
</tr>
</tbody>
</table>

Qualitative analysis of the data was done by using both theoretically derived categories and categories generated from the data. The first step of the analysis was to conduct a cross-sectional indexing (Mason 1996, 111-127) of the data, which means systematical classification of the whole data corpus. This cross-sectional indexing was done from an interactionist perspective, by searching for expressions of social complexity in the data. The textual unit to be identified was a topic or content of interaction, which pushed a social system in the network (e.g. customer, supplier) to accomplish an adaptive reaction of some kind in relation to actions performed in its environment (e.g. customer, supplier). This cross-sectional indexing aimed at identifying occasions, which according to Luhmann’s ideas mark the presence of social complexity and activate the need for trust (see chapter 2.2). According to Mason (1996, 109-117) this type of classification of the data may be called interpretive reading, because here some textual expressions are seen as reflections of more abstract phenomena on the basis of certain interpretation criteria (here derived from Luhmann’s theories).

The second step of the analysis was to classify the observations thus generated according to topics of interaction. This type of classification was data driven and can be called literal reading (Mason 1996, 109-110) of the classified text strips, since here identification of the contents of interaction was based on characteristics that are directly recognizable in the text (e.g., discussions about project.
assignment or test results). Together these first and second steps of the analysis resulted in fourteen categories, which reflected the various manifestations of social complexity in interaction.

Third step of the analysis was to trace the relational resources of trust available to the interactants when social complexity was encountered. The categories of relational resources were presented and marked on italics in chapter 2.1; i.e., the classification of the relational resources was based again on interpretive, theory driven reading of the text strips. When necessary, I returned to the data to complete the picture of the attributes that framed the interactions in occasions of social complexity. Hence, the third step resulted in eleven relational resource categories, which represented resources for trust building in cases of increasing social complexity.

The fourth step of the analysis was to trace the results of the interaction sequences described in the classified text strips from the perspective of trust and distrust. Classification of instances of trust and distrust was based on interpretive, theory driven reading, where trust was interpreted as the willingness to be vulnerable to the actions of others in spite of the possibility of being disappointed (intention to perform adaptive behaviors). Distrust was interpreted as the choice to refrain from reciprocating actions (refusal of adaptive behaviors).

In the fifth step literal, data driven classification was applied in order to identify, what kind of characteristics of project work were related to the increases in social complexity and need for various relational resources. Consequently the generated observation classes were grouped around the common project phase related denominators. The resulted categories represent analytically distinguished project phases that all entail differing demands on interaction, not clearly separate chronological project phases. Further, these project phases were grouped around common project environment related denominator, namely stability of the environment. The categories resulting from this grouping represent two general project types with slightly different conditions for trust building. It is worth emphasizing that the identification of these project phases and types is based on the analysis of the data, not on the models existing in the software literature.

4 RESULTS: THE MICRO-SOCIAL PROCESSES OF TRUST AND DISTRUST

Results from the classification of the data are summarised in Table 2. The generated framework of categories indicates that the micro-social processes of trust building differ according to type and phase of the project in question.

Table 2 Classification framework used to identify the sources of social complexity and the resources for trust building

<table>
<thead>
<tr>
<th>Content dimension (topic of interaction):</th>
<th>Assignment</th>
<th>Build integration</th>
<th>Changes</th>
<th>Coding standards</th>
<th>Development solutions</th>
<th>Errors</th>
<th>Functionality</th>
<th>…</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relationship dimension (resources for trust building):</td>
<td>Concern</td>
<td>Competence</td>
<td>Openness</td>
<td>Reliability</td>
<td>Interaction orientations</td>
<td>History</td>
<td>Third parties</td>
<td>…</td>
</tr>
<tr>
<td>Strategy to reduce social complexity:</td>
<td>Trusting orientation</td>
<td>Distrusting orientation</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project phase related activation of content and relationship dimensions:</td>
<td>Relationship initiation</td>
<td>Goal definition and task planning</td>
<td>Execution</td>
<td>Delivery</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Project environment related needs for trust building:</td>
<td>Projects in stable environments</td>
<td>Projects in unstable environments</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The sources of social complexity (categories on the content dimension in Table 2) vary depending on the stability of the environment in the project and depending on the phase of the development work. These differences in social complexity, in turn, are paralleled with a variance in the critical factors needed to maintain a trusting orientation between parties and to balance the turbulence generated by social complexity (categories on the relationship dimension in Table 2). In this chapter I present the project phase-related combinations of activated content and relationship categories in two project types, which are titled *projects in stable environments* and *projects in unstable environments*. In both project types I present examples of social complexity reduction achieved by trusting and distrusting orientations.

### 4.1 INITIATION OF THE RELATIONSHIP

The initiation of a relationship in the case of inter-organizational software development networks entailed a number of topics to be clarified and negotiated before the parties could proceed to the actual development work. However, on the content dimension of interaction there were some specific sources of social complexity that characterized stable projects and other sources that characterized more open and uncertain projects. The development of the collaborative relationship – trusting or distrusting orientation – seemed to depend on how these uncertain topics were framed on the relationship dimension. In this sense some initial arrangements can be identified, which guide the definition of the relationship between parties and seem to be quite difficult to renegotiate once the relationship has started to develop.

#### 4.1.1 Projects in stable environments

**Trust – the constructive path.** A major challenge in this type of project was to develop commitment towards common goals and objectives. Both the customer and the supplier preferred long-term cooperation, which would bring predictability, reduce learning costs and lead towards more mature common work practices. In this respect a significant source of social complexity was created by the need to agree on the continuity of the assignments and / or identify sufficiently motivating and rewarding tasks for the supplier. The crucial thing in the development of the relationship was to frame these topics of interaction with a relational orientation promoting *membership in a common category* (based on, e.g., some enduring common interest) and *concern of the interests of the other* (e.g., planning and informing about future assignments). When these relational orientations were established between interacting parties, the path for a trusting relationship was laid down, improving also the capabilities of the parties to cope with the social complexity inherent in the project initiation phase:

A: Well, we have a sort of moral contract, so that we know how much money subcontractors need in a month in order to run their business and keep their programmers. According to that estimation we are trying to figure out suitable projects for them, but at times it is quite difficult. (Network A, project manager of the customer.)

A: At the moment our subcontractor is also working on some more motivating assignments, because in the beginning the standard tasks are quite challenging to do, but once you learn it, it gets really boring. Q: Is that why you want them to try out also other kind of development work? A: Yes, that is the only reason, otherwise it would not be profitable to us to move any other kind of development there. Due to some issues related to product architecture, it is not easy for us to outsource things from our product development. (Network A, partnership responsible of the customer.)

Another challenge was related to the mutual fit between skills, roles and processes required in the development project. Often the customer could not form a detailed picture of the state-of-the-practice

---

1 In the presented quotations A: refers to the answers of interviewees and Q: to the questions of interviewers. These quotations are typical examples of the categories presented. Since all interviews were originally in Finnish, the quotations presented here are translations from the original ones.
at the supplier without a proper auditing and examination of the supplier’s capabilities. On the other hand, the suppliers could not assess the suitability of their skills, processes and roles to the expected assignments, unless customers were willing and able to provide detailed information about the nature of the tasks and the necessary attributes of a development process required in the projects. Knowledge and ability to arrange suitable training for the members of the network were in turn dependent on a proper clarification of these issues. Thus, the interaction concerning required skills, process, roles and training had to be framed on the relationship dimension with openness, reciprocal dialogue, rule internalisation and mediating link persons in order to provide the parties with the critical elements for trust building:

Q: Did all programmers participate in your collocated meeting in the beginning? A: Not all the time, rather so that we had a couple of short meetings with the managing director of the subcontractor, but otherwise almost whole personnel were present in our discussions. So we sat down there and took out a video projector and they were really impressed that we bothered to come over. We wanted all the personnel to be present in the clarification of the work, and I think it really did help and make clear that we really want to work with you, so it would be nice that we could send them deliverables that make their work more comfortable, but according to our experience it is not so obvious that such is the case, so let’s open the black box of the process a bit and see how it goes there. (...) Although theirs is not so much a real process, the projects are so small and the workers are so new. (...) I honestly think that we both understood things that we had not earlier understood, that we really have a common goal. (Network A, project manager of the customer.)

A: Yet there are problems at times in the beginning, since you cannot just drop by their rooms and have a chat. Maybe the worst things are such that those things which are not documented or described in the process, and anyway they ought to be done, are easily forgotten in this type of distributed development. Suddenly you notice that nobody has updated a document even though we discussed that. But now these role definitions help a bit, but you cannot define everything there either. So inside your own team it is easier for you to have a word and check out that did you remember to document that. But you rarely use e-mail to check out such issues, that who ought to do that. (Network B, project manager of the customer.)

Distrust – the destructive path. Failure to frame the aforementioned topics with adequate relational orientations in the beginning of the relationship led to disruptive inter-group processes in several cases. The inability to promote shared interests and mutually motivating incentives regarding the project assignment easily led people to opt distrust when encountered with increasing social complexity:

A: Well the company interface is pretty difficult. It has inevitably been the case that people are really skeptic about it, we recognized it for example with the Russian subcontractors that our people thought that they are certainly trying to cheat us. And you never have that with your own people. (...) And with that other subcontractor we had some kind of deep problem that whenever our developers get some work from the subcontractor which does not function properly, knowing that it’s not done by a friend of mine but someone who lives in Turkey and supposedly tries to do it cheap, then it raises immediate suspicions. And if the code does not function, then they punish the subcontractor and do difficult improvement requests on purpose, without even trying to keep them in schedule. Especially the testers felt that it did not motivate them a bit to test work that was done by strangers and contained a lot of errors. (Network A, project manager of the customer.)

Furthermore, destructive effects on collaboration and cooperation were also caused by the lack of openness, reciprocal dialogue and internalisation of common rules regarding skills, process and roles required in the upcoming projects:

A: When it comes to competence, like which subcontractor’s people are able to program properly, I trusted our project manager completely in that issue. The problem was not so much in the writing of the code but the understanding of the whole thing. That was lacking in the beginning, that we did not really know how to best clarify and facilitate subcontractor’s understanding. (...) And for the part of the Hungarian subcontractor, I think that there were no phases in project that would have really gone well or succeeded. At least they did not have the willingness to do the project, maybe they anyway had enough competence. Or then the other way around: no competence but willingness, but anyhow we did not get it out from them. (...) And finally they left the whole project. (Network D, marketing director of the customer.)
4.1.2 Projects in unstable environments

Trust – the constructive path. In the projects with more open environments and more innovative but uncertain objectives the incentive for the participants often already existed in the beginning of the relationship, from the perspective of the supplier not least in the form of challenging and innovative learning possibilities. Consequently the difficulty was not so much the perception of common interests or the dispersion of the background knowledge about the assignment, since the precise nature of the task was often uncertain and the supplier was much more involved in planning and specifying the tasks. Instead, the parties had to devote effort to make sure that the supplier’s competence suits precisely the field of the assignment, since the proper competence and experience had to be there right from the beginning. Otherwise the supplier could not take responsibility for the specification and planning. Thus, in the initial evaluation phase parties had to frame their assignment-related interaction with reciprocal dialogue, where they could assess the extent of competence and common experience base (common history) required in their relationship. These elements could provide the basis for trust building in unstable and turbulent environments, where the relationship between parties may end suddenly if the markets change or risky technology fails:

A: The subcontractor has competence and expertise on certain area, it has done related work earlier either with partners or as part of own product development. That means that they have their own interest to do such development work and it is profitable for them. And when they get to share our product, they also get to share our volumes. And they are ready to take the risk that either it has volume or it doesn’t. And according to that volume we share the results. (Network E, process developer of the customer.)

A: Well in the beginning subcontractors were taken in precisely because they had expertise, they actually did the first version of the requirements specification. (Network F, project manager of the customer.)

Another critical source of social complexity in the interaction between parties was the high degree of changes expected during projects. Frequent build integrations and synchronized inter-organizational processes were expected, otherwise the management of changing requirements and interface-connections was believed to be impossible. The customer expected that the supplier would be tightly integrated to their development process by internalising their development cycle and being ready to negotiate about changes without every time transforming the technical discussions into discussions about money. In this sense, interaction concerning development process and build integration was expected to be framed by rule internalisation, category membership and reliability. In other words, a high degree of assimilation and congruent behavior was expected in the manner of working. Ability to conform enough on these topics made it possible for the customer to trust the supplier, this being the first antecedent for the development of trust in the relationship:

A: One big area is related to all these technical equipments that are applied to carry out communication. How to manage requirements, how to manage errors, how to manage various issues related to product development, which are typically device-based solutions, the big challenge is how to integrate all these devices. So our aim is that as similar devices as possible would be used and then integrated. (Network E, process developer of the customer.)

Distrust – the destructive path. Failure to frame the aforementioned topics with adequate relational orientations in the beginning of the relationship led to difficulties also in the case of open projects. Insufficient awareness of the competence or lack of common knowledge regarding the technological field made parties hesitant about the suitability of the assignment:

A: In the beginning we did not know anything from the subcontractor’s Chinese site, only when we visited China did we get a hunch, we met these subcontractor’s workers and for the first time met the CEO of the Chinese site, whom we did not know even to be existing. The meeting at least helped to clarify a bit that who’s doing what and how many workers are there, since at that time we did not even know how many people are there participating in the project. (Network G, project manager of the customer.)
Inability to conform to the process and flexible technical negotiations also led inevitably to an imbalance of the relationship:

A: We are trying to tune this one subcontractor to the right development mode, but it does not quite succeed. It is not easy to find the common interest there, and it is pretty much due to the content of our basic contract, which provides the frameworks for our cooperation. (...) Price policy is one of the big issues here. If you aim at stable pricing, it leads to such big unit deliveries that it stiffens the relationship and change management turns always into price negotiation. Therefore only very small things can change without price negotiations in such conditions. Typically subcontractor reacts that okay if you want to change something then it costs you more, and we'd like to keep a fast development cycle since we do not know how it is going to turn out. (Network E, process developer of the customer.)

4.2 GOAL DEFINITION AND TASK PLANNING

When starting the actual development work in the project, the parties as social systems are encountered with a number of issues demanding mutual adaptations. Of course, the issues related to the planning of the project and specification of the tasks may be arranged, e.g., strictly in terms of coercive power and money, by insistence on standardized procedures and minimal communication, but a one-sided reliance on power or money in the definition of the relationship excludes the possibility of framing the relationship in terms of trust (see Luhmann 1979). Hence, the choices made in managing the sources of uncertainty and incongruity in the beginning also constrain the possibilities for trust building. In some cases a deep base of trust between parties may be required, whereas in other cases parties may fare well merely by adhering to more rigid norms and standards. Again, the degree of stability of the environment seems to make a difference.

4.2.1 Projects in stable environments

Trust – the constructive path. A major source of social complexity was generated first of all by the need to exchange sufficiently relevant task-related background knowledge between separate organizations. Clarification of the specifications and coding standards seemed to be key areas demanding good communication and reaction readiness at both organizations. Hence, openness and reciprocal dialogue seemed to be necessary ingredients of trust building on the relationship dimension. Critical distributors of information and tacit knowledge between parties seemed to be mediating link persons, who spent some time with the members of both organizations providing the people with an inside view to the state-of-the-practice at the other organization. These mediating links were especially important in building confidence in the commitment of members at remote sites when difficulties were faced in the beginning. Only mediating links could provide up-to-date knowledge of the sources of problems and incongruities between organizations, since in this type of projects the synchronization of the inter-organizational processes was not wanted or even possible. Mediating links could also inform the parties about the personal interaction orientations of the members at other sites, thus bridging the gaps between differing expectations about trustworthy behavior (e.g. “He just happens to act in a straightforward and short-worded manner, but does not intend to undermine the work of others.”). Mediating links could build some sense of common history and shared experiences, which would be otherwise improbable in the mode of quite independent work processes. However, the mediating links did not emerge by accident, the whole network had to support the establishment of mutually fitting role expectations:

A: So in our case it means that the subcontractor should be critical and have low threshold to communicate and make clarifying questions, for example like “how is this thing supposed to be, this does not make sense at all”. (...) And this was really good example that it was very useful to visit our Turkish subcontractor and spend a couple of days there. Earlier there was only one person occasionally communicating with us by e-mail, and on the other hand we did not have any visible contact persons. So we did not receive any up to date information concerning the progress of projects and absolutely no feedback from any kind of practical problems and stuff. But after we visited them and
talked with the people personally, clarifying various practical issues, the communication threshold lowered significantly. Now they contact even concerning small technical problems and development suggestions, like “I think this would be a useful solution, what do you think”. (Network A, liaison person of the customer.)

**Distrust – the destructive path.** Typical of this type of projects were lacking dialogical contacts at the time when tasks and specifications were communicated. When the customer was unfamiliar with the personnel and culture of the subcontractor, it was difficult for the customer companies to assess, whether the subcontractor had internalized the specifications properly or whether they were just reluctant to make clarifying questions. On the other hand subcontractors were often left alone with the vague specifications and insufficient background knowledge, because there were no stable media for the collaborative coordination of the open questions in the specifications. Generally a difficult issue for the companies was to figure out, what kind of information the other party would actually need. Additionally, customers had not included proper plans for supportive activities to their processes. Lack of these supportive relational orientations undermined the development of trust and made distrust more tempting when parties were faced with increasing social complexity, which did happen quite frequently also in the case of these projects with more stable environments:

**A:** The optimum quality of work was very unclear to us, like when you develop a part of a larger system, you can do it as well and thoroughly as you can, but if others do not follow a similar level of thoroughness, the larger system will not function, and that is source of our fears. And the other fear is that our level might be lower than the level of others, and in that case it will not function either. So we would really like to know more, how big the risk is the parts don’t fit together and fail to function. (Network C, project manager of the customer’s affiliate company.)

**A:** At the bottom there is the difficulty that it is quite difficult to define and document our needs, we do not have proper models for that. And specifications are almost always insufficient, and during the project we do not know at all how eager the people at the subcontractor are to contact us and to ask questions. Like if one guy does not know or understand something, then he might just try to solve it by himself and eventually the work may go totally wrong. (Network A, partnership responsible of the customer.)

**A:** In the early stages there are many training and instructing issues, which need to be taken care of before you actually can get on with the real project tasks. And traditionally we have just posted our internal instructions to subcontractors by e-mail, and every time the results are equally amazing: the first subcontractor outputs are not even close to what our instructions tell. So programmers have internalised some practices of their own so deeply, that it requires quite an effort before they agree to change their mindset and the work matches our expectations. (Network A, liaison person of the customer.)

**4.2.2 Projects in unstable environments**

**Trust – the constructive path.** In the projects with more open environments the relationship between customer and supplier was closer and communication between them took place on a broader level than in the type with stable environments. However, the starting phase of the project did not necessarily require extensive communication and adaptation regarding the specifications and requirements, since they were typically only about to take their shape during the planning phase, and the subcontractor did also contribute quite independently to the specification of the requirements on the basis of some rudimentary ideas. Instead, needs for proper **rule internalisation** and demands of proper **competence** were related strongly to the issues of convergent processes and knowledge about the expected functionalities and technologies:

**A:** The subcontractors are not able to do the development work, unless they do the specifications themselves. When they do the specifications, it indicates that they know what the work really is about. For example the functionality can be specified so that they are standardized, but it does not tell whether the subcontractor actually knows what the content is going to be. (Network D, process developer of the customer.)

**A:** This subcontractor has a development mode very similar to ours, and so the development rhythm becomes very easy, for example such things like when we want to try out various solutions, to test and then again optimize the integration, it does not pose big problems. (Network D, process developer of the customer.)
Relational orientations of openness, reciprocal dialogue and concern did not target to the exchange of already existing specifications but to the lively discussion about emerging development solutions, sharpening specifications and their effects on the schedules and resources of the project. After all, readiness to communicate and inform the other party had to extend to a large group of people across the network, since sharpening specifications had implications to other modules, schedules and resources needed. Since also the subcontractor was an active creator and sender of information, openness and concern were expected at both ends in order to facilitate the coping with high social complexity:

A: When we started we agreed that we would contact only the subcontractor’s process developer and project manager during project work, no direct contacts between the developers, in order to keep a sort of command order there. But once we started to dig deeper into the development project, clarifications and details, it turned out to be a problem that information does not spread fast enough. If somebody was not on-line for half a workday, there could be e-mails waiting in their inboxes that actually were targeted to other persons. And so we changed this communication arrangement, first agreed that this more free communication arrangement is okay to everybody and then implemented the more flexible mode where developers could contact each other. (Network G, project manager of the customer.)

Distrust – the destructive path.

Failure to frame the topics of process convergence and knowledge about required technology with adequate relational orientations (rule internalization, competence) led to serious difficulties in the case of open projects:

A: In the beginning we agreed on a process where a supplier from America delivers to us, but that was one of the biggest mistakes that were done in this project. The customer let them deliver us only once every second month, and we were required to deliver once a week. So once we recognized that something was missing, we had to wait the two months. (...) We complained about that problem to the customer, and finally they made the supplier adopt a similar once a week rhythm, but it took them unreasonably long. (Network E, project manager of the subcontractor.)

Inability to provide the other party with support on the dimensions of openness, concern and reciprocal dialogue also questioned the legitimacy of the relationship and discouraged members when faced with high social complexity typical of this project type:

A: A major problem here has been in communication, since we have all the restrictions right from the beginning. We do not get to see virtually anything, every time we have to apply for permission from the customer when we’d like to see something, from the documents, discussion forums, error reports, which means that we must know in advance what we need before we can ask for that. (Network E, project manager of the subcontractor.)

4.3 EXECUTION

In the execution phase parties often encountered the constraints of their relationship, which stemmed from the improper choices made in the relationship initiation and goal definition phases. In the execution phase the redefinition of the relationship and the reframing from distrusting disposition towards more trusting orientation required significant efforts.

4.3.1 Projects in stable environments

Trust – the constructive path. Since the assumption in this type of a distributed project was that the tasks, methods and expected quality of the development work should be rather well communicated or at least trained in the beginning of the project, developers were not prepared to negotiate with and assist each other extensively once the project execution had started. However, the need for assistance in technological problems, clarification of specifications and feedback of project progress were major sources of social complexity in this type of project. It seemed that there were a lot of issues coming up during the project, which required collaborative solutions and negotiating and could not be perfectly settled beforehand in the planning phase. Hence, a critical improvement on the relationship dimension was achieved by introducing such a role to the project whose responsibilities would include technical
assistance and e.g., clarification of development solutions, documents or test feedback between companies. On the relationship dimension this arrangement can be interpreted as establishing mediating links, consistent role expectations and openness. This arrangement directed the dispersion of knowledge and experiences via some salient key roles, which helped members to decide whom to contact on troubling issues. The roles of the mediating link persons made the flow of communication more predictable, thus decreasing the trust-eroding influences caused by miscommunication. Communication became at the same time more dispersed and more controlled. Furthermore, the development of common history seemed also to focus around these mediating link persons, who had experience of the work with the given partner; these experienced “relationship builders” transferred important knowledge both between groups and within a group when starting a new project:

A: But in this kind of arrangement of subcontracting the problem is that you have to find such longer term, trustworthy contact persons. Q: Do you think that your subcontracting has developed to that direction? A: Yes, definitely. I mean this is a product of evolution, I don’t know how it would be possible otherwise. In the beginning we monitored directly single developers and gave them tasks, because their group was so small, but then the whole thing started to grow and became more dependent on their key persons who took responsibility of the work done there. (Network B, project manager of the customer.)

A: We have noticed how important it is that there is a technical contact person very close to the subcontractor interface, and not a mere bureaucrat who can only pass papers and messages further to other persons. (...) The technical contact person can effectively reduce the workload generated by subcontracting. He can filter out certain questions and answer himself to a majority of the questions. Or he can buffer, search answers to many more questions, really understand the issues and then provide direct support. (Network A, project manager of the customer.)

The need for feedback of project progress and quality of own work was another significant source of social complexity, because in this type of project the processes of the organizations were typically fairly loosely connected and the systems of electronic communication not fully integrated. Thus, especially the supplier’s awareness about project progress and sense of the optimal quality were strongly dependent on the customer’s willingness – concern – to provide sufficient feedback and thus establish openness. This feedback seemed to be particularly important, because in the cases analysed also shared commitment, category membership and sense of common history were closely intertwined with the perceived displays of openness and concern. A partner who at least tried to keep the other party up to date and facilitate its adaptation to the development routine was perceived as trustworthy and worth of the efforts required to provide facilitating arrangements in return:

A: After the visit to our subcontractor, where we clarified the progress, development phases and expectations concerning the project, they have now equipped everybody with e-mail connections and we know the e-mail addresses and messenger addresses of each person, and thus the amount of communication has increased significantly. (Network A, liaison person of the customer.)

Distrust – the destructive path. Inability to differentiate and specify the relations between central technical and inter-organizational link-roles prevented timely dispersion of crucial information and experiences, at worst totally preventing the accumulation of common history and blocking the feedback between separate groups. In the cases where functional roles for mediating links were not formed, some technically or organizationally salient persons typically encountered overwhelming and conflicting communication pressures, which discouraged the parties involved in the frustrating communication:

A: In this project my role has turned out to be more or less a general contact person, because I have worked the longest in this whole group. Maybe I have deepest knowledge of the technical environment, and due to that reason I spend practically half of my working time speaking on the phone and writing e-mails. (...) The phone rings about 40-50 times a day, and at the same time I should carry out large programming tasks, write thousands of lines of code, and that gets really difficult, when you always have to jump between these tasks, not getting any time to concentrate. (...)
And my jobs have just slowly slipped towards a general contact person’s role, mainly because I happen to have gained most experience here. But, at times it gets really tough and difficult. (Network C, liaison person of the customer.)

It turned out that remote teams were often left without any proper guidance when they bumped into uncertainties or open questions, e.g., regarding expected development solutions or quality of work. In three of these four cases studied a prolongation of this kind of an uncertain situation resulted in mutual prejudices: suppliers felt that the customer was not really concerned about them nor interested in facilitating their work, whereas people at the customer felt that the supplier is purposely doing low quality work or tries to play games with them. This kind of experienced lack of concern prevented the formation of a common identity at the network level and deepened the experienced gaps between separate organizations. Lacking knowledge of the interaction styles of members at the other sites made it even more difficult to overcome the gaps opened by ethnocentrism:

A: At times it feels like a black whole surrounds us, we do something and send it somewhere, it goes somewhere but nobody knows where and what happens then. Then comes another job and nobody has time to think, what happened to the previous one and how it succeeded. (...) Honestly I feel that I get feedback from the customer’s project manager only when I have done something wrong. (...) In certain phase the project did so poorly that everybody wanted to quit it. But then we decided that let’s try one last time, and maybe customer’s side was even more willing to quit at that time, but also our side thought that it is not gonna work anymore. And maybe if customer would now say that they want to quit, our people might feel that just go ahead and quit. (...) And sometimes I don’t understand whether the sender is angry or whether he just has a certain style to express things, whether he behaves like that with everybody or only with me. (Network C, project manager of the customer’s affiliate company.)

4.3.2 Projects in unstable environments

Trust – the constructive path. Vague specifications and frequent changes – even sudden needs to cancel the development of certain features – implied, that the members of the network could not adhere to rigid development norms and rules. Instead the development teams had to show readiness to continuously evaluate the future directions of development work and collaboratively negotiate about the emerging dependencies to other modules. Since the specifications of the modules were evolving during the project and the changes in other modules had to be taken into consideration also in one’s own development work, developers had to be aware of the project progress in the whole network. The interaction concerning specifications, interfaces and development schedules had to be framed with extensive openness, reliability and reciprocal dialogue – a main reason why open communication across developers and managers was strongly supported in this type of projects. Much effort was made to improve the readiness to solve sudden interface-problems with remote teams: e-mails, mailing lists, phone, chat, or collocated visits were used depending on the nature of the communication need in question. Often, e.g., e-mails were not sufficient to create proper reciprocal dialogue and hence techniques like chat or collocated meetings were preferred. When development processes were integrated to a large extent and information dispersion was fluent, identification to a common goal (shared category membership) came quite naturally, since the members from remote sites could perceive fairly clearly how their contributions were connected to the larger project and affected its concrete progress. The possibility to concretely follow the accumulation of separate innovative contributions to a larger whole seemed to be quite a strong uniting force and characteristic of this type of projects:

Q: Were you able to give feedback to this team with whom you had common interfaces, if you noticed that ought to correct something? A: Yes, we had a common error database. In the beginning they had a separate database though, but pleasingly early they recognized that maintenance of a separate database is not going to work. Then they started to put their errors also for us to see in a common database. (Network E, project manager of the subcontractor.)

A: I have learned unbelievably much during this project, so for me this has been a great experience. Really from the perspective of our company, since we survived from a shaking journey like that and reached the goal, we have now
about fifteen top professionals in our company for the future projects. This was very useful for the company, also on other criteria than money. (Network E, project manager of the subcontractor.)

**Distrust – the destructive path.** Reliable access to emerging new plans, schedules and specifications were critical for all parties in the network. If the information needs and requests from other parties were ignored, accusations of information hiding resulted at worst in the accentuation of organizational boundaries and development of separate antagonistic identities. Another barrier preventing suppliers from accomplishing adequate organizational adaptations was the information block caused by uncertainty that people at the customer had concerning the scope of nondisclosure agreements and confidential information. When in doubt, the customer’s developers preferred to withhold all information, which in turn raised suppliers’ prejudices and made the reduction of social complexity in terms of trust improbable:

A: One problem concerning communication and company interfaces is that at times our people are uncertain about what they are allowed to tell the subcontractor’s people. Subcontractor should of course have access to certain information, but on the other hand we don’t want to reveal them everything, and especially if this subcontracting issue is new to some of our teams, they might be hesitant. We don’t want that the subcontractor feels like an outsider, but some issues have to be kept firmly inside the house. This is a problem. (Network D, partnership responsible of the customer.)

A: There are lots of interdependencies between us. And many times it has happened that someone who is dependent on our module has written an error report, which we have not seen for a month. And our answers in turn have not reached them for months. And again begins the wondering what’s the problem. (Network E, project manager of the subcontractor.)

Commitment to a common goal suffered also, if the roles and responsibilities of the members at other sites were not clear to the participants (vague role expectations). Then the required extensive communication became chaotic and unpredictable, effectively undermining the members’ possibilities to show that they anyway try to be cooperative:

Q: What are the worst problems? A: Communication problems, for example internal weekly reports, I always have to phone to check and ask all the details that I need to know. The basic issues are on such a general level in the report that I have to check all details. Another problem is ignorance of the information that has been exchanged between different parties, for example between customer and our remote site, and I do not necessarily know about it at all. Or vice versa, they do not know about our communication. (Network E, project manager of the subcontractor.)

**4.4 DELIVERY**

Delivery phase was a critical turning point in the definition of the relationship, because at least in this phase the possible deficiencies of other phases became visible: projects in stable environments recognized the existence of insufficiently internalized development methods and quality requirements, whereas projects in unstable environments recognized possible discrepancies in the synchronization of processes and delivery cycles. Both recognitions were powerful enough to turn a trusting orientation into a distrusting one.

**4.4.1 Projects in stable environments**

Trust – the constructive path. The interaction related to testing the deliverables entailed mutual adaptation needs – i.e. social complexity – concerning sufficient quality and correction of errors in the code. When testing the deliverables was done on a different site than the actual development work, it became crucial to provide the sites with sufficient background knowledge concerning on the one hand broader, contextual logic related to the deliveries and on the other hand testing procedures and identification of errors. This kind of transparency in background knowledge was successfully achieved by mediating link persons or collocated visits to the partner’s site. These practices offered the members a deeper understanding of the underlying working logics and the sources of encountered
problems, reconstructing common rule internalization. Thus groundless accusations of incompetence could be avoided and confidence in the competence and concern of the other party could be maintained. The mediating link persons also succeeded in building feeling of a common identity (category membership) and providing visibility of project progress:

A: Our big problem has been that subcontractors have developers whose capabilities vary so much, and also the capabilities of our testers have varied much. Therefore I have started to do some initial check ups on the work, making sure that the delivery functions correctly, and I read some code in order to find out whether it is even close to the criteria for testing. If the criteria are not met, I give immediate feedback to the subcontractor about the issues demanding further improvement. And when the testing reports arrive from our people, I do some comparing and try to make the testing reports from different testers uniform. (Network A, liaison person of the customer.)

A: This subcontracting depends much on the persons and that the team there functions properly. I do not think that such things could succeed, where some part of a new product is taken out and just given to a subcontractor, instead the subcontractor team should somehow grow up and evolve. It must be very hard to start without some background. (Network B, project manager of the customer.)

Distrust – the destructive path. When quality of the deliveries did not meet the expectations, competence of the supplier was typically questioned; in some occasions the developers and testers lost their interests to cooperate with a supplier whose first deliveries seemed strange and did not meet quality requirements. If the confidence in the competence of another party was lost, the basis for a trusting relationship was very weak:

A: Eventually it started to happen that our acceptance testing lasted for ages, and there emerged this attitude that since the subcontractor sends so poor deliveries then we can’t be bothered to test. Somehow like the subcontractor would be an enemy. (Network A, project manager of the customer.)

A: It was really difficult since there was no real ownership at the subcontractor’s site, there were no clear key persons, there were only workers, developers, designers. Everything that came out from there had to be checked out by our people and it was really ineffective, I didn’t like it at all. (Network B, project manager of the customer.)

A closely connected obstacle to the confident relationship building was also identified: since transparency on the level of the whole project seemed to be very difficult to achieve, some remote teams felt that without concerning link persons they were not at all aware, whether their quality of work had really been suitable for the project or how their contributions finally affected the progress of the project. Consequently these suppliers did not have possibilities to meaningfully adapt their own working to the requirements of the network, thus undermining possibilities of both shared rule internalization and shared category membership. Without these supporting relational resources the reduction of social complexity was likely to take the form of distrust instead of trust:

A: We can do the tasks but we do not see whether they really work. Okay, the code works and you do not get error messages, so then we can say that this has been tested and it should work. But when you do not see the whole package, you do not know the final result and you get frustrated, you do not know whether it after all functioned or not. (Network C, project manager of the customer’s affiliate company.)

4.4.2 Projects in unstable environments

Trust – the constructive path. In the projects with more open environments the synchronization of the deliveries (e.g. software builds, test reports) seemed to be highly critical for the functioning of the whole network. Since the project work included much uncertainty with innovative and risky technological solutions, the project required a frequent and reliable delivery rhythm. This was a way to achieve better reaction capabilities across the whole network; loose or unreliable deliveries tended to cause immediate problems in the development work and frustrate the working and visibility of many developers. A reliable delivery cycle, which was well internalized by the members already early in the project, functioned as the pulse of the whole project and once obstacles or delays disrupted the rhythm, the whole development work tended to lose predictability and controllability. Frequent deliveries were also sources of rule internalization across the network in another sense, since in the deliveries the
development logic of other sites was concretely materialized, which facilitated common understanding between parties. In this sense the delivery phase was clearly an area, where the companies accomplished rapid mutual adaptations based on each other’s communicative messages (the deliveries):

A: The subcontractor’s mode of working is similar to ours. They did an increment once a week, they could tell what functionalities they have developed every week, and the error level in that functionality. So we could very clearly follow and keep up-to-date with their working. And typically in these projects the functionality gets ready, but it’s not enough since then we need to start tuning it up. And that is the reason why it is very important for us to see, where the subcontractor is going and receive some forecasts indicating whether they can optimize their module to the required level. (Network D, process developer of the customer.)

Distrust – the destructive path. In cases where the reliability of the delivery cycle was broken, definition of the relationship tended inevitably to slide towards distrust. Other relational resources simply did not manage to compensate for the lack of reliability and rule internalization in case of deliveries.

A: In the beginning the customer’s remote site delivered us bi-monthly, and we were required to deliver once a week. When we noticed that something was missing, we had to wait for the two months. (...) Our situation was really chaotic, because we were asked to give schedules about the readiness of certain functionalities, and we had dozens of errors that were dependent on the modules of this slow delivering site. And we could not know whether they tried to correct some errors and what kind of side effects the corrections would cause. Our schedules were totally destroyed, and we received all the complaints from the customer, since we are subcontractors. (Network E, project manager of the subcontractor.)

5 DISCUSSION AND CONCLUSIONS

The findings from complex inter-organizational software development projects seem to support Luhmann’s idea about trust and distrust as functionally equivalent but opposing strategies to reduce social complexity. The parties of an inter-organizational software development network may be well advised either to avoid the higher degrees of social complexity by stabilizing the environment of the development project or to promote the building of trust when stabilization of the environment is not possible. In inter-organizational networks with salient group boundaries – the focus of this paper –, increasing social complexity tends to create difficulties, because members cannot rely on traditional organizational bases of trust in their interactions. Hence the parties have to somehow compensate for the absent sources of trust by supportive practices in order to make the development of trust more probable. However, the establishment of compensating relational resources seems to require significant supportive arrangements from the part of the member organizations of the network.

For the part of projects in stable environments, the early phases of the project seemed to be critical for the definition of the relationship, since communicative activity settled somewhat as the project progressed and did not provide such good opportunities to deepen the relationship in further interactions. The most decisive relational resources in building trust seemed to be the early framing of the interaction by establishing mediating link persons and providing them with salient role expectations. The mediating link persons could function as buffers, compensating for the lack of some other important relational resources in the later phases. Effectively functioning connections mediated by link persons could buffer the destructive effects caused by, e.g., occasional problems in work quality, insufficiently internalized development methods or insufficient feedback.

For the part of projects in unstable environments, the focus was strongly on active communication during the whole project and internalization of standardized development rules could be utilized to a much lesser extent. However, the most decisive relational resources of trust were related to
predictability, namely reliability and rule internalization concerning the frequent delivery cycle and synchronization of the processes. Here the distinctive feature was that trust building seemed to depend on the predictability of the manner of working, not on the content of working. When this framework of development was reliable and established openness between parties, deficiencies in other relational resources could typically be compensated for. In occasions where the delivery cycle generated unpredictable consequences, the basis for trust in the midst of increasing social complexity began to vanish and the relationship between parties turned towards distrust. To summarise the results of the analysis, the activated content and relationship categories in different project phases for both project types are presented in Table 3.

Table 3 Activated content and relationship categories in different project phases in two project types

<table>
<thead>
<tr>
<th>Projects in stable environments</th>
<th>Projects in unstable environments</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Content category</strong></td>
<td><strong>Relationship category</strong></td>
</tr>
<tr>
<td>Assignment</td>
<td>Category membership, concern</td>
</tr>
<tr>
<td>Skills, process, roles, training</td>
<td>Openness, reciprocal dialogue, rule internalization, mediating links</td>
</tr>
<tr>
<td>Specifications, quality, coding standards, skills, process</td>
<td>Openness, reciprocal dialogue, rule internalization, mediating links, role expectations interaction orientations, common history</td>
</tr>
<tr>
<td>Development solutions, specifications, technology, interfaces</td>
<td>Mediating links, role expectations, openness, common history</td>
</tr>
<tr>
<td>Project progress, quality</td>
<td>Openness, concern, common history, category membership, interaction orientations</td>
</tr>
<tr>
<td>Quality, errors, project progress</td>
<td>Competence, concern, category membership, rule internalization, mediating links</td>
</tr>
</tbody>
</table>

Although the presented results are only preliminary, striking features of the analysis were the strong impacts of stability of the project environment and nature of development process on conditions for trust development. On the one hand discrepancies in the quality of work were surprisingly strong sources of distrust in projects in stable environments and fairly difficult to overcome without concerning mediating links, who could help in building shared identity and common history. On the other hand better visibility to the contributions of other parties and frequent negotiations about evolving development solutions helped to build common understanding and prevent sudden lapses into distrust in the case of projects in unstable environments. This suggests that the work process itself may contribute significantly to the building of trust in networked conditions, e.g., by preventing or facilitating visibility. Further, the approach, where the antecedents of trust are examined as possible relational resources in interaction instead of fixed necessary and sufficient conditions, might prove to be useful in the analysis of empirical, textual data. After all, a major challenge in trust research has
been to single out the critical antecedents of trust in specific contexts, and by concentrating on both adaptation demands and relational resources, the multiple context specific dimensions determining trust and distrust could perhaps be traced better.

When assessing the generalisability of the results, it seems that the applied content categories were not that software specific that similar tendencies in trust and distrust could be found also in other kinds of work-oriented networks. On the other hand the categories and the relations between them need still further clarification in order to provide a more precise picture of the dynamics of trust and distrust in networks with varying degrees of stability. This clarification of the categories and collection of rich data from stable and unstable projects will be the research objectives in near future.

REFERENCES


