UNIVERSITY OF OSLO
FACULTY OF EDUCATION

Pål Fugelli

Thesis submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy (PhD) awarded by the University of Oslo 2010

INTERSUBJECTIVITY AND OBJECTS OF KNOWLEDGE:

Making sense across sites in software development

Series of dissertations submitted to the
Faculty of Education, University of Oslo
No. 128

ISSN 1501-8962

Cover: Inger Sandved Anfinsen.
Printed in Norway: AiT e-dit AS, Oslo, 2010.

Produced in co-operation with Unipub AS.
The thesis is produced by Unipub AS merely in connection with the
thesis defence. Kindly direct all inquiries regarding the thesis to the copyright
holder or the unit which grants the doctorate.

Unipub AS is owned by
The University Foundation for Student Life (SiO)

Acknowledgements

I would like to gratefully acknowledge the reliable supervision of Prof. Leif C. Lahn. I thank Dr. Anders I. Mørch for enthusiastically stepping in as a co-supervisor from 01.01.2008. My colleagues in the FALK research group are thanked for stimulating discussions and comments on the conceptual approach adopted in this thesis. Furthermore, I would like to express gratitude to my colleagues in the research project 'Professional Learning in a Changing Society' (ProLearn) for their general comments and advice. In particular, I acknowledge Prof. Karen Jensen for her invaluable inspiration and kind help, and Dr. Monica Nerland for our encouraging collaboration when organising and performing joint tasks in the ProLearn project, as well as her insightful remarks as this thesis progressed.

I am also indebted to Prof. Sten R. Ludvigsen and my colleagues from InterMedia, University of Oslo for introducing me to the research community throughout the Master's thesis period, and for encouraging me to continue as a researcher.

Finally, I am eternally grateful for my supportive family.

Pål Fugelli

Blindern, 23.03.2010

Table of Contents

List of figures

List of tables

Screen images

1 Introduction

The potential for new forms of qualitative interaction and communication in a networked knowledge society implies a "reconfigured" view of the objects and sites for knowledge production. In some respect, new information structures are changing the manner in which we socialise, interact, and learn (see e.g. Brown & Duguid, 2002). Participation around shared objects in distributed, networked communities can perhaps generate opportunities for new forms of (and constraints on) intersubjectivity to develop; for example, to render online communities more visible. The introduction of new networked cultural tools may challenge classical theories of intersubjectivity as we know them today. For instance, the Norwegian social psychologist Ragnar Rommetveit's (1977, 1979) framework for intersubjectivity is mainly concerned with how a partly shared understanding is achieved in everyday face-to-face interaction. Perhaps his notion of intersubjectivity can be adapted and developed further for studying interaction in a computer-mediated environment. I will suggest that distributed arenas for software development provide a rich context for exploring this theme. The networked knowledge practices of software development have attracted attention in recent literature on social theory as "an embodiment of the spirit of the times" (Hernes & Czarniawska-Joerges, 2005:12). Commons-based peer production allows participants to freely interact with the resources that are jointly produced in networked environments. According to Benkler (2006), these resources are produced outside the conventional proprietary system "in a framework of social relations" (p. 62). Moreover, in these distributed scenarios, social and affective dimensions of shared knowledge are often explored as collective acts that are not reducible to individual cognition. However, little is currently known about how the objects jointly attended to in software development contribute to maintaining social relations across contexts and 'boost' processes of intersubjectivity.

The notion of objects has recently gained new attention in social science, philosophy and educational research (see e.g. Engeström, Puonti, & Seppänen, 2003; Knorr Cetina, 2001a; Law & Singleton, 2003). In this literature, object relations appear in a variety of forms. For example, object-orientedness or the object of activity is regarded as a core concept in the framework of activity theory (Kaptelinin, 2005). According to Nardi (2005), a common understanding in this framework is that "an object orients attention and actions to fulfil a motive" (p. 39). The object thus gives an activity its meaning and direction. However, it is argued that collaborative systems often fail to capture the passions and desire behind object

formations. Accordingly, such 'emotional' dimensions must be considered if we are to have a fully flavoured theory of collaborative object-oriented activity. The focus on object relations is also a key concern in the new sociology of knowledge. In this tradition, Knorr Cetina (1997, 2001a, 2004; 2002) put forward the idea of *epistemic objects* as characterised by a changing and unfolding nature and the lack of completeness. In expert cultures, it is assumed that such open-ended epistemic objects have an important binding role and the potential to generate new forms of sociality and wanting structures. An example can be open-source software developers jointly working on a global project in which the source codes act as knowledge objects. Numerous skilled programmers scattered around the globe work jointly on an advanced object via internet technology. The unfinished, continually evolving software programmes towards which they orient imply a commitment to an evolutionary process or practice. Novel bonds may form in such communities as material or semantic objects attain social roles. This does not imply a common form of socialisation whereby people meet in a society or organisational context and form relations. Engaging with distributed objects of knowledge over time, however, can perhaps create a form of relation that goes beyond our everyday notion of what is social. I will suggest that these distributed scenarios represent unique cases to explore *how intersubjectivity evolves around shared objects of development*.

This thesis is dedicated to providing an account of a few key dimensions in the knowledge practices of software development. Particular attention is granted to knowledge processes in relation to mediated expert networks. I embark on providing a contribution to the understanding of software developers' knowledge practices; initially by exploring the knowledge resources applied by software engineers as part of the ProLearn project, and subsequently by investigating the processes of intersubjectivity in an open-source development project. The research aims for this thesis are as follows:

Research aims

1. To explore how key artefacts applied as knowledge resources mediate relational aspects associated with knowledge objects in software development.
2. To describe the overall network structure of a distributed open-source development project, and identify central and peripheral sub-groups of participants.
3. To analyse how processes of intersubjectivity evolve around shared objects of development in the mediated communication among participants in the sub-groups**.**

Table 1: Thesis research aims

In the formulation of thesis aims above, the verbs *explore, describe* and *analyse* are intentionally applied to reflect different stages in the research process. This indicates a progression from initial exploration and thin descriptions of software developers' knowledge practices, to higher levels of abstraction when analysing object relations and processes of intersubjectivity. In order to provide a logical structure for the inquiry, a few issues based on these initial aims are further developed throughout the thesis, leading to more precise research questions as we approach the empirical section.

In order to achieve the overall aims, multiple data sources are purposely selected from two different developer scenarios; a) professional software engineering in a workplace context, b) quotation practices in a mediated open-source development project. Furthermore, I develop and suggest a conceptual framework for analysis, drawing on literature that is of explanatory relevance for the aims stated above. In the last decade, there have been numerous empirical studies targeting the innovative dynamics and organising features of open-source communities (see e.g. Lanzara & Morner, 2005; Lee & Cole, 2003). However, conceptual frameworks for studying relational aspects of knowledge objects and processes of

intersubjectivity in the context of open-source software development are not extensively developed in the research literature. Drawing on empirical examples, this thesis is also committed to providing a theoretical contribution. By combing and re-contextualising a set of interrelated definitions from different traditions, I suggest and try out a conceptual framework for analysing relational aspects of shared knowing in technology-mediated networks. In this way, the empirical studies are partly theoretically driven. They also serve as a breeding ground for developing a new theoretical framework in order to increase our understanding of phenomena that might also have a wider application in a networked knowledge society.

1.1 Background and relation to the ProLearn Project

In the following, I briefly introduce the ProLearn research project and my involvement in it. The project ran from 01.02.2004 to 01.02.2008 as a part of the Programme for Knowledge Education and Learning (KUL), funded by the Research Council of Norway. ProLearn was lead by Prof. Karen Jensen and organised as a collective effort between researchers from the Institute for Educational Research, University of Oslo and the Centre for the Studies of Professions, Oslo University College. Project organisation consisted of 7 core members and 3 associated members. The overall objectives stated in the project application are:

> first to expand theories on professional learning in a direction that is sensitive to the rapidly changing institutional and cultural contexts typical of today's society. Secondly to make comparative studies of learning trajectories among teachers, nurses, engineers and accountants, emphasising the transition from higher education into working life (Karen Jensen & Lahn, 2003:1).

Departing from these objectives, participating researchers have studied the interfaces between young professionals' practice, their learning strategies and orientation towards knowledge in the light of a changing knowledge society (Karen Jensen, et al., 2008). Special attention has been granted to the challenges that shifts in the structure of knowledge associated with a post-industrial knowledge society represent for the targeted professional groups. In order to explore how young professionals form binding relations with their knowledge, the techno-cultural perspective of Knorr Cetina served as a common ground and inspiration during the project (see e.g. Knorr Cetina, 1997). The underlying idea is that object-centred, distributed knowledge cultures integrate minds and activities of individuals that are not co-present,

forming new kinds of social relations (Karen Jensen & Lahn, 2004; Knorr Cetina, 1997). The set of assumptions or postulates underlying the formation of object relations in this perspective is treated in further detail in chapter 2.3.1.

Now, let us clarify my role as research fellow in the project. First, the project proposal for this thesis was specifically adjusted to the central research problems and conceptual framework of the ProLearn project (targeting object relations). Furthermore, my PhD scholarship at the Faculty of Education, University of Oslo incorporated 25% contractual obligation 30% of this was earmarked for tasks in the ProLearn project, whereas 70% was delegated to teaching obligations. Throughout the project's lifespan, I have been involved in a wide range of tasks which are highly relevant to this thesis. These include project planning and organisation, joint data collection, preparation and structuring of data for analysis, participation in analysis workshops, attendance at conferences and contribution to the project reports. In a retrospective light, it is reasonable to acknowledge that my participation in the project has influenced the research themes and issues developed in this thesis. I have benefited greatly from collective efforts in data collection and the peer review processes during project meetings. The joint data collection strategies and the additional data gathered specifically for the purposes of this thesis are further described in chapters 5 and 6.

Data collection in the ProLearn project progressed in two major phases; the first in 2005 and the second in 2006. Decisions concerning data collection strategies had to account for intersecting interests of individual researchers as well as the unifying problems and theoretical framework of the overall project. As I write this, the ProLearn project is complete and the results are disseminated at research conferences, popular science reports and various academic publications[1].

1.2 Thesis structure

The chapters are organised around two case studies, knitted together by the overall aim of this thesis and its theoretical framework. In the following paragraphs, the two scenarios representing the core data are briefly explained. The first case embarks on mapping and

[1] See the project's website for a full list of publications (http://www.pfi.uio.no/prolearn/).

describing software engineers' knowledge resources and exploring how key artefacts mediate relations to temporally and spatially distant knowledge objects. This research is conducted as an integrated part of the ProLearn project. Primary data sources are in-depth interviews, focus groups, learning logs and documents. The interviews and logs cover topics such as 'triggers' for knowledge searches, applied sources and participation patterns in local or distributed arenas. A typical software development environment is complex, constantly changing and partially distributed. Under these circumstances, it is difficult to identify and study isolated variables. Due to the tentative nature of the initial research aim of this thesis, explorative case study research is applied in the first phase of the inquiry. According to Yin (1994), in an explorative case study, data may be accumulated prior to the final definition of research questions. This study does not rely solely on existing theory, but aims to discover qualities of the phenomena studied by approaching the field partially inductively. The investigation is also regarded as explorative as it opens the way for new theoretical perspectives on artefacts and object relations.

The second case focuses on the role of intersubjectivity around shared objects in open-source software development. To interact and work on a common set of objects in a distributed software development community, it can be argued that the participants must share some categories of understanding. Rommetveit (1974; 1985) offers illuminating insight on how joint activity is made sense of by participants who are concerned about a matter, yet interested in it in their own particular way. Rommetveit (1974) argues that every communicative act builds upon the commitment to "a temporarily shared social world" (p. 29). The meaning emerges in an interaction and is considered mutually constructed and grounded in the social realities that participants bring to the situation, thus forming a partly shared social world. However, Rommetveit's approach is mainly concerned with how intersubjectivity is achieved in face-to-face interaction. It can be argued that his framework is not adequately developed for studying interaction in a mediated, technologically-rich environment. The introduction of new cultural tools implies new challenges to the theory as we know it today. Interaction patterns among software developers are not only co-located, but also mediated through advanced technology. How will processes of intersubjectivity around joint objects unfold in this particular mode of interaction? With the introduction of new mediating tools (i.e., a virtual arena for joint software development), it is reasonable to assume that the processes of achieving shared understandings around joint objects have been qualitatively transformed.

The process of reaching partial intersubjectivity in this context has not yet been explored in depth.

In order to peruse the research aims stated above, the thesis starts out in chapter 2 by discussing relevant concepts for analysis. Subsequently, a conceptual approach is adopted in chapter 3 to inform empirical studies in chapters 5 and 6.

An overview of the thesis' interrelated parts follows below.

Chapters	**What**	**How**	**Why**
Chapter 2-3	Conceptual framework	Review relevant literature on objects and intersubjectivity. Examine how concepts and ideas from different disciplines can be combined and re-contextualised for the purposes of this thesis.	Develop theory and put forward a conceptual framework to shed light on empirical analysis in chapters 5 and 6.
Chapter 4	Overall research design	A discussion of the overall research design and practical matters of data collection.	Describe the 'meta design' in common for the two thematically related studies in chapters 5 and 6. Detailed explanation of how the research project is organised to pursue the aims.
Chapter 5	Exploring artefact-mediated relations	Explorative case study drawing on multiple	Explore and understand how artefacts applied as

	towards objects of knowledge shared in distributed expert networks.	sources of evidence collected from one professional software developer.	knowledge resources in a local work context mediate relations to objects of knowledge in distributed expert networks.
Chapter 6	Processes of Intersubjectivity in open-source development networks.	Case study of an open-source development project. A social network approach is applied to structure the data corpus for a more detailed content analysis targeting processes of intersubjectivity.	Contribute to the understanding of how intersubjectivity evolves in distributed expert networks. Add to present theory on these issues.
Chapter 7	Concluding discussion	Draw together the main threads of argumentation, explain findings and examine whether the aims have been achieved. Assess the thesis' original contribution and limitations.	Summarise key points made in the data presentations and link the argumentation into a coherent body of understanding.

Table 2: Overview of thesis' interrelated parts

2 Conceptual framework

This chapter deals with the relevant ideas and concepts for the subject matter of this thesis. The overall aims stated in the introduction imply notions of practice, artefacts, object relations, networks and intersubjectivity. The discussion that follows is intended as a primer for adopting a conceptual framework for analysis; it seeks to construct a basis to inform the research comprising the empirical section of this thesis. As a part of this work, I examine how concepts and ideas from different disciplines can combine, and perhaps re-shape, to form an analytic framework for studying knowledge practices and distributed processes of intersubjectivity. I thus find it reasonable to focus primarily on research and academic writing that is related to the research aims of this thesis. For readability and transparency, literature that concerns methodology is treated in dedicated chapters as we approach the data.

2.1 Intersubjectivity

Intersubjectivity is a multilayered, interdisciplinary concept with no clear and united definition. In the book *Intersubjectivity: The fabric of social becoming*, Nick Crossley (1996) attempts to unpack the different dimensions of the concept. Emerging from this discussion, there are a few relational aspects of an 'intersubjectivity space' that may inform software programming; namely *participation*, *language* and the *desire for recognition*. For example, sharing ideas and problems among developers is partly made possible by using common programming environments and language for speech. Furthermore, being recognised by others may be of particular relevance for contributions in open-source development projects. By externalising problem-solving ideas and design into programme code, open-source developers are afforded the opportunity for mutual recognition in their respective communities. The desire for intersubjective recognition is thus potentially achieved through relational objects. Yet, this desire is not peculiar to open-source software developers. It has also been suggested that recognition by others is essential to well-being and apt social agency more generally (Blackburn, 1996; Crossley, 1996; Honneth, 1995).

Processes of intersubjectivity are often associated with achieving shared understandings; helping us to "relate one situation to another" (Bober & Dennen, 2001:241). They usually refer to relations between people, rather than individual subjectivities. Furthermore, states of partially overlapping individual subjectivities are often coupled with intersubjectivity. To my

understanding, the 'inter' of the concept intersubjectivity mostly refers to relations between living people (or subjects). However, the social relations between humans and advanced non-human objects suggested by contemporary object-oriented theorists (see e.g. Knorr Cetina, 1997; Lash, 2001; Law & Singleton, 2003) are not accounted for in the current research literature on intersubjectivity and technology. Let us explore the analytic potential of Rommetveit's framework, and in the subsequent chapter, how joint purposes and the medium of communication can influence processes of intersubjectivity.

2.1.1 Rommetveit's socio-cognitive approach

Rommetveit is recognised for the conceptual and theoretical framework labelled the 'socio-cognitive' approach to language, thought and communication (Blakar, 1979). Within it lies a dialogical alternative to the cognitive tradition and an attempt to construct a dialogical-based paradigm in research on human cognition and communication. Rommetveit (1992) maintains that "such a paradigm represents a much-needed constructive alterative to representational-computational models within mainstream individual cognitive psychology and cognitive science" (p. 19). However, in this research tradition, the term paradigm may appear ambiguous. Claims for sudden fundamental shifts in the assumptions underlying a multi-disciplinary research field seem grand and problematic. I suppose it is more appropriate to state that Rommetveit's work inclines towards a *dialogical position*, or as Linell (2003) puts it, "a struggle for meaning, and for explaining what meaning is, in a dialogical world" (p. 228).

According to Wold (1992), approaches to language and meaning are often dichotomised as either dialogical or monological. Dialogists typically maintain that sense and meaning are constituents of interactive dialogues rather than single monologues. Real-life utterances are argued to be primarily 'dialogic', intimately embedded in contexts of dialogue; although Rommetveit "objects to the idea of a language (and culture) as being embedded in a homogeneous collectivity and as being fully socially shared" (Linell, 2003:221). According to this point of view, our social world can only be partially shared and known in fragments. By joining dialogical interactions, we temporarily establish shared understandings of utterances in a sociocultural context. Every communicative act builds upon the commitment to "a temporarily shared social world" (Rommetveit, 1974:29). The social distribution of knowledge and understanding is articulated using "metaphors of shareholding and co-

authorship pertaining to language and situated meaning making" (Linell, 2003:221). It follows that knowledge and understandings are asynchronous, unevenly distributed among 'shareholders' in verbal communication. To illustrate, I adapted and re-contextualised Rommetveit's (2003:214-215) 'carburettor story':

> My friend is an experienced java programmer. In his spare time, he works for a non-profit organisation, developing an application for collecting donations online (money transfers). He applies open-source code gathered from the Internet as 'raw material' for programming his applications. One day, while he is out travelling with his laptop, he discovers a serious 'bug' in the core of the shared code. This is a serious flaw that has to be urgently resolved. Due to lack of Internet access, he calls his wife, who knows very little about computers and programming and who does not have a clue what a software bug is, and asks her to send an e-mail to the respective community of developers to report the bug. She writes a message informing the group of developers that "there is apparently a major bug on line 389 in the code." This action saved the programmers a lot of time and effort in 'de-bugging' the problem.

In the above example, the meaning of the word 'bug' need not be fully shared to have a function in the intersubjective space. The utterance is effectively passed on, mediated by his wife, and serves the practical purpose of telling co-developers about a serious bug and where to locate it. The word 'bug' is here uttered with reference to a software error, a failure in the source core preventing the application from executing specific commands correctly. Thus, as Linell (2003) suggests, words might "function by simply being transferred between language users across situations" (p. 221). However, sense and meaning can be relative to numerous context factors and are not simply transferred between individuals, conveniently wrapped in words. For instance, among software programmers, the word 'bug' first acquires meaning at some level of joint understanding when it gets interpreted by knowledgeable co-developers. Extracted from this particular intersubjective space or context, the meaning potential of the word could, for example, connote an infection caused by a microorganism from an insect (e.g., I got a bug when I travelled to Central America last year). In line with a sociocultural perspective, situational meaning can be seen as partly grounded in cultural resources established prior to the current situation. Foundational dialogical constructs are thus already grounded in what we might call 'sediments for intersubjectivity'. Linell (2003) refers to such processes as "double dialogicality, one pertaining to sociocultural traditions and one

pertaining to situated interactions", where "knowledge about worlds and languages has been shaped and appropriated in actors' biographical experiences of prior situations and by previous generations in sociocultural history" (p. 226).

Rommetveit's conceptual framework on intersubjectivity represents a rather pluralistic approach to language and communication. The "architecture of intersubjectivity" presented by Rommetveit (1979a) is inspired by and closely related to the works of Mead (1962), Vygotsky (1962, 1978), Merleau-Ponty (2002), Wittgenstein (1968; 1969) and others. An important lexis is that, "communication aims at transcendence of the 'private' worlds of the participants" (Rommetveit, 1979a:94). 'In-situ', transcendent perspectives can accordingly be only partial, never complete. Utterances produced here and now are considered grounded in a pre-supposed shared reality. Human dialogue is thus not to be seen as exclusively private or social, but transpires from the partially shared social reality. Between the 'I' and 'You' in Figure 1 below, potential states of intersubjectivity can emerge along spatial-temporal coordinates in three dimensions.

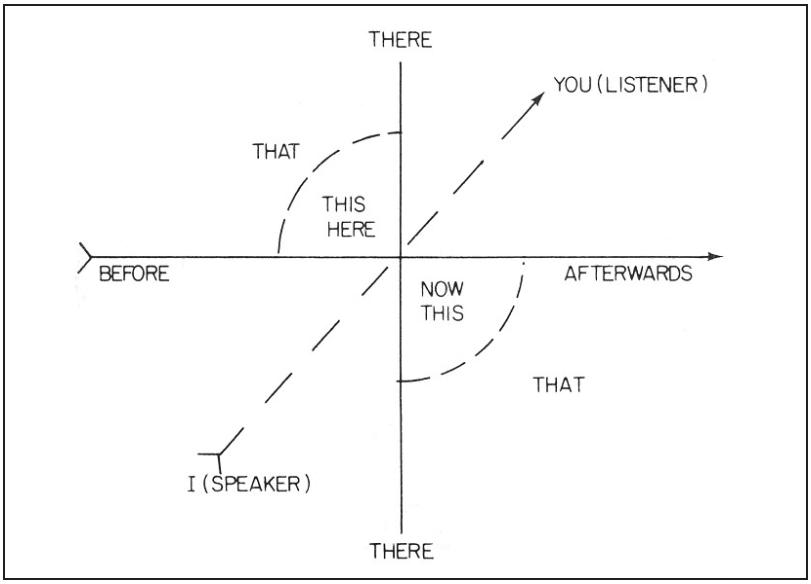


Figure 1: The spatial-temporal-interpersonal coordinates of the act of speech defining the emergence of intersubjectivity (Rommetveit, 1979b)

In this system, the 'I' is, in a sense, privileged in controlling what 'spots of awareness' penetrate the shared space. The speaker gets to point out "the objects, events and states of affairs to enter the field of shared attention" (Rommetveit, 1979a:95). Bringing the spatial-temporal dimension of intersubjectivity into the equation allows us to consider not only the object of speech acts but also their premises. What is already tacitly assumed or pre-supposed? How will this common ground nurture processes of constructing shared understandings about the joint objects attended to in activities? These questions also reflect some of Rommetveit's (ibid.) theoretical concerns, when exploring how activities are made sense of by participants who bring different understandings to the situation. Following this frame of mind, meaning emerging in interaction is to be seen as jointly constructed, grounded in the mutually accessible social realities which participants bring to the situation, forming a *partly shared social world*. It follows from the discussion that sense and meaning is considered relational, and perhaps also inherent in the trajectories connecting culturally situated subjects. However, the outlines of intersubjective states and the coordinates displayed in the above model may need to be re-written and adapted to the new interfaces connecting today's 'networked' knowledge practices. In order to re-contextualise the framework, I have modified a few general propositions about human cognition and communication issued by Rommetveit in the edited book *The Dialogical Alternative Towards a Theory of Language and Mind* (Wold, 1992). The propositions are selected and adapted to fit the theme (intersubjectivity and object relations) and the distributed context of software development. It is important to note that the following propositions are not intended to present a comprehensive account for human cognition and communication; rather, they are fragmented suggestions for exploration. This is in line with Rommetveit's encouragement for the original theses he proposed. In his own words, they "remain open to modifications in response to progress within related academic fields" (Rommetveit, 1992:24). In line with Rommetveit's encouragement, I have selected the original propositions that I consider to be most relevant for the aims and evolving issues of this thesis. Furthermore, these are modified and re-contextualised to a technology-mediated scenario involving joint object-orientation. My modifications thus bring in a few 'distributed dimensions' by referring to online discussions, distributed knowledge practices and joint object-orientation.

i. Online discussions involving joint object-orientation or shared attention towards objects is characterised by an "attunement to the attunement of the other". Partial

intersubjectivity can thus develop among the participants, but are reliant on reciprocal 'tuning' of perspectives and 'fixation of perspectives' on shared objects.

ii. Reciprocal adjustment of perspectives on shared objects may materialise when some features of the object are attended to, "made sense of, and talked about from a position temporarily adopted by [some of the] participants in the communication" (p. 23).

iii. Intersubjectivity on "some state of affairs S is attained by verbal communication when some aspect or set of aspects Ai of S is brought into focus", and jointly attended to by one or more participants (p. 23).

iv. Moving on to new levels of intersubjective understandings in distributed knowledge practices requires that participants assume (or take for granted) that some state of affairs S is sufficiently understood by the others.

The above propositions may appear cryptic without explanatory empirical examples. Note that Rommetveit (1974, 1979a) originally developed a framework for understanding processes of meaning making in face-to-face interactions. However, the research contexts of the empirical studies conducted as part of this thesis are predominantly technology-mediated. In distributed discussion forums on the Internet, software developers interact beyond a localised 'here and now', and thus move their argumentative problem solving to text-mediated (and highly specialised) peer networks on different time-place scales. By jointly attending to rather abstract shared code objects online, the process of reciprocal 'tuning' of perspectives might be influenced or restricted by the mediating technologies applied in the communication. What is made known in common about abstract code objects in these scenarios might perhaps, as Rommetveit (1979a) suggests, become part of their expanded shared social reality and serve as a prerequisite for making sense of subsequent utterances. However, in the research literature, there are currently no comparative studies available explaining such phenomena in the context of distributed software development. The modified propositions above are merely intended as suggestions for exploration in a somewhat unorthodox context. If a significant part of software developers' practice (argumentative problem solving) takes place outside the co-located 'here and now', it is reasonable to assume that the dynamics of meaning making will also be affected and the processes of achieving mutual understandings are of a somewhat different nature than the traditional framework predicts. Perhaps meaningful interaction around shared objects in distributed expert networks requires that more information is presupposed or taken for granted prior to the interaction. An interpretation of Rommetveit's (1979a) notion of *shared prolepsis* can be especially relevant for understanding the assumed

and taken for granted premises for intersubjectivity to develop around shared objects in these communities.

2.1.2 Shared prolepsis

The states of partial intersubjectivity are connoted by Rommetveit (1979a) as *shared prolepses*; a communicative act among participants in conversation where the speaker presupposes or takes for granted something that has not yet been discussed. Processes of intersubjectivity through this analytic lens involve relations between conscious minds, as individuals unite in their prolepses. According to Rommetveit, intersubjectivity is a premise to be presupposed in order to engage in communication. In the *Merriam-Webster Dictionary*, prolepsis is described as a form of anticipation or assumption of a future act or some other development as if it is presently existing (prolepsis, 2009). Michael Cole (1996) refers to prolepsis as a mediating cultural mechanism that relates past and future in human development. In his words, prolepsis "brings the end into the beginning" (p. 183). The mechanism is accordingly seen as operating throughout ontogeny. The term *prolepsis* dates from ancient Greek philosophy and rhetoric. It can be traced back to Epicurus' theory of language, where prolepsis is applied as a truth criterion relating to the process in which our experiences become general (Blackburn, 2008). His doctrine was that proleptic anticipations allow us to foresee the kind of objects to which abstract terms refer. Knowledge is thus described as a set of physiological processes that are derived from our perception of external objects. However, Epicurus also indicates the existence of cognitive processes in relation to prolepsis (ibid.). The subject is not merely seen as a passive recipient but rather in dynamic contact with what is perceived. Prolepsis may accordingly emerge as a consequence of repeated exposure to, or interaction with, a given object.

The concept of prolepsis was reintroduced into modern social theory and socio-linguistics by Rommetveit (1974) to explain how intersubjectivity 'here and now' is based on a set of tacitly taken-for-granted premises. In conversation among actors, the premises include a blend of past experience and future expectations. He argues that "the temporarily shared social world is in part based upon premises tacitly induced by the speaker" (p. 87). The 'tacitly induced' in this context means that some elements are left out of the conversation and the listener is subsequently invited to step into a partially shared space of intersubjectivity to make sense of what is left out. To paraphrase Rommetveit (1979:167), the process involves the listener who

is made an insider of a tacitly expanded and enriched ‘here and now’. Moreover, the formation of prolepsis is seen as a dynamic and recursive process in the sense that the actors continually fine-tune their assumptions and expectations in reaction to the others’ contributions and feedback (Matusov, 2001). An example of prolepsis is offered by Rommetveit (1974) in a personal letter received from a friend:

> Today, I walked with one of the psychologists here past the Mayflower cinema in Eugene, where Bergman’s latest film movie is being shown. He asked me whether I had seen it. I said no, and asked if he had. He said yes, he had. I asked him how he liked it, and he said ‘I liked it very much, but Mary Ann did not’; without ever explicitly having ‘made known’ to me that he is married and that his wife’s name is Mary Ann, that they went to see the film together, and a lot of other things – and (if I am correct) without assuming that I knew all this. His utterance was proleptic in that it triggered a search on my part for a shared social reality which in turn would provide a basis for understanding the sentence. Incidentally, it would have been barbaric and pedantic to say, ‘Oh, Mary Ann is your wife’. To be precise, prolepsis here served to establish a relationship between his wife and me as persons who should at some time get together. My comment would have been a crude rejection of that implication. (1974: 87–88)

An issuer of utterances might, in this sense, presuppose something that is not yet introduced in the conversation. An important point is that such presuppositions may challenge the listener and “trigger anticipatory comprehension, and what is made know will hence necessarily transcend what is said” (p. 88). Such presuppositions or proleptic instances may thus challenge the receiver and trigger the construction of new understandings of the intended meaning. Prolepsis can then serve as a catalyst for making sense of not-yet-provided information. This process is therefore accorded a crucial role in interpersonal communication and an essential aspect of intersubjectivity. It can be seen as prompts or cues for constructing intersubjectivity in allowing the issuer of utterances “to presuppose shared knowledge that has not yet been introduced into the interaction, but which is essential to making utterances interpretable” (Cole, 2002:312). For example, prolepsis in literature or narrative storytelling can be conceived as presenting a hint or an ingredient (subtle or explicit) earlier than what is logically related, triggering anticipatory world-building practices of the reader (Bridgeman,

2005). Rommetveit (1979a) also points out that such assumed shared presuppositions are often encountered in fiction:

> When the reader feels (correctly) that he has comprehended something in addition to what he actually has read. What from strictly 'objective' or 'public' point of view appears an unwarranted presupposition on the part of the creative writer may then more appropriately be conceived of as self-fulfilling assumptions by which the reader is made an insider of a tacitly expanded and enriched 'here and now'. He is made an insider — not merely informed about it — precisely because that expanded social reality is taken for granted rather than explicitly spelled out (p. 167).

This example suggests that processes of prolepsis can take the form of creative assumptions that are somewhat unwarranted, but nevertheless invites the listener to take part in a partially shared intersubjective space (represented by the inner circle of Figure 1). Furthermore, the temporal social reality to which the listener is made an insider is seen as 'tacitly expanded' and draws on elements that are only implied in the situation without being spelled out in detail. It follows that the proleptic processes are not clearly expressed, but may perhaps be inferred from partakers' semiotic interactions. As I read Rommetveit (1974, 1979a), prolepsis is not an absolute requirement for constructing intersubjectivity; but proleptic incidences may enhance or 'boost' the progression towards greater intersubjectivity. He emphasises that the transcendence of the private perspectives of the actors in communication into states of partial intersubjectivity, "presupposes a capacity for decentered categorization and attribution, reciprocal role-taking, and complementarity of intentions" (Rommetveit, 1978:30). This reciprocity and complementarity among actors in communication can accordingly be seen as "pragmatic postulates in the construction of intersubjectivity" (ibid.). For instance, in a forum for software developers on the Internet, partakers are liable to assume that replies are actually trying to answer the question posed previously in the thread in order to make sense of the reply. In Rommetveit's argument, this is also the case when the response sounds peculiar and imprecise. The faith in our peers will lead us to

> search for some potential taken-for-granted (by him) aspect of our only partially shared here-and-now which may confirm my faith. If I succeed, our shared here-and-now [intersubjectivity] is immediately expanded, and we have an instance of 'prolepsis' (p. 30).

In this manner, actors can search the 'tacit', taken-for-granted or otherwise hidden information for clues to assist their current interpretations. In this process, they may infer from tacit information, reaching beyond the tangible situation. Proleptic incidences may thus carry the potential to advance intersubjectivity by initiating sequences of "anticipatory comprehension" and "what is made known will necessarily transcend what is said" (Rommetveit, 1974:88).

Rommetveit's notion of prolepsis has been contrasted with the one of ellipsis by Leo Van Lier (2004) when exploring a sociocultural perspective on language learning. He maintains that ellipsis is what takes place when information is omitted because it is regarded as redundant. The listener or reader is thought to be capable of filling in the missing pieces. Opposed to prolepsis, an issuer of elliptic utterance in speech or written text "does not explicitly check or facilitate the listeners' interpretive process, or invite the listener into a shared intersubjective space" (p. 152). As a negative connotation to the notion of prolepsis, the issuer of utterances does not provide the receiver with clues that invite enlargement of their intersubjectivity. However, such an utterance in proleptic communication also omits information, but "invites the less-competent into sharing with the more-competent" (ibid.). It follows that proleptic communication is more inviting than elliptic and perhaps also more pragmatic in foreshadowing an expansion of joint understanding. The process of omitting information, combined with providing minimal cues, invites the listener to participate in the co-construction of a greater intersubjective space (Rommetveit, 1974). As a consequence, the receiver of proleptic utterances can actively participate by filling in the missing pieces and thus produce the information needed to achieve a level of joint understanding. Assisted by a more competent other, something is seen that is not yet added to their common ground. In this way, new information can be 'nested' in what is already (partially) known in common. If successful, the 'private worlds' of participants may transcend into an expanded intersubjective state.

For example, in the open-source software development project considered in this thesis, I find it reasonable to assume that proleptic incidences play an important role in the processes of constructing intersubjectivity around joint objects of development. Communication among peers in the project is conducted entirely by exchanging e-mail messages and short comments integrated in the programme code. One can argue that a prerequisite for meaningful

interaction in the project is the ability to construct a set of assumptions in order to make sense of the other developers' utterances. Without this ability, the messages from other developers may appear short and incomprehensible. On the contrary, the capacity and willingness to interpret proleptic utterances by filling in the missing pieces and recreating indications or presuppositions concerning their joint object might be essential for developing and maintaining intersubjectivity in this context.

2.1.3 Clark and Brennan's extended contribution theory

In the article "Contributing to discourse", Clark & Schaefer (1989) introduce a collaborative model for human communication. The model expands a more traditional sender/receiver perspective on communication by broadening the analytic frame from single utterances to *contributions* developed in interaction. Conversation is thus seen as a collaborative achievement, and not merely as sender-receiver turn-taking. They maintain that conversations progress as a collaborative effort, by contributing to a *common ground*. In order to produce a meaningful contribution, participants should accordingly take an active part in establishing what the speaker meant. The process of adding information to a shared understanding in this view is labelled *grounding*. In the process of grounding information while communicating, it is suggested that participants take mutually intelligible collaborative steps in order to produce contributions. However, the article primarily explains how shared understandings evolve in unmediated face-to-face conversations. In a later work, Clark and Brennan (1991) elaborate and extend the contribution theory by discussing two main factors that 'shape' grounding processes. The first is *purpose*; what participants in communication are trying to accomplish. The second is the *medium* of communication and associated resources and constraints on grounding.

For the purpose of analysing the mediated communication encompassing the third aim stated in the introduction of this thesis, I regard the extended theory proposed by Clark and Brennan (1991) as most relevant. An account of how these processes develop through mediating technology is absent from Rommetveit's framework for intersubjectivity, presented in the previous chapter. The extended theory of Clark and Brennan (ibid.) can be a useful accumulation for understanding how intersubjectivity develops around joint knowledge objects in a networked, technology-mediated scenario. In the following paragraphs, the extended theory is further described.

Clark and Brennan (1991) postulated that “all collective actions are built on a common ground and its accumulation” (p. 222). To coordinate the content and process of human conducts, participants must accordingly contribute and update their common ground moment by moment. From this perspective, the organisation of the different elements of complex activities builds on the participants’ assumptions of some shared information or common ground. Such presupposed shared knowledge relates to Rommetveit’s (1974, 1978) notion of prolepsis. However, Clark and Brennan are not explicit in explaining how anticipations about what is known in common prior to the interaction can expand their intersubjectivity here and now. On the other hand, they maintain that the grounding processes in which partakers create and maintain such a common platform vary according to the context of interaction. They suggest that grounding processes in face-to-face conversations are qualitatively different from those in distributed mediated forms of communication, such as asynchronous e-mail correspondence. This idea is elaborated in detail when explaining how processes of grounding develop with different media for communication (see Table 3).

Analogous to Rommetveit’s initiative that shared understandings are only partially developed and never complete, Clark and Brennan (1991) point out that “understandings can never be perfect” (p. 223). They continue by postulating that people will attempt to reach a *grounding criterion* in conversation. This criterion states that a sufficient condition for ‘moving on’ with their collective activity is that “the contributor and the partners mutually believe that the partners have understood what the contributor meant to a criterion sufficient for the current purpose” (p. 223). Moreover, their research indicates that the grounding criterion relies on the goals and features of the joint activity. It follows that this criterion may vary according to the collaborators’ joint object of activity and the type of communication medium applied. The foundation for these claims is the model of grounding in conversation proposed by Clark and Scheaefer (1989). This original model is also a basic premise in the extended theory, but the developed version predicts that “people should ground with those techniques available in a medium that leads to the least collaborative effort” (Clark & Brennan, 1991:229). Let us now look more closely at how contributions to a common ground occur in this combined and extended framework.

Clark and Brennan (1991) divide the production of contributions to conversation into two general phases:

i. *Presentation Phase*: A present utterance *u* for B to consider. He does so on the assumption that, if B gives evidence *e* or stronger, he can believe that she understands what he means by *u*.
ii. *Acceptance Phase*: B accepts utterance *u* by giving evidence *e* that she believes she understands what A means by *u*. She does so on the assumption that once A registers that evidence, he will also believe that she understands (p. 224).

Potential contributions to a common ground are thus initiated when an utterance is presented for a collaborator to consider. The issuer intentionally specifies the content of his contribution for the partner to register it. However, according to the theory, A cannot possibly know if he accomplished a contribution to their common ground unless B provides some clues or evidence of understanding. In consequence, both phases have to be completed in order to produce a valid contribution to the common ground between A and B. After the presentation phase, they portray the partner (B) in one of the following four states:

State 0: B didn't notice that A uttered any *u*
State 1: B noticed that A uttered some *u* (but B wasn't in state 2)
State 2: B correctly heard the content (but wasn't in state 3)
State 3: B understood what A meant by *u* (ibid.:224)

The point here is that if B reaches state 3 with regard to understanding the utterance, it can be added to their common ground. The logical intent of A's utterance would thus be for the partner to reach state 3. However, Clark and Brennan argue that for the utterance to become 'grounded', B must grant A some positive evidence that the content is understood in a mode that is sufficient for the current purpose. Although if the uttered content *u* is not in state 3 after the presentation phase, B can communicate negative evidence (indicating that *u* is misheard or misunderstood) as part of a repair sequence. The model also claims that "people will ultimately seek positive evidence of understanding" (p. 224). This means that throughout a conversation, A will try to confirm that the partner understood the content well enough for the current purpose. This monitoring of the partner's understanding could provide the presenter of utterances with both positive and negative evidence, e.g., in the form of verbal and non-verbal cues. Moreover, Clark and Brennan assert that if negative evidence is found, repairs will be attempted. Examples of this are repetitions or asking questions for clarification. On the

contrary, if no negative evidence is indicated by B, A will assume that the *u* is understood and the information can be added to their common ground.

The basic forms of positive evidence in the view of the contribution model are *acknowledgements*, initiation of *a relevant next turn* and *continued attention* (Clark & Brennan, 1991:225). Examples of acknowledgments mentioned are continuers like "yeah," or "mmhm". They are generally produced by receivers of utterances (without taking a turn) to indicate the belief that they have so far understood what the issuer has meant. When initiating a relevant next turn as positive evidence of understanding, B launches the next contribution at the equivalent level of relevance as the current one. B thus produces an utterance that is an appropriate response to the previous turn, for example, by answering a question. Moreover, this contribution carries meaning that makes sufficient sense in the context to carry on the conversation. However, for the B to be able to present a relevant next turn, it is reasonable to assume that some degree of mutual understanding has already been obtained. A relevant next turn can then be seen as a signal of a pre-established shared understanding. Showing continued attention is the third basic form of positive evidence mentioned. This can be provided by monitoring the conversation partner's action moment by moment. For example, if B continually attends to A's presentation with serene eye contact, it implies that the message is understood well enough for the current purpose. On the other hand, if B does not pay attention, i.e., if B is interrupted or looks the other way, A can hardly assume that the content is understood.

Inferred from the grounding criterion described above is the principle of *least collaborative effort* (Clark & Brennan, 1991:226). It states that, in communicating, actors are unlikely to spend more collaborative effort than is needed for the current purpose. Clark and Brennan (1991) declare that "in conversation, the participants try to minimize their collaborative effort – the work that both do from initiation of each contribution to its mutual acceptance" (p. 135). It follows that the total effort of the conversation partners tends to be minimised. This means that actors are liable to not work harder than they have to in order to be understood. For instance, it can be easier to produce a short, 'imperfect utterance' that can later be repaired, than formulating a comprehensive and 'perfect utterance'. In the former mode, speakers can also try out more complex or provisional utterances (by trial and error) and check the receivers' level of understanding.

When studying open-source software developers' mediated communication as part of this thesis, the collective purpose (or shared object of development) can be an important aspect when analysing how shared understandings evolve. Following the grounding criterion, if receivers of messages in the developer forum are to comprehend the meaning sufficient for their current purposes, the criterion should also change along with the collective purpose. This indicates a dynamic relation between the participants' grounding processes and a moving object of development. As the shared object of development to which they are committed progresses and new problems emerge, it follows that the content to be grounded and the associated criterion need to be reconsidered. This means that we, as analysts, ought to be sensitive to changes in the collective purpose of conversation when studying acts of grounding. Modes of grounding might be affected by the continuous shifts associated with Knorr Cetina's objects of knowledge (see chapter 2.3.1). When a shared knowledge object evolves as a result of the joint effort among experts, new problems may arise, and it is therefore reasonable to assume that the content of their subsequent communication will be transformed.

Clark and Brennan (1991) continue by discussing how grounding changes with different media. They put forward that different media offer various constraints and costs on grounding processes. For example, communicational media such as video-teleconference, offer visibility and co-temporality, while e-mail or personal letters are coupled with constraints on grounding due to lack of these features, causing delays in turn-taking. They propose a set of possible constraints that a medium may impose on communication and, consequently, affect grounding between actors A and B. Their prediction is that "people should ground with those techniques available in a medium that leads to the least collaborative effort" (ibid.:229). What takes collaborative effort can accordingly change with the communication according to the properties cited in the table below.

Constraints	**Properties**	**Examples**
Co-presence	A and B share the same physical environment.	Face-to-face conversations.

Visibility	A and B are able to see each other.	Face-to-face conversations and video conferences.
Audibility	A and B can communicate by speaking.	Face-to-face conversations. Video conferences. Telephone or Voice over Internet Protocol (VoIP).
Co-temporality	B receives approximately the same time as A produces.	Same as above.
Simultaneity	A and B can send and receive utterances concurrently.	Face-to-face conversations, telephone calls or synchronously mediated text (e.g., instant messages).
Sequentiality	When A's and B's turns cannot be out of sequence.	Face-to-face. Phone calls. Video conferences or voice messages.
Reviewability	When B can review A's message. Utterances are 'stored' so that they can be reviewed at a later time.	E-mail correspondence or other types of asynchronously mediated text (discussion forums, blogs, etc.).
Revisability	When A can revise messages before sending them to B.	Same as above

Table 3: Constraints that a medium may impose on communication between two people, A and B, adapted from Clark and Brennan (1991: 229-230) and Fugelli (2004:28-30).

If a medium lacks one or more of the characteristics mentioned, participants may use alternative grounding techniques. As indicated above, the effort of making contributions to the common ground may vary according to the medium of communication. The following costs are considered for the different media (Clark & Brennan, 1991: 230-233).

- ≠ *Formulation costs* are the effort it takes to formulate utterances. How easy is it to decide what to say?
- ≠ *Production costs* are costs associated with producing interpretable utterances. How difficult is it to articulate or type messages?
- ≠ *Reception costs* are what it takes to receive and process the utterances. It can be easier to listen to oral messages than to read text on a monitor.
- ≠ *Understanding costs* are the effort needed to interpret and understand the utterances in the shared context.
- ≠ *Start-up costs* are associated with the effort required to initiate a conversation.
- ≠ *Delay costs* are the costs linked with delayed communication.
- ≠ *Speaker change costs* are the endeavour coupled with turn-taking.
- ≠ *Display costs* are the ones needed to provide visual cues or gestures.
- ≠ *Fault costs* are the costs coupled with producing faults or mistakes while communicating.
- ≠ *Repair costs* are the ones associated with repairing faulty communication.

Since different media are suggested to afford different 'blends' of constraints and costs, it is reasonable to assume that the principle of least collaborative effort, and perhaps also the prerequisites for development of intersubjectivity, are influenced by the media applied in communication. The internet-based technologies applied in the communication among the actors focused upon in this thesis may be constrained by several of the properties rendered in Table 3. However, the contribution model and the notion of common ground have been criticised by researchers examining how shared understandings evolve in computer-mediated collaborative systems (e.g., CSCL and CSCW). For instance, Traum (1999) points out that it is unclear what to count as a valid acceptance phase for new additions to a common ground. According to Baker et al. (1999), there is also a cultural dimension missing in approaches that solely target the inner dynamics of interpersonal interaction. Baker et al. (ibid.) put forward a sociocultural alternative which builds on Clark's grounding model in which the unit of

analysis is moved beyond the given (speaker-listener) interaction. It is suggested that grounding can occur on both a pragmatic and a semantic level. The former involves processes associated with coordination and learning how to collaborate. The latter includes grounding of semantic referents or meanings of utterances. In this view, important facets and prerequisites for processes of grounding are for the participants to appropriate the cultural tools applied in joint actives. Another relevant critique to mention here is provided by Nova et al. (2008), who argue that the notion of common ground is "far too mentalist and lacks situatedness" (p. 1). Their main claim is that the model can be more useful for analysing joint activities in computer-mediated contexts if it is broadened to account for general notions of group coordination. In computer-mediated joint activities, they propose that the organisation of different elements generally requires the ability to foresee the actions of others (ibid.). Their studies have shown how groupware coordination devices may afford such predictability. A relevant issue derived from the research aims of this thesis is if such predictive abilities also apply to shared understandings of knowledge objects. Here, the concept of prolepsis, reintroduced by Rommetveit (1974), might be useful in order to explore how processes of intersubjectivity around joint objects in software development are based on a set of tacitly taken-for-granted premises. It is reasonable to assume that the premises for achieving shared understandings around such objects would include appropriation of necessary cultural tools, participants' past experience with the object, as well as future expectations about how the object ought to be developed. These issues are developed further in dialogue with the empirical data.

2.2 Artefacts and mediation

"the most fundamental element of any technology, the artifact"
(Cole & Derry, 2005:211)

The word *artefact*, applied in post-Marxist social theory, originates from scholarly Latin. Dissected into its two parts, *arte* refers to 'by or using art' and *factum* to 'something made' (Soanes & Stevenson, 2006). As the Latin name implies, artefacts are constructed by humans

for some practical purpose. They are commonly seen as basic components of human culture, e.g., more or less advanced tools, books and various representational technologies (Darvill, 2002). In academic literature, there is no general agreement on the relationship between the terms ‘tool’, ‘object’ and ‘artefact’ and they are sometimes used interchangeably. Wertsch (1998) points out an important attribute that may be common for physical entities; namely, that their structures may continue to exist (altered or changed) across time and space even when not incorporated into human action. As such, they can possibly represent reifications of ideas or external memories. This brings us to a rather useful classification of artefacts, and a major influence on today’s cultural-historical activity theory, constructed by Wartofsky (1979). In his historical epistemology, he probes how activity is interconnected with representations and the significance of artefacts in social practices. He rejects a simplistic notion of activity as purely unmediated perception and puts forward the idea that human action is constantly mediated by tools and signs. In this line of thought, tools and signs have representational structures which are purposeful in human practice.

However, there is also an evolutionary/historical aspect associated with this approach. In our practices, we not only create new tools and signs, but also evolve new modes of working and interacting with the artefacts produced. Wartofsky thus implies that artefacts operate on different levels, depending on their closeness to action or practice. To categorise such aspects of artefacts, Wartofsky (1979:201) suggests three levels: primary, secondary and tertiary. The primary level comprises tools, practices and skills directly applied in human labour/production. Most productive activity is assumed to involve this first level of artefacts. For instance, a metal craftsman’s air hammer has embedded functions which become evident when acted on in practice. This tool is a sophisticated hammer that uses compressed air to get the right momentum to shape or smooth pieces of metal. The next level is secondary artefacts; the representations of the activities and practices involved in performing with the primary ones. They embody the potential to conserve and pass on modes of action associated with utilising the primary artefacts. Continuing the above example, this can be a paper-based manual on how to operate and use the air hammer, including a graphical model representing the actual activity of shaping metal. In Wartofsky’s (1979) own words, secondary artefacts are

> representations of such modes of action [associated with primary artefacts], and in this sense are mimetic, not simply of the objects of an environment which are of interest or

> use in this production, but of these objects as they are acted upon, or of the mode of operation or action involving such objects (p. 202, my comment included in brackets).

Furthermore, secondary artefacts convey *reflexive embodiments* of activity; they are considered to externalise or reflect modes of action. This means that the artefacts of this kind categorise externally embodied representations, not metaphysical entities existing in the minds of humans. In short, they are described as "distinctive artefacts created for the purpose of preserving and transmitting skills in the production and use of primary artefacts" (ibid.:202). Secondary artefacts can thus be 'objectified resources', transmitting and reproducing knowledge and skills about the productive use of primary artefacts in practice. Finally, tertiary artefacts are not directly related to productive practice, but rather have an indirect influence on it. They are mere abstractions from secondary artefacts without a direct representational function. However, as mediators of visions or future states of affairs, they may embody a transformative function in (indirect) relation to practice in changing our 'mind set', and in this manner amend how we act. This class of artefacts has gained renewed attention in contemporary social theory as "one of the key signifiers of the knowledge society is the structure of its knowledge, whose axes shift from local and personal to abstract and symbolic inputs" (K. Jensen & Lahn, 2007:10). The ability to relate to and work productively with abstract knowledge may perhaps be a necessary *habitus*[2] for today's knowledge workers. What is, however, embedded in tertiary artefacts can be seen as rather transformable, detached abstractions (derived from secondary artefacts) that can cross the boundaries of localised knowledge practices to preserve and transmit higher-order knowledge.

In software production, tertiary artefacts may influence the act of coding by mediating higher-order abstraction on how to organise programmes. Examples from modern object-oriented software development are design patterns, representing ideal (best case) practices of organising computer programmes. The aim of patterns within the software community is to provide concepts to assist developers in resolving recurring problems throughout the development cycle:

[2] The concept of habitus was introduced by Bourdieu (1977) to signify acquired dispositions of thought, actions, preferences, etc.

> Patterns help create a shared language for communicating insight and experience about these problems and their solutions. Formally codifying these solutions and their relationships lets us successfully capture the body of knowledge which defines our understanding of good architectures that meet the needs of their users. Forming a common pattern language for conveying the structures and mechanisms of our architectures allows us to intelligibly reason about them. The primary focus is not so much on technology as it is on creating a culture to document and support sound engineering architecture and design (Appleton, 2000:1).

Design patterns are thus intended to influence modes of software programming by inscribing experienced designers' abstract knowledge. As I read Wartofsky, this represents the type of 'detached knowledge' that is not written down in the user manual or documentation (secondary artefact) of an application (primary artefact). Even though every level of artefact described above might be understood autonomously, it is suggested by Cole (2005) that "each with its own mixture of materiality and ideality arises from, and acts back on, the other" (p. 213). It follows that in order to understand the role of artefacts in social practices; their potential interacting levels ought to be considered. Also, processes of intersubjectivity can be problematic at the various levels of artefacts (see chapter 2.1). For instance, are tertiary artefacts, like design patterns, presupposed or taken for granted prior to developers' interactions (as parts of their prolepses)? To what degree are they made explicit in joint problem solving? Perhaps they are partially shared in an implicit way.

Of relevance for the issues under development in this thesis are the humanly imposed attributes and 'crystallised' experience embedded in artefacts that we can study and from which we can learn. According to Miettinen (2005), "it is necessary to study the objectified, embodied and material forms of organizing and acting" in order to understand practices and routines (p. 442). He refers to Latour (1991), suggesting that we ought to search for the nuts and bolts of social order in material objects, not in abstract forms of sociality. However, understanding the role of artefacts in human conduct is not straightforward. As aspects of materiality, they appear along different continuums, ranging from simple tools like chalk and blackboard to interactive tutorials in a 3D virtual world such as Second Life (secondlife.com). The development of artefacts can thus reflect basic problems or advanced functions. Inspired by Kroeber and Kluckholm's (1952) conceptualisation of artefacts as the essential core of

cultures, Cole (2002) presents a definition that expresses their transformative incorporation into human socio-cultural activity:

> An artefact is an aspect of the material world that has been modified over the history of its incorporation into goal-directed human action. By virtue of the changes wrought in the process of their creation and use, artefacts are simultaneously ideal (conceptual) and material. They are ideal in that their material form has been shaped by their participation in the interaction of which they were previously a part and which they mediate in the present (p. 307).

The definition above is adjacent to Wartofsky's secondary artefacts mentioned earlier; the symbolic representations of primary artefacts. In his definition, Cole claims that artefacts are dual in nature, simultaneously having conceptual and material characteristics. A man-made physical object may, as Wartofsky also argues, represent ideas. In this sense, they are conceptual or ideal when modified to incorporate human intentions. They are thus shaped as functional objects to assist intentional, goal-directed actions.

The functional and structural properties of artefacts/tools are also a central concern in the Russian cultural historical tradition (Leont'ev, 1978; Luria, 1979; Vygotsky, 1962, 1978). It is assumed in this school of thought that there is a close relationship between human actors' unique environment and their primary mental processes (Cole & Wertsch, 1996). Here, artefacts are regarded as important *mediators* of action. Vygotsky (1978) postulated that instead of acting in direct response to the environment, humans predominantly react indirectly through *mediation*. This relation of artefacts to humans is illustrated in a basic mediating triangle:

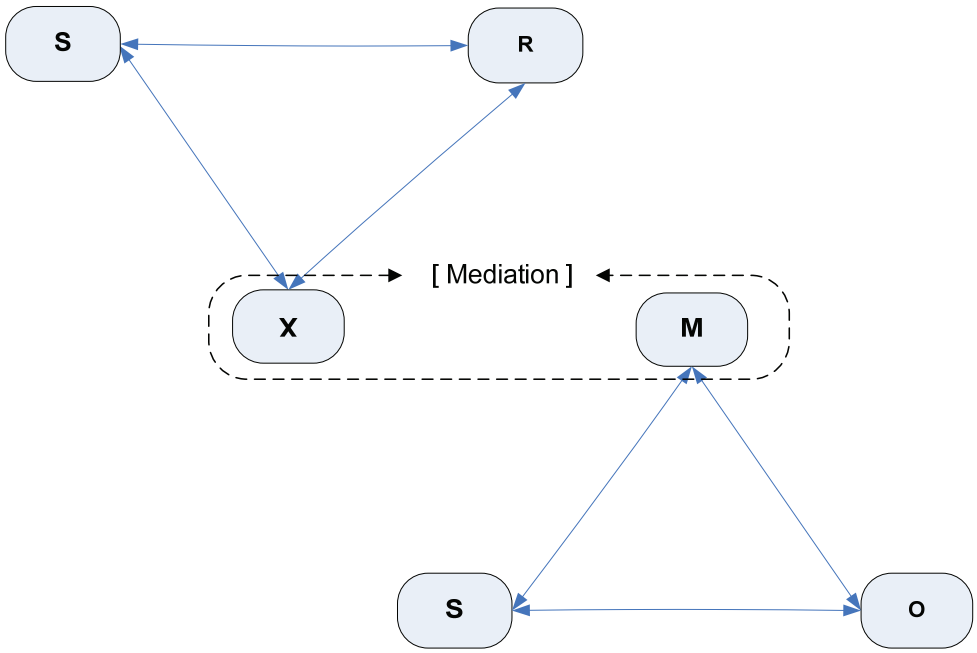


Figure 2: The mediating triangle from Vygotsky (1978) to its reformulation by Activity Theorists

In the triangle to the upper left in the model above, a mediating relation (X) is set up between the stimuli (S) and response (R). The mediated functions first emerge when the relation between the subject and its environment 'connects' by means of tools or artefacts (Cole & Derry, 2005). In activity theory, this classic mediating triangle is often reformulated with the S indicating the subject, O the object, and M the mediating artefact (the lower right triangle in Figure 2 above). Human conduct in this framework is generally directed towards an object and mediated by artefacts (Nardi, 1996). However, according to Cole (1996), this does not mean that actions are always mediated and a more natural path of cognition is replaced. It follows that activity is in *general* mediated and directed towards an object. Artefacts consequently play a role as mediators in the equation and are perceived as meaningful objects "created by people for specific use" (Stahl, 2002:66). They are also assumed to have an important mediating function in collaborative interactions.

In line with the above reasoning, artefact-mediated and socially transmitted human activities may serve as resources for current doings. In this sense, artefacts are functional as reifications

of ideas, carrying knowledge inherited from a specific culture. For example, many software providers offer end users some variety of an electronic 'Frequently Asked Questions' (FAQ) on their online support pages. An FAQ is a list of questions and answers relating to particular problems that end users might request assistance on how to resolve (Soanes & Stevenson, 2005). An FAQ can then be interpreted as a mediating artefact containing accumulated experience, guiding present and future problem-solving activity. It reflects intended purposes, accumulated experiences and knowledge about a set of problems. Such entities can thus function as mediators across time and space, convenient resources for accumulating and transmitting social knowledge. However, the artefacts and tools of social practices are not necessarily static resources resisting change during a period of time. Artefacts may develop progressively according to prior experience. In the book *The Evolution Of Useful Things*, Petroski (1992) introduces a hypothesis that artefacts evolve because form follows failure. Through a series of case studies, he indicates that modifications and new innovations in artefacts follow users' discontent with the existing 'status quo'. For instance, many software applications develop further on the basis of users' 'bug reports' and missing features. In the light of Petroski's (1992, 2006) insight, the current version of a software artefact can be seen as a response to inadequacies in the preceding version.

The concepts examined so far can potentially illuminate or inform our understanding of artefacts in technology-mediated contexts. Although, the different levels at which artefacts operate, suggested by Wartofsky (1979), may be problematic to distinguish and perhaps operate in a more 'blended' and integrated fashion with respect to computational artefacts. For example, among software developers, it is common to use a so-called Integrated Developer Environment (IDE) to support the process of writing software. This is a set of tools compiled into one application that can be installed on the developers' personal computer. Applying Wartofsky's concepts, these can be interpreted as secondary artefacts. Furthermore, an IDE commonly incorporates a help system that explains abstract concepts (such as design patterns) which I assume operate at the level of tertiary artefacts. However, in such integrated computational tools, the secondary and tertiary levels are not clearly detached from the primary ones. Another complicating aspect is that the integrated secondary artefacts (help system) may include hypertext linking to distributed resources on the internet. According to recent developments in socio-cultural theories on the relation between humans and objects (see e.g. Miettinen & Virkkunen, 2005) and actor network theories (see e.g. Hernes & Czarniawska-Joerges, 2005; Latour, 2005), such entities are said to be of 'hybrid' nature.

Investigators observe and describe networked/ICT-mediated contexts in which ideas and materials are merged, and hybrid entities play a central role in structuring collaborative activities.

An empirical example related to this thesis is the study reported by Lanzara and Morner (2005), which focuses on the role of artefacts in organising Open-source Software projects. They find that artefacts (technological or non- technological), inscribe major coordination and governance functions that in complex activity systems are usually assumed by formal organizational structures. In the 'ecology' of artefacts applied in a distributed development project, the programme code jointly attended to and the project mailing lists showed as significant in coordinating activities. Moreover, the study indicates that source code attracts and organises the swarm of contributing developers by evolving into multiple versions. Earlier bugs are reported and new functionality (issues resolved from earlier versions) is documented along with the code. In this manner, a programmer's act is partly structured by previous developments and from the current state of the code. Artefacts in this case thus play a decisive role. However, the available research literature is limited with respect to explaining how artefacts or objects jointly oriented to in distributed development projects intersect with the localised problem solving of software developers. In the next subsection, I explore ideas on objects and practice in order to illuminate these issues. When interpreting how artefacts mediate relations in distributed expert networks, it is pertinent to ask: Do particular knowledge-intense trajectories emerge between local actors and knowledge objects separated in time and space? If so, how can they be represented and rendered meaningful for analytic purposes?

2.3 Objects and practice in contemporary social theory

To gain a general impression of how contemporary perspectives on objects are tied to a theory of practice, I dedicate a few paragraphs to what is referred to as the practice turn (Knorr Cetina, Savigny, & Schatzki, 2001). In recent social research and philosophy, the notion of practice and object has gained renewed attention (Knorr Cetina, 2001a; Lash, 2001; Law & Singleton, 2003). The 'practice approach' is demarcated by Schatzki (2001) as all analyses that "(1) develop an account of practices, either the field of practices or some subdomain thereof (e.g., science), or (2) treat the field of practices as the place to study the nature and transformation of their subject matter" (p. 11). According to Schatzki, the commitment to "a

field of practice" is central in this approach. Furthermore, it is emphasised that "practice theorists conceive of practices as embodied, materially mediated arrays of human activity centrally organized around shared practical understanding" (ibid.:11).

The concept of knowledge practice is reflected on by Nespor (1994) in the book *Knowledge in Motion*. He proposes to account for knowledge practice as "interaction with others distant in time and space" (p. 8). Following the ideas of Nespor, activities distant in space and time are commonly transported into particular settings through material resources and representations (i.e., textbooks, journals and representational technologies). Distributed knowledge practices can be seen as interaction with others on a different time-space scale. For example, the usage of representational technologies expands the social organisation of interaction and learning beyond the immediate face-to-face situations, mediating abstract knowledge and activities of individuals who are not physically co-present.

Knorr Cetina (2001a) applies the term 'epistemic practice' when contributing to the concept of a knowledge society. In her view, knowledge-generating practices in post-industrial societies are no longer restricted to the work of scientists. She maintains that:

> The transition to knowledge societies involves more than the presence of more experts, more technological gadgets, more specialists rather than participant interpretations. It involves the presence of knowledge processes themselves — in the terms chosen here, it involves the presence of epistemic practice (p. 2).

For example, the coding practices of software developers may perhaps contain 'scientific' features such as researching new technologies, consulting peers, contributing to the field (e.g., by documentation and sharing code) and disseminating knowledge in distributed developer forums. Furthermore, the practice of software development can be seen as dynamic in the sense that their knowledge base is continually moving. Epistemic practices are, according to Knorr Cetina (2001a), centred on objects of knowledge and "sustained by object relations" (p. 184). In order to warrant such claims, she has, during the last decade, conducted several empirical studies of experts' work targeting the social aspect of objects (see e.g. Knorr Cetina, 1999; Knorr Cetina & Brugger, 2002). Sociality with objects, she maintains, is a social form that represents an alternative to contemporary experience of individualisation (Knorr Cetina, 1997). Moreover, it is suggested that "these object worlds need to be included in an expanded

conception of sociality and of social relations" (ibid.:9). In the knowledge society, the 'loosening up' of conventional social ties comes with a side effect; namely, the growth of object-centred environments that promote new forms of sociality with non-human objects. This growth is argued to be a consequence of the increased 'spill over' of knowledge processes and knowledge structures from expert communities into other areas of social life. These scattered knowledge processes are regarded as closely tied to objects of knowledge. She endorses the idea that modern sciences has provided for and reinforced this type of object relation. Hence, what characterises objects of knowledge? Can we discern useful tenets and operational definitions for empirical analysis from this generic concept?

2.3.1 Relations towards objects of knowledge

Knorr Cetina (1999, 2005) and Rheinberger (1997) derive concepts such as epistemic things, epistemic objects or knowledge objects from studies of scientists in laboratories and experimental settings. In the literature, however, these notions are sometimes applied interchangeably. My intention in the following section is to explore the analytic potential of conceptualising 'software entities' as objects of knowledge. I have chosen to use the term *knowledge* instead of *epistemic* to demarcate a non-scientific context. I find it reasonable to assume that the knowledge practices of software developers are less reliant on scientific inquiry as followed in the natural sciences and the associated epistemology. Yet, the practices of natural scientists and software developers may share several similar features.

Elaborating on experimental systems as the functional micro-unit of research, Rheinberger (1997) distinguishes between two elements within knowledge-constructing systems: *scientific objects or epistemic things* and *technical conditions or technical things*. The first is to be understood as the scientists' main object of research. In Rheinberger's words, "they are material entities or processes — physical structures, chemical reactions, biological functions — that constitute the objects of inquiry" (ibid.:28). As epistemic objects, they appear as irreducible and vague, embodying the unknown. During acts of scientific inquiry, these objects need to be recurrently redefined as new knowledge emerges in experimental contexts. The other component is what Rheinberger refers to as technical conditions or technical things. They "determine the realm of possible representations of an epistemic thing" (p. 29). When sufficiently stabilised, the epistemic things or objects of inquiry can turn into the technical repertoire of an experimental setting. An illustrative example can be when a researcher within

the field of pharmacognosy studies the antioxidant effects of a specific molecule in a newly discovered plant. The molecule can be interpreted as having the qualities of an experimental epistemic thing. When the structural formula of the molecule is mapped and antioxidant effects established, the molecular formula can become part of the technical repertoire of the lab and perhaps also be functional as a stabilised intermediate form (subroutine) in new research objects. From this example, we see that the difference between epistemic conditions and epistemic things is rather functional, depending on the entities' role in the inquiry process. Rheinberger also points out that "whether an object functions as an epistemic of a technical entity depends on the place or 'node' it occupies in the experimental context" (p. 30).

Partly building on the ideas of Rheinberger, Knorr Cetina (1997, 2001b) takes the relational aspect of objects a few steps further. Knowledge and technology are generally regarded by social scientists as fundamental building blocks in contemporary society. However, she argues that few scholars have considered the implications of an increased 'objectualisation' for our core concepts of what is social. Her approach attempts to break up notions of knowledge and expertise to profile object relations as defining expert processes of knowledge. Furthermore, she maintains that "the idea of an object that is relevant to understand these relationships contrasts sharply with our received notions of an instrument, a commodity or an everyday thing" (p. 23). It is the 'binding mechanism' of experts' ties with these objects that makes it plausible to interpret the relations as social forms rather than merely acts of work. Such binding mechanisms are referred to as *libidinal* and *reciprocal*. The former notion is borrowed from Freud's psychoanalysis, describing a stage or phase in individuals' psychosexual development where the 'ego' engages in an object relationship that is especially directed towards an instinctual object (Colman, 2009). However, when applied to knowledge-based activities, the libidinal aspect indicates a somewhat different process. To my understanding, this term is included in the conceptual framework to bring forth the emotional side or desire, constituting relations with knowledge objects. As Knorr Cetina (2001) explains, "the conjunctional or libidinal dimension gives practice a flavour and quality distinctively different from that of routines and habits" (p. 186). This aspect is manifested in her idea of a structure or chain of wanting triggered by knowledge objects. The rationale is thus to designate a dynamic sequence of moves in relation to objects rather than isolated actions and associated motives. When the metaphor is applied to object relations, it denotes the lack of completeness that objects of knowledge display. The signals provide 'triggers' for

the chain of desires postulated. In expert settings, these signals are assumed to be mainly mediated through representations. The representation experts' constructs "tend to imply what is still missing in the picture" (Knorr Cetina, 2001a:185). The representations might thus indicate further development of the knowledge object by mediating insufficiency.

Consequently, how can we define and operationalise knowledge objects? Knorr Cetina (2001b) suggests that the objects that we encounter in our everyday and professional lives increasingly assume features resembling the epistemic objects of research inquiry:

> The definition I want to offer of large classes of objects in contemporary life breaks away from received concepts of objects as fixed things of a material nature. In fact, I want to go in the opposite direction, and characterize the objects relevant here by their indefiniteness of being (p. 528).

The theoretic 'core characteristic' for this class of objects is hence their changing and unfolding nature (indefiniteness). Objects of knowledge in many fields of expertise have material instantiations, but they should concurrently be looked upon as rather unstable and unfolding structures. Opposed to Rheinberger (1997), she proposes that technological objects and consumer goods may also embody these unfolding qualities. In the experimental systems of Rheinberger, technological objects are regarded as stabilised. For example, electron microscopes or LCD displays are commonly seen as fixed and stable entities. Knorr Cetina (2001b) finds this conception problematic as "contemporary technologies are simultaneously things to be used and things in a process of transformation" (p. 528). Computers and computer programmes are applied as typical examples of such dual structures. These technical objects are frequently updated and released in new versions. For example, when I recently purchased a new computer with the latest software installed, it was ready to be used for my current purposes. However, it is likely that the various software drivers included will be subject to further research and development and will soon be obsolete. In fact, I would expect them to change over time as software issues are identified, resolved and released through the automatic updating systems. It is this dual structure of contemporary technological objects which makes it problematic to draw a sharp line between purely technological objects and epistemic objects. However, in the context of software development, Knorr Cetina's conception of knowledge objects might perhaps be illuminating. Let us now look more closely at her contribution to the understanding of experts' object relations.

Through a series of ethnographic studies of scientists, Knorr Cetina (1999) draws attention to the binding role of objects in epistemic cultures. She defines epistemic cultures as "those amalgams of arrangements and mechanisms —bonded through affinity, necessity, and historical coincidence— which, in a given field, make up how we know what we know" (p. 1). The assumption here is that expert subjects tie themselves to object worlds. Knorr Cetina (1997, 2001a) thus puts forward the idea of *knowledge objects* when describing the emergence of social relations in epistemic cultures. These cultures are in essence seen as object-oriented; experts and scientists are often joined in their collective efforts around central objects of knowledge. However, the situated 'knowledge machineries' of late modernity are accordingly not restricted to science laboratories. Epistemic cultures are also seen to infiltrate other systems of expertise and may also be of relevance to many areas of social life in knowledge societies. The above notion of culture indicates a relational account of knowledge, something to be acquired and shared through interaction. Note that interaction in this context can involve both humans and non-human objects. Of significance to the issues of this thesis is the 'matter in between' software developers and their objects. In line with the research aims, my attention in the remainder of this chapter will be focused on the postulated object relations. A key argument made by Knorr Cetina (1997; 2002) is that new kinds of ties are formed between humans and knowledge objects. They are assumed to emerge in knowledge cultures as 'centres' toward which knowledge workers or experts orient themselves. A simple model of the object relation may be rendered as follows:

S <-----> Ko

The *S* signifies the subject and *Ko* the knowledge object. The dotted line and arrows represent the object relation. The idea of reciprocity between S and Ko is indicated with an arrow pointing in two directions. Note that the relation therefore is not directed, postulating a causal nexus bonding the two elements. As I read her work, the tie between humans and non-human knowledge objects does not involve causality in the form of x-->y. On the contrary, the formation of object relation reveals several dimensions. For practical and analytic purposes, I will refer to these functional dimensions as the 'assumptions of the theory'. These are the set of theoretical assumptions or postulates underlying the theory. Some of them are empirically demonstrated, while others remain undemonstrated. However, they are assumed to directly or

indirectly influence the bonding between the two elements, S and Ko. My rationale is to scrutinise the theoretical claims implied in object relations, and at the end of the chapter, define (or sum up) a set of propositions that can be used for empirical analysis.

A principal claim concerning object relations is their *social nature* (Knorr Cetina, 2001b; Knorr Cetina & Brugger, 2002). This indicates an alternative conception of sociality, specific to late modern societies in arguing that "social forms as we knew them have become flattened, narrowed and thinned out" (Knorr Cetina & Brugger, 2002:6). However, it is argued that the flattened social structures do not necessarily lead to a simple decrease in sociality. On the contrary, an expansion of social engagement with non-human objects may lead to a wider social nexus in contemporary life. Accordingly, the social world has been prolonged as more advanced objects are available (i.e., consumer technologies).

Another defining characteristic of relations with knowledge objects is a *structural affinity* between the elements (Knorr Cetina, 2001b). The assumption is that S and Ko have a corresponding form connecting them to each other. On the subject side, the theory suggests the presence of a structure of desire that uniquely matches the open-ended, unfolding character of knowledge objects. As Knorr Cetina (2001b) aptly phrased it, "in a nutshell, the argument is that the incompleteness of being to which I have attributed contemporary objects uniquely matches the structure of wanting by which I have characterized the self" (p. 530). In this sense, both elements (S and Ko) are conceived of as transmutable, "moving entities that provide 'ports' and targets for one another" (ibid.:530). This again involves a semiotic dimension, the existence of some form of communication between the elements. Accordingly, "for the relationship to continue, the object must be signalling what it still lacks, and the subject must be interpreting these signals" (ibid.:530). I assume that this also involves non-linguistic signals; including various visual cues of knowledge objects' conduct or functioning that can be translated into words. However, within this argument lies a theoretical problem as one partaker in the communication is a non-human object. Can the subject meaningfully interpret the signals by empathically taking the position of the object? Will the processes of role-taking or position-taking developed by social theory readily apply to this dyadic relation? Knorr Cetina (2001b:31-32) reflects on this issue with reference to G.H. Mead's role-taking formula for interpersonal sociality. This model is found limited in the context of object relations as we do not have the same familiarity with a molecule or a computer code as with fellow human beings. Nevertheless, the process of perspective-taking may perhaps apply at an

early stage in the relation as a subject somehow needs to comprehend the objects by taking its perspective and figuring out its incompleteness and dispositions. Knorr Cetina (ibid.) recognises the complexity of this exchange amongst intellect and matter and suggests that "the process of position-taking involves the subject's 'becoming the object', a sort of cross-over through which the subject attempts to see the object world from the inside, to 'think' as it does, and feel its reactions" (p. 531).

However, what about the object side of the equation? Is the relation reciprocal in the sense that the object can take the position of the subject? Here, studies of biologists in laboratories indicate that knowledge objects might be internalised and become part of the subjects' mental processing environment (ibid.). When intensively attended to over time, the objects are assumed to 'live in' the subjects' minds, perhaps also unconsciously. Subjects are thus thought to be partakers in the object worlds and vice versa:

> An individual looping his or her desire through an object and back is not only likely to learn something about the object in the process. He or she is also likely to develop a shared lifeworld with these objects (Knorr Cetina, 2001b:532).

This S-Ko dynamic is suggested by Knorr Cetina and Brugger (2002) to develop in contemporary experts' work contexts; for example, those provided by science and technology. They specifically mention software and internet development companies as recent examples. The particular inference made in this regard "is that the self as a structure of wanting becomes articulated in work contexts when the subject has agency in relation to objects" (ibid.:173). It follows that when the work objects of experts' project key features associated with knowledge objects (unfolding structures) and the subject has agency in relation to these objects, social bounds may form. Yet, what role the subjects' agency plays in relation to knowledge objects is not clearly stated. In its wider meaning, the notion of agency implies that we, as analysts, attend to the psychological and social psychological 'make-up' of the actor in order to investigate the capacity for voluntary action (Calhoun, 2002). This is a widely debated problem in social theory. For the current purpose, I merely point to an area of concern, to signify an unexplored facet of relations with knowledge objects. If reciprocal social relations are to develop, one might argue that the object also has agency in relation to the subject. Hence, the objective and subjective aspects of the relations might be bound in mutual agency, so their structures are not clearly distinguished.

Another feature worth considering when assessing the theory is what is referred to as 'solidarity' with objects (Knorr Cetina & Brugger, 2002). Solidarity in social settings usually connotes cohesion among individuals within social groups. However, it is argued that "the knowledge we acquire of non-human things can also give rise to sociality with objects as a form of solidarity with them" (Knorr Cetina, 2001b:532). In this line of thought, solidarity extends to human relationships with non-human objects. An illustrative example can be if a software programmer develops an attitude towards commercial software after learning about the open-source movement from a knowledge object related to her practice. The central ingredient of object-centred solidarity is thus rooted in knowing something about an object. Perhaps a level of common ground or shared prolepsis must be present prior to meaningful interactions with knowledge objects (see chapter 2.1). This concern is also signified by Knorr Cetina (2001b) when recommending that "in order to take an object's position, we must already know something about it, and we extend this knowledge through position-taking and by opening ourselves up for transference" (p. 532).

But how can we identify and interpret relations towards objects of knowledge in empirical analysis? At a basic level, a relation can mean that two or more people or things are connected (Soanes & Stevenson, 2005). However, in the expert practices studied by Knorr Cetina (2005), conventional human bonds are partly replaced with object relations, creating social ties between human beings and non-human objects. Furthermore, she maintains that expertise principally relies on object relations:

> Object worlds make up the embedding environments in which expert work is carried out, thus constituting something like an emotional home for expert selves. As a consequence, object environments define individual identity and situate and stabilize selves. Experts also develop intimate relationships with objects of expertise. They learn to handle and observe them and they also imagine their interior states as they attempt to understand them (p. 2).

Experts such as molecular biologists, educational researchers or software developers are accordingly intimately linked to their objects of expertise. The binding force of knowledge objects is attributed to sequences (or chains) of wanting that are projected onto the objects. Such features render the objects transformative, 'open-ended' and continually generating new

questions. In a keynote speech at a conference, Knorr Cetina (2004) suggests that engagement with knowledge objects can take a few different forms. For example, engagement can vary in time (short/long term) and be personal or collective. This is a matter to consider when choosing foci for analysis. I will return to this in the empirical section. The quality of engagement with objects may also vary in interdependence, level of commitment, intimacy and so on. At a collective level, object-mediated relationships may emerge when collaboration is centred on a joint knowledge object (ibid.). Accordingly, communication among peers in such settings tends to be object-related, and the joint knowledge object commonly exists simultaneously in multiple appearances. Knorr Cetina (2001a:182-183) uses the case of a detector in a high-energy physics experiment as an example. The detector circulates in the collaborative community of experts (as drawings, calculations, written reports, models, etc.). These appearances are only seen as 'partial objects', not fully containing the object itself. For example, a 3D rendering or an early prototype model only shows an unfinished, intended object.

To illustrate the conceptual framework associated with relational knowledge objects, Knorr Cetina and Brugger (2002) turn to the global finance market as an object of attachment for currency traders. Before I sum up and define the set of propositions underlying the theory of object relations, I will discuss a few parallels between contemporary open-source software development and financial markets on screens. One can argue that these collective and geographically dispersed scenarios share similar characteristics. The financial markets of today are, according to Knorr Cetina and Brugger (ibid.), profoundly mediated: "it is a global market entirely exteriorized and embodied on computer screens" (p. 162). As collective systems created in a symbolic or virtual space, conceptualisations of these markets may have a broader appliance in illustrating 'technology-saturated' contemporary expert cultures. Computer programmes are also suggested by Knorr Cetina (2001a:184) as relevant examples of constantly changing and unfolding objects of knowledge. For increased relevance to this thesis, I take the freedom to adapt her example to open-source development. In open-source programming, contributing developers commonly go through cycles of writing software, executing, testing and updating the code. This activity can conceivably be driven by individual developers' own interests. On the other hand, there is a community of other developers and users who also 'orient' themselves towards the code by working on different versions, discussing their features/deficiencies and contributing with updates to the code-functionality, and so on. The software programme under development can thus be interpreted

as displaying properties of a knowledge object in its incomplete, unfolding state. In a similar manner as with the global currency traders, spatial-temporal barriers are crossed in online open-source projects. Advanced technologies in these scenarios allow for various types of interactions (i.e., synchronous and asynchronous communication) with the objects to which they are committed as well as with other participants. Open-source projects may also exemplify the bounding mechanisms associated with relational objects of knowledge. Knorr Cetina and Brugger (2002) argue that "traders not only participate in these markets, they relate to them as a complex 'other' with which they are strongly, even obsessively, engaged" (p. 162). From this perspective, relationships with knowledge objects are defined by the new kinds of bonds that are constructed between humans and non-human objects. Such social ties among the self and others refer to situations in which "interaction, space and even communication appear to mean something different from the accustomed understanding of these terms" (ibid.:163).

Following this reasoning, participation in global open-source projects can be seen as relations towards non-human objects. An intrinsic connection may form between developers and a new programme under development. The programme code displays deficiencies for the developers to interpret and act upon. Moreover, other developers or clients might demand new functions to be included in the programme. Over time, the code evolves and perhaps transmutes into a more attractive or desired version. New lacking and wanting cycles may be initiated by new object varieties in a serial or chain-like structure. This also implies the importance of time when it comes to object relations. A certain commitment over time is assumed a constitutive facet. According to Knorr Cetina (2001b), "this structural equivalence fulfils one condition of a relationship, which is that it should continue over time and not be reducible to a short experience" (p. 530). It follows that a brief interaction with a knowledge object, for instance by consulting an object-centred community online, does not inevitably form sustaining object relations; neither does a closed or finished object that does not embody the related structures of wanting that trigger the subject.

However, there are a few structural features distinguishing the two scenarios described above. Knorr Cetina and Brugger (2002) highlight the computer screen as a crucial component of traders' object relationship with the marked. By doing so, they also call attention to the information systems that it embodies. Drawing on Husserl, it is suggested that the screen is an

"appresentational[3] device that enhances and routinizes such relationships" (p. 163). The notion of appresentation is purposely applied to underscore that screens as representational devices do not simply render an external reality, but are rather constitutive of it. When it comes to contemporary software development, it is common to make use of a set of integrated development tools to support the process of writing software. The majority of these are probably networked and rendered on computer screens. However, the screens themselves are hardly the only gateways to connect software developers to their distributed software objects, neither are the information structures they display. I find it more reasonable in this context to regard the integrated set of artefacts as 'devices' for forming and sustaining object relations. Hence, the framework for conceptualising software developers' engagement with objects of knowledge may perhaps be supplemented with notions of integrated meditational artefacts (see chapter 2.2).

Based on the review and discussion above, I sum up a few propositions concerning object relations that are relevant for the research conducted as part of this thesis. My intention here is to provide a set of propositions that can be useful in organising concepts for empirical analysis. The following propositions form an interrelated set of definitions and assumptions and should thus be seen in relation to each other.

i. The objects towards which knowledge experts tend to orient themselves in their creative practices are characterised by incompletion. As unfolding processes, they are not equivalent to stabilised structures of instruments, commodities or everyday things.
ii. Object relations involve a semiotic dimension or some form of communication between the elements (S and Ko). For a relation to be sustainable, the knowledge object should signal what it lacks and the subject must be able to interpret these signals. This may involve a process of position-taking as the subject needs to understand the object's conduct and figure out its deficiencies and dispositions.

[3] The term *appresentation* designates an indicative function of something that is not directly presented, i.e., a sign showing a protective helmet at a construction site may indicate a referential structure of use (Russell, 2006).

iii. The deficiencies communicated (or signalled) by unfolding knowledge objects provide 'triggers' that match experts' desires. These signals are generally mediated through representations, indicating further development of the object.

iv. Joint knowledge objects in collaborative expert communities commonly exist simultaneously in multiple appearances. These appearances are only partial as they are not fully containing the object itself.

v. A certain commitment over time is assumed a necessary prerequisite for an object relationship to form. An isolated short experience with a knowledge object is not considered sufficient to be included in this definition of object relations.

vi. Social bonds may form when the work objects of experts display the characteristics of knowledge objects and the subject has agency in relation to these objects.

2.3.2 Reflections on the ontology of knowledge objects

The following paragraphs reflect on the ontological assumptions implicit in frequently cited object-related theories within this thesis, paying particular attention to the notion of knowledge objects. In applying the concept of ontology, I refer to the set of entities whose being is acknowledged by these theories (Lowe, 2000). I also reflect on the role of multiple representations in the ontological structure of the objects asserted. The case of open-source development is applied to illustrate how the entities anticipated by these theories depend on relations and multiple representations rather than underlying substances.

According to Packer & Goicoechea (2000), epistemological questions are concerned with theories of knowledge "when knowledge is valid, what counts as truth, and so on" (p. 227). Ontology, on the other hand, deals with more fundamental problems such as identifying the kinds of things that exist as well as their nature or essence (Glock, 2003). Lash (2001) maintains that "in phenomenological engagement, we make sense less through logic and epistemology than through ontology" (p. 107). In our quest for meaning

> the neutral and detached space of the scientific observer can yield epistemological knowledge, as Kant noted, of the *appearances* of things — i.e., cause and effect,

> explanation. But experiencing things, through being in the life-world with them, can open up knowledge of *things-in-themselves*. To know things-in-themselves is to know them not epistemologically, but in their ontological structures (ibid.).

Thus, by engaging with things in the world, we can perhaps reach an understanding of their ontological structures. But what about the 'virtual worlds' that can only be accessed through mediating technology and various representations? I will return to this problem later in the discussion. First, I explore the basic assumptions behind an 'unfolding ontology' of objects proposed by Knorr Cetina (2001a) to emerge within knowledge-intense practices. The problem is rooted in the postulates concerning objects of knowledge (see chapter 2.3.1). When considering the 'being' of knowledge objects, such as how they exist, what their conditions of existence are, and on what relations they depend, I find it practical to focus on a few key tenets used in their postulation. The categorical scheme underlying the theory does not appear in a hierarchical structure, nor does it exhibit a strict division between abstract and concrete entities. However, some key aspects are suggested: *open-endedness*, *unfolding structures* with the capacity to form *social relations* and *existence through multiple representations* (Karen Jensen & Lahn, 2004; Knorr Cetina, 1997, 2001a, 2004; Knorr Cetina & Brugger, 2002). These are illustrated in the model below.

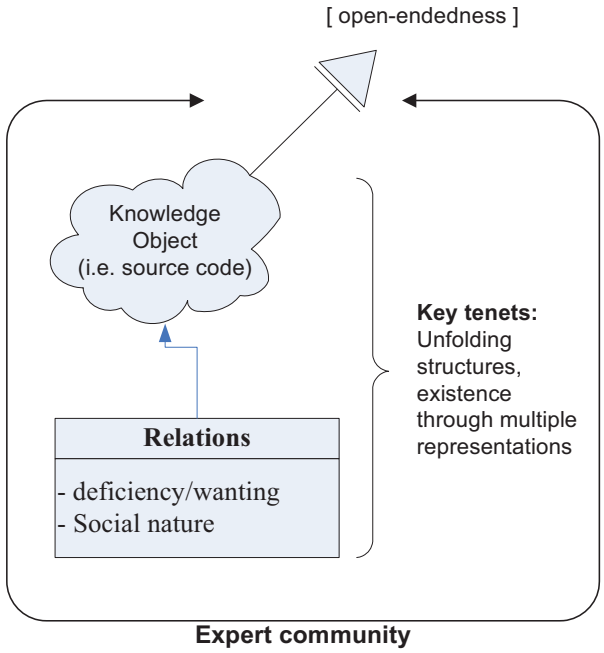


Figure 3: Key tenets of knowledge objects

What are the ontological assumptions underlying these formulations? The assumptions above do not reflect a harsh dualistic distribution of mind and matter. The wish to avoid dualisms has motivated much contemporary work on this intricacy. However, the theory at hand apparently steers clear of this problem by postulating objects as emerging structures; something both 'is' and is also in the process of 'becoming'. Furthermore, the advanced objects are said to be 'non-identical' to themselves and continuously transforming. The unfolding process is thus not considered in terms of underlying substances, but rather as constituted by relations. This implies an ontological commitment to processes and relationships rather than predetermined entities in a physical space.

In an associated mode of thought, studies applying Actor Network Theory (ANT) are often coupled with a relational approach that discards a cognitivist notion of objects/actors as meaningful entities prior to any involvement in semiotic networks of interactions (see e.g. Hernes & Czarniawska-Joerges, 2005; Law, 1992). However, within this perspective, a

symmetrical treatment of human and non-human elements in networks raises basic epistemological concerns, while also reflecting an ontological commitment. Latour (cited in Packer & Goicoechea, 2000) calls upon a network-like ontology in which actors are not regarded as fixed entities but as "new ontological hybrid, world making entities" (p. 232). In this context, he argues that "the old distinction between things and representations, between material and texts, is dissolved: Both have the same ontological status" (ibid.). Yet, when it comes to the ontological structures of technological practices and the infrastructures of virtual worlds, multiple representations may play a vital role. The online arena for open-source development is commonly regarded as a distributed or virtual community in which human interaction is made possible by internet technology. The source code is visually represented in multiple forms and rendered on computer screens. According to Knorr Cetina (2001a), knowledge objects flow through expert communities in multiple forms (or in her words, 'instantiations') ranging from figurative to material realisations.

Possible representations of such objects in a software development community are design patterns in programming, logical flowcharts and ongoing documentation of a project. The objects distributed in the community do not necessarily represent physical entities 'out there', but rather are abstract in form. For instance, the code of an open-source project can be represented in a series of instructions written by a programmer by means of a compiler or interpreter. The source is commonly written in a particular code language (i.e., java or c++) that is comprehensible by human beings. Source code must be further converted to object code (or machine language), however, in order for computers to be able to read and execute the programme. Developers thus rely on inscription devices and multiple representations in their interaction with the source code. Therefore, one can reasonably argue that the bits and bytes stored and distributed electronically do not appear as ontological structures without the aid of multiple representations (e.g., programme language, compilers) and relations to one or more interpreting actors.

In sum, knowledge objects seem not to rely on underlying substances, but rather are constituted by their relationships. This implies an ontological commitment to processes and relations rather than predetermined entities in a physical space. As objects, they are rather abstract and may emerge as ontological structures through multiple representations (i.e., programme language, compilers) and interpreting actors.

2.4 Epistemic trajectories and associated events

The term *trajectory* usually describes some kind of path or progressing line of movement. In *The Oxford Dictionary of English*, a trajectory is referred to as "the path described by a projectile flying or an object moving under the action of given forces" (Soanes & Stevenson, 2006). Applied as a metaphor guiding empirical analysis, it may, according to Ludvigsen et al. (in press), offer "levels of description that cut across different traditions and make learning across sites more transparent" (p. 15). When it comes to professional learning and the analysis of object relations across different contexts separated in time and space, notions of trajectories can perhaps be illuminating.

A trajectory model for social studies was pioneered by sociologist Anselm Strauss in collaboration with Barney Glaser through a series of studies (Glaser & Strauss, 1965, 1968). These studies paved the way for accounting for different movements in time and space as significant aspects of social organisation. In the course of the last decade, different theoretical perspectives and research agendas on trajectories have flourished within the learning sciences. For instance, Lave and Wenger (1991) introduced the term 'trajectory of participation' to reveal the paths of movements in communities of practices. It frames the understanding of newcomers' transition from a legitimate peripheral participant to full member in communities of practices. They thus imply an interactive process in which the individual learner starts out by 'orbiting' the actual practice of experts. The concept of trajectories can also be used as a hypothetical construct; i.e., to aid teachers choosing an instructional design based on anticipations concerning how individuals' learning may progress.

Of particular relevance for issues explored in this thesis are the spatial-temporal dimensions of a trajectory. Learning and meaningful interaction with knowledge may occur when the localised problem solving of software developers intersects with global projects focused on joint knowledge objects. The temporal dimension indicates a variety of engagement among different actors in time. Ludvigsen et al. (in press) review related concepts that have been used to understand the temporal aspects of trajectories of participation in learning and concludes by recommending both a horizontal and vertical approach to analysis:

> Across the different accounts, there are arguments for a combination of a vertical in depth analysis of moment-to-moment interactions and a horizontal perspective to

> include more longitudinal timescales that are made relevant in the moment. This is because culture and artefacts have time scales other than those of immediate interaction (…) an important analytical endeavour is to investigate the ways in which participants make use of historical and cultural knowledge represented in artefacts to understand what constitutes the content and processes of learning (p. 5).

Let us consider the artefact example from chapter 2.2; Integrated Developer Environments (IDEs), the platforms commonly applied by today's software developers. Key features of an IDE are the ability to edit and execute programme code. Following the analytic request above, one can embark on investigating how developers make use of the knowledge embedded in the artefacts in their daily (moment-to-moment) problem solving. However, applying trajectory as an analytic concept also implies seeing beyond the short term interaction and accounting for temporalities that are brought into and made use of in the situation. The IDE can be seen as a cultural/historical artefact reflecting years of development along several intersecting trajectories. As such, it represents vast amounts of accumulated expert knowledge on how to create professional applications within the domain of programming. Hence, intersecting trajectories of participation can be traced both in relation to the evolving artefact or different objects of activity depending on the foci of analysis. Furthermore, we can say that social processes associated with the transformation of knowledge within a specific domain of software development are partly mediated by the integrated set of artefact. However, if we isolate an interactional sequence with such a complex tool, the developers are likely to only be exposed to a small part of the accumulated knowledge embedded in the artefact. The artefact may, as Ludvigsen et al. (in press) recommend, "be understood as a product of multiple timescales" (p. 10) adding knowledge and experience throughout its development history. But, what happens if artefacts acquire properties of knowledge objects? Perhaps it would entail moving the analytic lens towards the relational aspect; how actors connect and relate to object-centred communities distant in time and space.

When giving an account of how to analyse trajectories, I assume that a 'temporal' unit of analysis must be stated. Work-related studies and studies in education targeting learning trajectories or trajectories of participation range from analysing moment-to-moment interactions to longitudinal career patterns stretching over decades (Lahn, in press). However, for the purposes of analysing relations to distributed knowledge objects, it may be useful to look into shorter phases or episodes. For the current purpose, I will refer to these as

associated events; timeslots that are specified by a few knowledge object-related dimensions involved in triggering actors' connections or movements towards knowledge objects. Seen in relation to epistemic trajectories, the events can be coupled with partly relevant knowledge acquired in the past. This means that knowledge-laden trajectories i.e., from previous experience with the task at hand, may intersect and possibly influence the event. Furthermore, the associated event is defined by its relation to a knowledge object that indicates trajectories for future development; how to move on from the current state of the object or how it can be used to influence a set of future events (i.e., creative problem-solving detached from the original knowledge object). Knowledge sources applied 'in-situ', but which are not displaying the open-ended characteristics associated with knowledge objects (see chapter 2.3.1), can be seen as *supplementary sources* outside of, but not completely detached from, the associated event.

When assessing the usefulness of trajectories for the analytic purposes of this thesis, the rough classification of learning trajectories provided by Lahn (in press) may serve as a point of departure. His typology is rendered state-of-the-art by sorting different perspectives on learning trajectories. The typology is represented in a compact version below:

Trajectory type	**Distinguishing attributes/qualities**
Educational or didactic learning trajectories	Derives from cognitive psychology. Primary metaphor of learning is assumed to be the skills-acquisition model. Progression along trajectories is characterised by "individual learning of a new task or differences in strategies between novices/experts" (p. 4). Levels of analysis are individual learners engaged in problem-solving. Timescales ranging from short episodes to longer intervals.
Lifelong learning trajectories	Also referred to as 'learning careers'. Applied in research on adult education and personal development. Associated with a participation metaphor of learning. Grounded in narrative theory and personality

	psychology. Process characterised by personal reconstructions of life episodes. Levels of analysis can be individual learners in context, lifelong learning histories and transitions across lifecycles. Extensive timescales.
Community learning trajectories	Also referred to as 'organisational trajectories', a term connoting "ladders in occupational careers and more horizontal moves that turn newcomers into proficient professionals" (p. 3). Theoretically rooted in social theory and cultural studies. Coupled with a knowledge creation metaphor of learning. Progress along trajectories is characterised by "collective sense-making of past, present and future events" (p.4). Emphasis on interaction and socialisation into professional communities. Variations in timescale.
Epistemic trajectories	Novel category suggested by Lahn (op.cit.). Partly derived from the new sociology of knowledge and systems theory. Associated with a knowledge creation metaphor of learning. Epistemic trajectories are characterised as 'knowledge development' in relation to mediated expert networks (p. 4). This type of trajectory is assumed 'linked' to object-centred epistemic communities distributed in time and space. They represent flexible, knowledge-intense paths, mediated by new technologies. Dynamic and intersecting timescales (the dimensions of time and space may be experienced as fused along these paths). A network-level analysis is recommended.

Table 4: A typology of learning trajectories adapted from Lahn (in press:4)

The above table is a slightly condensed and adapted version of the original typology. The didactical implications and methodological preferences associated with each category are omitted due to limited clarifying relevance for the purposes of this thesis.

As indicated above, contemporary empirical studies focused on participation trajectories, learning trajectories and the like, often explore participation patterns and knowledge development over time. From Lahn's (in press) review of different perspectives on trajectories, we see significant variations in scales; ranging "from students' performance on school tasks to life course transitions or cultural typifications of collective experiences" (p. 18). In addition to the spatiotemporal scaling associated with the analytic levels of empirical studies focused on trajectories, Lahn also addresses a third dimension, which is *content*. In order to better understand the movement of knowledge content in expert communities and not only the tracing of individuals' learning along trajectories, he suggests applying the notion of epistemic trajectories. With reference to Stutt & Motta (1998), Knorr Cetina (1997, 1999) and others, it is argued that a modern working life is characterised by epistemification and that the issue of knowledge has to be reintroduced into research on professional learning. As Lahn (in press) puts it, "there is a need to preserve the concept of epistemic trajectory in order to remind ourselves that issues of knowledge content and its dynamics are important when studying expert communities" (p. 23).

I consent with Lahn's overall argument concerning the importance of knowledge content. However, it is not obvious what knowledge content is when studying expert or professional communities. Is it the abstract/theoretical side of the continuum, the scripted/procedural side or the knowhow inscribed in conceptual artefacts? To narrow the focus and provide operant tenets for empirical analysis, I will suggest *fragments of knowledge objects* as the content characteristic for this particular class of trajectories. My assumption as the thesis progresses is that distributed knowledge objects move along trajectories in small parts or fragments, broken off and spilled over from the mediated expert community in which they first appeared. A hypothetical example can go along these lines:

> Edith is a consulting software developer currently working on an intranet solution for a client. The last couple of days she has received error reports concerning the access to database from the client's intranet. Her initial trial-and-error approaches hit nothing

> but dead ends. The problematic function is based on XML technology with which she has only limited experience. She thereby presents the problem to Adam, a competent team member. He suggests checking out an open-source project called Hypergate. They have worked out a similar problem and shared an XML script. Adam is kind enough to e-mail the link to the project.

This exchange can be interpreted as movements along a set of intersecting trajectories. Imagine that this shared XML script and attached documentation is collected by Edith. How are the sequences of events triggering this movement and subsequent transformation of knowledge described? The shared lines of code are taken out of their original context, adapted and reused to solve her local database problem. The movement of functional code across locations can be interpreted as chunks of knowledge following a trajectory, initiated by an associated event. In this example, the trajectory goes through her colleague, via e-mail, and to a mediated expert network. She might do some browsing around the community site before the proper XML script is located. The collected 'chunk' is part of a larger knowledge object, the overall Hypergate project. It is transferred across time and space to be further modified by Edith and her colleagues and integrated into their client project. However, more detail can be added to observe the actual knowledge content of this trajectory.

How are epistemic trajectories purposefully represented? Is it possible to construct a map of software developers' typical movements towards distributed knowledge practices? This may involve looking into the obligatory *passage points* along the trajectories (or the nodes in network terminology). My underlying assumption is that knowledge objects developed in expert networks may 'spill over' and act as resources for local problem solving. This means that when the partial objects move along rather flexible trajectories across contexts, they can transform and acquire different attributes depending on the dyadic relation. In a local context, the knowledge object may perhaps intersect with individual actors' problem solving and transform into a conceptual artefact or resource, whereas in the expert network, it attracts and binds globally dispersed developers towards an open-ended knowledge object.

In "Allegory and its others", Singleton and Law (2003) scrutinise techniques for mapping trajectories. However, they also reveal complex interference problems that have to be dealt with. One may face what they refer to as "an ontological commitment to singularity and the realism of out-thereness, combined with the epistemological frustration that comes with

representational failure" (p. 237). Accordingly, there are more in between the lines than what is said. When investigating how typical patients diagnosed with alcoholic liver disease move through the hospital system, they find that trajectory maps are restricted when it comes to seeing "the hidden stuff of allegory" (ibid.: 246). The consequences are that there is something that can hardly be represented by trajectories. It cannot be meaningfully drawn or mapped. For instance, when interpreting how specific parts of a trajectory (i.e., passages or nodes) are related to contexts or objects separated in time and space, there may perhaps be meanings that are hidden from the visible structures. Do unseen emotional or affective processes run parallel to the structured system of trajectories, representing different levels of meaning? The metaphor of allegory thus reminds us to read between the lines and to be sensitive to hidden meanings when interpreting trajectories.

Based on the review and discussion done by Lahn (in press) and Knorr Cetina's knowledge objects (chapter 2.3.1), I sum up by suggesting a set of key tenets characterising epistemic trajectories. As the previous discussion suggests, concepts of epistemic trajectories and knowledge objects can perhaps have analytic power in understanding movements and relations within systems of expert knowledge. The perspectives embrace interrelated definitions and relationships that can potentially assist and organise the empirical analysis. Theoretical assumptions, or idealised parameters, associated with the phenomena of epistemic trajectories are summed up in the following:

i. This class of trajectories represents knowledge-intense paths that are mediated by new technologies.
ii. Movement along the paths is triggered by associated events. Intersecting trajectories within the event incorporate potential forces leading the knowledge seeker towards knowledge objects located in mediated expert networks.
iii. An epistemic trajectory stands in relation to object-centred expert communities distant in time and space. They have dynamic and intersecting timescales. The dimensions of time and space may be experienced as fused along these paths. These features are attributed to the application of asynchronous and synchronous mediating technologies.
iv. The moving of knowledge content along the trajectories can be partial knowledge objects 'spilled over' from the expert communities in which they first appeared.
v. Along epistemic trajectories are "openings" or invitations for participation in the epistemic communities; for instance, a hyperlink or button pointing to a form for

signing up as a community member. Mediating artefacts may thus offer social affordances for participation in object-oriented expert networks.

vi. The movements along epistemic trajectories can be traced across time and space by following parts of knowledge objects (the knowledge content) through mediating artefacts.

vii. Epistemic trajectories can partly be rendered visible by identifying the forces and restrictions influencing their passage points or nodes.

3 Summary and proposed conceptual framework

This chapter sums up the previous discussion and suggests a conceptual framework to inform empirical analysis in line with the research aims. The framework outlines a set of concepts and propositions to guide and inform systematic inquiries into software developers' object-related practices.

Based on the discussion of objects and practice, a few propositions concerning object relations stand out as central. They mainly draw on Knorr Cetina's ideas on knowledge objects in expert communities. In short, *objects of knowledge* are characterised by their dynamic, unfinished and open nature; as the centres of attention in experts' creative practices. The formation of *object relations* is assumed to involve a semiotic dimension in which the knowledge object signals its shortcomings. These signals are mediated through multiple representations, indicating further development of the object. A *structural affinity* or corresponding form connects the elements in the relation, providing ports and targets for one another. Commitments over a *time* period and subjects' *agency* in relation to the objects are considered essential parameters for object relations to form.

When exploring the potential of *key artefacts* in the field of software development to mediate relations to objects of knowledge, it can be useful to describe the integration of artefacts/resources in software development. It is indicated that developers may rely on sets of integrated computational tools (software environments) that do not easily fall into the categories discussed. In the reviewed perspectives on artefacts and mediation, we find a relative consensus that artefacts play an important role as resource and mediators in productive practices, and the knowledge and experiences which they embody are things we can study and from which we can learn. Artefacts may serve as resources for current activities and be functional as reifications of ideas and carriers of knowledge from a specific culture. However, the different levels' artefacts suggested by Wartofsky (1979) may be problematic to distinguish and perhaps operate in a more 'blended' or integrated way when it comes to networked, computational artefacts. Recent developments in sociocultural theories on the relation between humans and objects can perhaps complement more traditional notions of artefacts. However, current research literature does not provide adequate insight into how artefacts mediate relations to distributed objects of knowledge, and how the latter intersects with the localised problem solving of software developers. In order to pursue the first research

aim, namely to explore the potential of key artefacts to mediate relational properties associated with knowledge objects, notions of trajectories are probed. The category of *epistemic trajectory* suggested by Lahn (in press) can be a useful addition to the analytic framework in order to conceptualise how artefacts applied as knowledge resources in software development are linked to object-centred communities distant in time and space. Epistemic trajectories are characterised as knowledge-intense paths in relation to mediated expert networks. Moreover, this allows us to reflect on the knowledge content moved along the trajectories. I put forward that the content can be seen as partial objects of knowledge derived from networked expert communities.

For the descriptive purposes of the second aim stated in the introduction, *a network-level analysis* is considered relevant. An open-source project can be interpreted as a network of actors oriented towards a joint object of knowledge. By incorporating a descriptive network analysis, we open up the possibility of rendering the overall network visible and identifying central and peripheral sub-groups of participants. In this manner, contrasting sub-groups can be recognised for analysing variations in how processes of intersubjectivity evolve around shared objects' development in a distributed community. However, to render the network structure visible and identify different sub-groups of participants, more formal descriptions are necessary. For the latter purpose, formal methods associated with a Social Network Analysis (SNA) and computer-mediated communication is investigated in the chapter on specific methods and research techniques (chapter 6.3).

As pointed out in the theoretical review on *intersubjectivity*, potential new entries for exploration in Rommetveit's framework are processes of intersubjectivity constituting object-oriented activities in mediated contexts. Introducing the spatial-temporal dimension of intersubjectivity allows us to consider the tacitly assumed or pre-supposed premises of shared knowledge. The adoption of Rommetveit's framework to account for mediated intersubjectivity can be useful for analysing how intersubjectivity evolves around shared objects in distributed contexts. From this angle, we move the analytic lens beyond participants' subjective realities to account for what is already known (or assumed) and the potential meaning making around shared objects in joint activity. For this purpose, the notion of *prolepsis* can be of analytic value. The use of challenging presuppositions in the form of prolepsis, as explained in chapter 2.1.2, might be a highly relevant notion for analysing shared understandings around joint knowledge objects in expert communities. Proleptic instances

may serve as catalysts for sense-making in challenging the receivers of incomplete utterances and trigger the construction of new understandings. However, as I interpret Rommetveit, this is not an absolute requirement for constructing intersubjectivity; but such incidences may enhance the progression towards a greater intersubjectivity. By omitting information and providing minimal cues, the issuer invites the receiver to participate in the co-construction of an expanding intersubjective state. Yet, the role of mediating technology is not attended to in this framework for intersubjectivity. The extended theory of Clark and Brennan is argued to be a supplementary approach for understanding how intersubjectivity develops around joint knowledge objects in a networked, technology-mediated context. Clark and Brennan's notion of *grounding* suggests that processes achieving shared understandings are qualitatively different in mediated forms of communication compared to face-to-face conversations. In the mediated communication among open-source software developers, it is reasonable to assess potential resources, costs and constraints on processes of intersubjectivity associated with the main communication channel applied in the project. Finally, the *grounding criterion* and *principle of least collaborative effort* can help us to attend to possible changes in participants' grounding techniques and what needs to be jointly understood in relation to a moving object of knowledge.

4 Overall research design

This chapter describes the pragmatic aspects of the overall research design as well as its rationale. The question of how to conduct a systematic inquiry in order to obtain empirical evidence for the phenomena comprising the aims and ultimately what inferences can be drawn from them, will be addressed. The particular methods and research techniques undertaken in each of the two cases will be described in further detail in the empirical sections (chapter 5 and 6).

There are quite a few factors to consider when adopting a functional research design in order to pursue the aims stated in the introduction of this thesis (chapter 1, p. 3). It is to be noted that a substantial portion of the empirical data for the first explorative case is collected jointly in the ProLearn project. When I joined the project in the autumn of 2004, the ProLearn project was in its early phase. Approaches to data collection were planned and refined during project meetings. The data collection strategy stated in the original project proposal was still open to slight modifications (Karen Jensen & Lahn, 2003). However, as we were several researchers with our own interests, consensus decisions had to be undertaken. Moreover, we had a reporting commitment to the Norwegian Research Council and project milestones to pursue. A balance between participants' research interests, commitment to the project proposal and practical considerations with regard to time and resources, resulted in the following methods of inquiry for the ProLearn project: In-depth semi-structured interviews, focus group interviews, self-reported learning logs (or diaries) and document analysis. Each of the data collection procedures is described in further detail in chapter 5.2. It is important to note that the overall research design and units of analysis of the ProLearn project are not congruent with what is adopted in this thesis. The foremost rationale guiding data collection in the ProLearn project was to supplement and elaborate a longitudinal survey (StudData) in the context of postgraduate employment (ibid.). 10 individuals from each of the four targeted professions participated as respondents, and multiple sources were obtained at different points in time (two and five years of working experience). However, in order to pursue the research aim of this thesis, priorities are made within the ProLearn database. One individual case is selected following a purposeful sampling process (Gall, Borg, & Gall, 2003). This means that multiple sources from one software developer are purposely selected for a more detailed case study. Selection criteria are thus based on richness of data, field notes and general impressions

regarding from whom among the respondents we can learn the most. The case study approach adopted and the specific borders for exploration are further described below.

As indicated above, the first case study comprising the empirical portion of this thesis primarily utilises data sources collected jointly in the ProLearn project, whereas the second case draws solely on sources gathered for the specific purposes of this thesis. Now, the two empirical studies are thematically related, but the context and analytic centre of attention varies between them. Consequently, data collection strategies for the two studies also diverge. Considering the use of multiple sources of evidence and the nature of the social phenomena focused upon, it is reasonable to adopt *a case study approach* as an overall research design (Stake, 2005; Yin, 1994). The general research purposes for the two studies are aimed to develop an understanding of complex social phenomena (object relations and intersubjectivity) by combining an *emic* and *etic* perspective. The former intends to describe a participant's viewpoint or "an insider's perspective" on the targeted knowledge practices (Calhoun, 2002; Gall, et al., 2003). It refers to the reflexive self-reports given by participants within the practices. From the latter viewpoint (etic), the practices are accounted for by using the conceptual framework adopted in this thesis. I thus also maintain my own interpretive perspective as an outside investigator of the phenomenon. I am aware that this distinction can be challenged with regard to the objectivity of my observations and descriptions. However, the emic/etic perspectives are only intended to be rough categories to separate different levels of analysis; to distinguish participants' descriptions from my interpretations of the phenomena. In the following, I describe the common case study research design framing the studies. Distinctive data collection methods are presented and discussed as integrated parts in each study.

4.1 A case study approach

As Eisenhardt (1989) notes, the case study methodology can be seen as a comprehensive research strategy aimed at understanding "the dynamics present within single settings" (p. 532). According Yin (1994), within this methodological approach, one might focus the inquiry on either single or multiple cases and several levels of analysis. Single cases can be targeted because of their uniqueness and multiple cases in order to search for generalised themes and patterns. The following quote from Stake (1995) also indicates a genuine interest in the *unique* and *particular* as the hallmark of this approach:

> A case study is expected to catch the complexity of a single case. The single leaf, even a single toothpick, has unique complexities — but rarely will we care enough to submit it to case study. We study a case when it itself is of very special interest. We look for the detail of interaction with its context. Case study is the study of the particularity and complexity of a single case, coming to understand its activity within important circumstances (p. 11).

Yet, in the literature on research design in the social sciences and educational research, the case study approach has been conceptualised in diverse ways. For example, Yin (1994, 1997) advocates that a case study is an empirical inquiry that is focused on a contemporary phenomenon within its real-life context, and for which the boundaries between phenomenon and its context are not clear. Gall & Boerg et al. (2003) define case study research as a "the in-depth study of instances of a phenomenon in its natural context and from the perspective of the participants involved in the phenomenon" (p. 125). From the above, we notice that the study of *phenomena in context* is central characteristics associated with this approach. I assume that the nature of phenomena referred to depends, in part, on the researcher's foci for analysis. Gall & Boerg et al. (ibid.) note that a case study is primarily conducted to shed light on phenomena; "which is the processes, events, persons, or things of interest to the researcher (p. 125)". Accordingly, when the phenomenon of concern is clarified by the researcher, the case (or cases) for intensive study can be selected. A case can then be seen as a particular example of a phenomenon, but as I understand it, not fully comprising all aspects of the real events or social processes investigated. Phenomena can thus be seen as the appearances of such social processes. This is, to some extent, in line with the metaphysics developed by Kant (1996), in which phenomena are regarded as objects and events as they appear in our experience. It follows that phenomena are of a different nature from the objects or events themselves. In this line of thought, phenomena are shaped and interpreted by our cognition (Blackburn, 1996). Actually, we can go much further with regard to this matter by investigating the consciousness involved when experiencing and theorising phenomena, drawing on the philosophical method of phenomenology; although, this would be far beyond the limited format of this thesis. For the current purpose, I maintain that the case study approach is appropriate for studying aspects of the social phenomena of concern using multiple sources of evidence. In the two studies comprising the empirical section, cases are selected to represent the phenomena of interest and data are collected from several sources.

However, as any of the phenomena may have many aspects, according to Gall & Boerg et al. (2003), as researchers we must consciously select a focus for the investigation. The focus is here understood as "the aspect, or aspects, of the case on which data collection and analysis will concentrate" (p. 125).

When studying object relations and processes of intersubjectivity in the context of software development, the case study approach can thus be purposeful in allowing us to direct the focal point on a few specific aspects, events or situations representing these phenomena and explore them as interactive, social processes. To paraphrase Bell (2006:8), these processes may remain hidden in a large-scale survey but can be the essential glue of social systems or organisations. The above research aims indicate specific aspects for exploration, namely the relational dimension of knowledge objects in Study 1, and shared prolepsis in Study 2. The latter describes an aspect of intersubjectivity; an anticipated prerequisite for intersubjectivity to develop among participants in the mod_perl project. It is part of a common understanding that is presupposed or taken for granted by participants prior to interaction (see chapter 2.1.1 and 2.1.2).

We have seen from the above that inquiries following a case study approach vary in their design, ranging from in-depth studies of single cases to large-scale multiple cases. Stake (2005) categorises these into three types: intrinsic, instrumental and collective/multiple case studies. An *intrinsic* study aims to understand the unique and particular. Inquiries are undertaken to gain deeper understanding of a single case that may be potentially interesting and illuminating on its own (i.e., by studying a unique social event). Theory building, understanding abstract constructs or phenomena, is not the overall intention. On the other hand, the term *instrumental* is used for case studies where "a particular case is examined mainly to provide insight into an issue or to redraw generalizations" (Stake, 2005:445). It is about the illustration of an issue (or several issues). The case in this manner is secondary; it has an instrumental role in exploring and understanding an external issue or research interest. As with the intrinsic study, the case is nevertheless examined profoundly. This may, for example, involve scrutinising in detail its activities/functioning, physical/social context, applied artefacts and historical background. The justification here is to peruse the issue or external interest. The case selection hence depends on its value in order to advance the understanding of the issue. However, Stake (2005) maintains that there is no hard and fast line distinguishing intrinsic case study from instrumental, "but rather a zone of combined purpose"

(p. 445). In multiple case studies, the instrumental aspect is extended to several cases. More than one case is accordingly applied to study a phenomenon, a set of cases are analysed collectively. Yin (1994) refers to this strategy as "multiple case study" (p. 39).

The three types of studies described above can be useful guidelines when reflecting on the purpose of the current study and bounding the cases. I also keep in mind that there is, at present, no universally accepted protocol for designing cases. Stake (2005) points out that a case study is, in principle, *a choice* of what to study; "as a form of research, case study is defined by interest in an individual case, not by the methods of inquiry used" (p. 443). It is rather an analytic process within a specific bounded system over a time period "through detailed, in-depth data collection involving multiple sources of information and rich in context" (Creswell, 1998:61). However, unique cases might be coupled with some limitations, restrictions or borders for exploration. Bounded systems are to be studied within a specific context. The borders or bounded system defining the case in studies of social sciences, educational science and the like, can hardly be rigid and unconditionally enclosed. Some outside features may be significant for understanding the context of the case. For example, the contextually embedded local problem solving of a software engineer may impact, interfere and/or cross the borders to learn from the systems of other developers. Preliminary findings in the ProLearn project indicate that single developers' local problem solving is linked to distributed knowledge practices operating on different time-place scales. It is not straightforward to state where the work context of single cases (individual developers) in the ProLearn project starts and ends. Furthermore, various networked artefacts seem to be intertwined in the problem solving loops of programming, stretching beyond the spatial-temporal relations of physical context. This indicates that the case boundaries of individual developers are rather networked and perhaps *selective-permeable*. Here, I use an analogy from biology, where specialised cell membranes allow the passage of only certain substances, selected on the basis of the size and usefulness of molecules (Allaby, 1999). In a similar manner, when defining the case borders, it can be useful to let certain features from the external environment pass through the boundaries enclosing the case, e.g., based on their significance in answering the research questions. The features that are selected within the system and which are considered outside will be further defined in the case descriptions (see chapters 5.1 and 6.2). I am aware that this approach raises the epistemological question of what can be learned from a single case. Stake (2005) encourages researchers to design their inquiry for optimal understanding of the case rather than searching for generalisations outside

its boundaries. For the cases presented in this monograph, this may involve situating the case within their physical and virtual spaces.

According to Yin (1994), a case study's research design typically includes the research questions, unit of analysis, propositions (if any) and the reasoning connecting these to the empirical data and finally, some criteria for interpretation. However, the procedures for reporting on case studies are currently not standardised and the methods for assembling data are, to a certain extent, determined by access to, and acceptance by, the subjects (Marshall, 1998b). Following Yin (1994) and others, a more coherent feature of case study research is the practice of *triangulation*; a form of cross-checking by using several different sources to validate findings. The sources can, for instance, be different methodologies and data, several interpreting investigators and multiple theories. The idea is that propositions and interpretations will be strengthened when they are confirmed by different sources. Examples of diverse sources to be triangulated in case study research are: participant observation, unstructured/structured interviews, focus groups, documentary data, physical artefacts, surveys, archival records, etc. (Patton, 2002; Phillips, 1992; Yin, 1994). To examine the cases in detail, Yin (1994) categorises two basic types of data analysis: holistic or embedded. The former refers to when the entire case is studied in detail and the latter to when what is studied is narrowed down to specific aspects of the case. Furthermore, a choice has to be made by the researcher(s) as to whether a single case study or multiple cases will be functional when attending to the research questions. The rationale behind such a decision can be the representation of an extreme or unique case or an explorative device. If the study does not rely solely on existing theory, but seeks to discover features of the phenomena by partially approaching the field inductively, it might be considered as explorative (ibid.).

To structure the inquiry process, it can be helpful to follow the major conceptual responsibilities for case study researchers, issued by Stake (2005) as guidelines. These are summarised as follows (p. 459):

a. Bounding the case: conceptualising the object of study
b. Selecting phenomena, themes or issues (i.e., the research questions to emphasise)
c. Seeking patterns of data to develop the issues
d. Triangulating key observations and bases for interpretation
e. Selecting alternative interpretations to pursue

f. Developing assertions or generalisations about the case

The first screening of the data (ProLearn) takes as its subject one single case of a professional software developer, while the next case analyses selected examples of social interaction within an open-source software developer project. What is unique and specific for the individual cases will be treated in the case descriptions. The research purpose of the initial case is foremost explorative; namely to screen for applied knowledge resources and explore potential relations towards knowledge objects in the field of software development. It can also be pertinent to search for the presence of patterns or themes in the phenomena. Themes are defined by Gall et al. (2003) as "salient, characteristic features of a case" (p. 439). For example, if a software developer connects with a particular object-oriented expert community online, and consults it to solve a problem on one single occasion, it can be observed as an isolated incident. However, if the developer participates in the community on a regular basis, e.g., by contributing programme code and sharing knowledge in the expert forum, it might indicate a significant theme in her orientation towards objects of knowledge. In Stake's (2005) terms, this case can be seen as instrumental, whereas a single case is examined primarily to gain insight into an issue. Furthermore, the study also aims to create thick descriptions of the phenomena informed by contemporary theories. The term thick descriptions is associated with the interpretive works of Geertz (1973). It connotes a level of description that is sensitive to the details of the studied activity or event, but also accounts for multiple meanings by interpreting the relationship between symbol and context. This implies rather dense, small-scale descriptions of the software developers' relations with knowledge objects. When constructing the thick description, I apply theoretical constructs in order to structure my understanding of the descriptive data. I will also relate the conclusions drawn from data to related research findings reported in the literature.

The research purposes of the second case are partly descriptive, but also involve an explanatory dimension investigating the role of intersubjectivity in software developers' mediated communication. However, the intention is not to claim any causal relationships. The explanations will rather describe a set of variations in patterns among two contrasting sub-groups within the mod_perl project. In this manner, I examine if one sub-group's processes of intersubjectivity systematically relate to another's. If a pattern exists, it can be described a relational rather than causal (see e.g.Gall, et al., 2003).

With regard to the two case studies and associated interpretive analysis, the notion of validity may refer to the degree to which the analysis is properly conceived in order to address the research aims (Calhoun, 2002). Yet, there are numerous definitions of validity available in the literature. It is not clear-cut how these apply in case study research and inquiries which 'mix' numeric and qualitative sources. Can we say that the research procedures and interpretation of results from such diverse sources are justified on reasoned grounds? What about reliability; would another researcher obtain similar results if the research procedures were to be repeated? A common criticism of case study research is the limited ability to generalise or transfer results to other cases. However, several influential authors from the case study tradition and multimethod/mixed method approaches contest that criticism by suggesting that the inferences made from these studies need to be treated differently from statistical generalisations. For example, Yin (1997, 2008) maintains that analytic generalisations can be drawn from case study research when previously developed theory is used as a pattern for comparing the empirical results. Tashakorri and Teddlie (2003) suggest using the umbrella term *inference quality* as an alternative to validity or trustworthiness in order to express the quality of conclusions drawing on multiple sources of evidence (combining qualitative and quantitative data). An inference from this point of view is considered to be the researcher's interpretation of the study's results. It refers to the degree of accuracy of both our inductively and deductively derived conclusion from a study, "a process that encompasses both internal validity and credibility" (ibid.:38). Furthermore, they define design quality, interpretative rigor and transferability as interrelated aspects of inference quality. The study's inferences are discussed further when evaluating the thesis' limitations in chapter 7.1.

4.2 Studying Computer-Mediated Communication (CMC) with Computer-Assisted Qualitative Data Analysis Software (CAQDAS): Some Issues and Implications

A few practical advantages are associated with incorporating CMC into the design of qualitative studies. For example, networked computers allow researchers "to cross the time and space barriers which might limit FTF research" (Mann & Stewart, 2000:17). However, because CMC can potentially speed up the process of readying a comprehensive script for analysis, according to Mann & Stewart (ibid.), we need "to consider the quality of the data provided in this way" (p. 23). In the ProLearn project, the learning logs were registered in an

online form (on a webpage), and immediately stored in a protected database. Hence, there was little need for transcription. Raw data could easily be converted into an appropriate text file and imported to NVivo[4] and Atlas.ti[5] for data management and analysis. Furthermore, in combination with the other data sources, online textual data provided a new angle to shed light on central themes in the second case study included in this thesis (chapter 6). According to Silverman (2001), a conventional observational field study seeks to collect firsthand information about social processes in a naturally occurring context. Yet, online observation is associated with various costs and benefits, and also "challenges the basis of terms such as 'observation' and 'natural contexts' as used in traditional research" (Mann & Stewart, 2000:84). Text-based CMC does not inevitably aptly represent the real world. When conducting fieldwork in an online environment, observers are unlikely to gather data from co-located, face-to-face interactions. The field site therefore provides grounds for somewhat limited observations, because researchers cannot observe 'real people' (Wittel, 2000).

4.2.1 Structuring the analysis

Both the ProLearn project and my distinctive online research approach called for substantial data organisation. In both cases, this involved tracking participant activities over time, structuring background/context information, tying together different sources around cases, making notes as the study progressed and retrieving/combining significant passages of text. Most of these organising tasks can also be done by hand, for instance, by applying paper, coloured text markers, scissors and index cards; although CAQDAS tools such as Atlas.ti or NVivo can facilitate data management and analysis in multiple ways. In fact, these packages are found to "facilitate chores of data management which are tedious and subject to errors when done manually," making the "analytical process more transparent and reviewable" and offering "support for new approaches to analysis or approaches that would be very cumbersome if done manually" (Fielding, 1999:96). Such potentials are also regarded positively in relation to handling the electronic databases gathered from online research (Mann & Stewart, 2000). Digitalised textual data from online interactions can be swiftly transferred into the representations provided in programmes such as NVivo.

[4] *NVivo* software to assist qualitative data analysis: QSR International Pty Ltd. Version 8, 2008.

[5] *Atlas.ti* software to assist qualitative data analysis, management, and model building: ATLAS.ti Scientific Software Development GmbH. Version 5, 2005

I am aware that computers do not actually perform the intellectual work essential for interpretations and qualitative research; but, when used sensibly, they can support the research processes to a considerable extent. However, taking advantage of such tools requires some basic computer skills, methodological training and the will to use both. As the ProLearn project progressed, it became evident that the researchers involved had various preferences with regard to using software to facilitate their analytical work. The code-and-retrieve features and the ability to get organised printouts with references to original context (including paragraph numbers) were viewed positively by most. This turned out to be practical during seminars for joint analysis. Printing compendiums with complete datasets provided rapid access and common references for discussions. Comprehensive memos were written and attached to their respective data sources. However, few of the researchers involved used the potential within the software to make conceptual links and support theory building.

4.2.2 Notes on the coding process

Systematically coding text segments by hand is commonly regarded as creating the basic building blocks for qualitative data analysis and theory development (Strauss & Corbin, 1998). Codes that represent categories of data more or less embody our assumptions throughout the analysis, as abstract representations of an emergent object or phenomenon. Codes can operate on a continuum from simple descriptions to more interpretative or analytical concepts (Richards, 2005). Because several researchers worked on the same set of data from different angles in the ProLearn project, we adopted a team-based coding approach (Bourdon, 2002). A consensus was reached during project meetings to develop a thematic codebook for an initial, broad sorting of data into major categories. The thematic codes were based on the project's main dimensions for exploration. The purposes were to isolate and organise text sequences relevant to central project themes, and to pave the way for further analysis. In the research group, there was general agreement that theme codes should not be too reductive nor exclude valuable data. Individual researchers were urged to go through data again (after the initial theme coding) with their own specific labels in order to achieve more detailed and purposeful coding than would otherwise have been possible with only a single pass.

A research assistant familiar with the field of inquiry was hired to initially sort the data. She was provided with a sheet explaining the codes, and with detailed guidelines on how to apply them. Following is a summary of theme code guidelines:

- Theme codes are intended as an initial broad sorting of the data (Bourdon, 2002).
- They can overlap and are therefore not mutually exclusive. One sequence of text can be marked with multiple theme codes.
- Paragraphs that are not covered by the codebook, and that seem irrelevant for the project's theme, will not be coded (redundant material to be deleted).
- Paragraphs that do not fit the theme codes but are nevertheless considered relevant can be marked using the 'other' category.
- An item of general advice is to do two rounds of coding with each source: initially to identify and code what apparently fits the prescribed codes; then subsequently the complete text is reread to determine whether coded sequences could be given additional codes, and if uncoded material could be coded or left as redundant (Steadman, 2005).

When I started to work with the data collected jointly in the ProLearn project, I refined the initial categories to illuminate my research question. After this, the theme coding done by the assistant had limited value. Even if the tools provided in the software were flexible and could be adjusted as the analysis progressed, the theme codes were appraised as unreliable. After a closer reading of several sources, I found crucial passages of data left uncoded and our views also diverged in relation to how the labels should be applied. On the other hand, data sources were neatly formatted with subheadings. This enabled me to do a rough auto-coding in NVivo, organising the material into questions and answers. I was thus able to read and code a blank dataset using labels more consistent with my research questions. Moreover, I developed concepts that went beyond the descriptive labels and started identifying patterns in the data.

In sum, digitalised textual data from online interactions were effortlessly imported into NVivo. Working with a large database, analytic tools aided data organisation and structured the analysis. However, the team-based coding approach adopted was problematic. The NVivo auto-coding feature (from subheadings) enabled me to rapidly organise the sources and

subsequently perform a more detailed and purposeful coding consistent with my aims and research questions.

5 Artefact mediated relations towards objects of knowledge in software development

[Empirical study I]

5.1 Issues and borders for exploration

In the previous chapters, the combined research purposes of this thesis have evolved into more specific issues for exploration. The raison d'être for organising the study in themes and issues has been twofold; to learn from the literature and to provide grounds for undertaking inquiries into the knowledge practices of software development. In this chapter, rough themes are further developed into key issues, providing a more specific conceptual structure and research questions for analysis. Following Stake (2005), the selection of key issues is crucial: "Which issue questions bring out our concerns? Which would be the dominant theme?" (p. 448). The key concerns of this first explorative case can be summarised as follows:

A. Topical issue: Artefact-mediated relations towards knowledge objects.
B. Research question: What are the key artefacts applied as knowledge resources in software development? How can they mediate the relational properties associated with objects of knowledge?
C. Assertion: Embedded in key artefacts applied as knowledge resources are codified trajectories and social attributes inviting participation in distributed knowledge practices.

According to Stake (1995), important parameters in the case studies' design are the boundaries enclosing the cases. The unit of analysis for the current case is defined as an individual professional software engineer and the system of artefacts applied when searching for knowledge in a workplace context. As indicated in the chapter on overall methodology, some outside features may also be significant for understanding the context of the present case. The artefacts applied as knowledge resources are likely to span the borders of time and space, connecting to the distributed communities of other developers. For example, different modes of interaction can be classified in a 2 x 2 matrix as suggested by Johansen (1988). The time-place matrix below identifies four classes of corporative work situations along the dimensions of time and place. The content of the four classes in the matrix is adapted to illustrate two contrasting knowledge seeking scenarios.

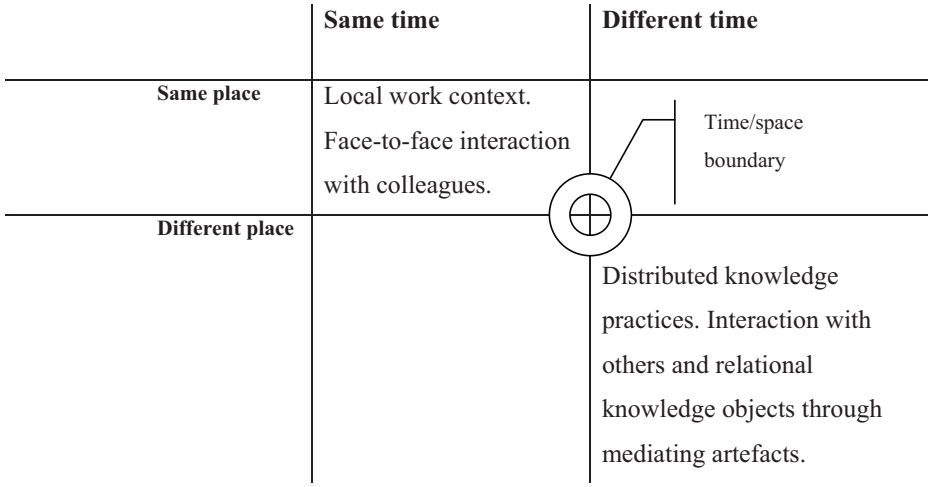


Table 5: Time and place matrix adapted from Johansen (1988)

In order to investigate the potential of key artefacts to mediate the relational properties associated with distributed objects of knowledge (lower right square in the matrix), the borders for exploration may stretch beyond the spatial-temporal relations of the local physical context (the upper left in the matrix). This suggests that it is reasonable to include features from the external environment in the boundaries enclosing the case. Such external features can, for example, be an online help library that is incorporated into the Integrated Developer Environment (IDE) applied locally. For example, when the help function embedded in an IDE is loaded on the developer's local computer, it might connect automatically to an external support database. It is reasonable that the computational artefacts stretching beyond a local context ought to be included within the case borders if they are reported as resources when searching for knowledge in a workplace context.

The next issue is how to identify the key artefacts applied as knowledge resources. Here, a combination of reports from the field and background theory can be constructive as selection criteria. It is pertinent to draw on the respondents' reports on what is considered important resources in work-related problem solving. However, selection criteria in this case should also comprise theoretically derived features of significance for the aims and research questions. To sharpen the inquiry, central artefacts are to include crucial issues of concern for this thesis; namely the potential to mediate objects of knowledge and connect the subject to distributed, networked knowledge practices. For example, this may be an integrated artefact with

embedded links to relevant discussion forums or databases that are shared among distributed communities of software developers. Moreover, this implies computational artefacts with embedded interactional features that allow information to flow between the subject and object-centred communities. It can also be useful to assess their closeness to what is commonly regarded as the central phases in software design; namely producing code (constructive phase) and argumentative problem solving/documentation (see e.g. Fischer, Lemke, McCall, & Morch, 1991). The latter are the artefacts involved when 'getting stuck' in the construction phase and one needs to consult peers in e.g., developer networks.

5.2 Joint data collection in the ProLearn project

The empirical data for this group-specific inquiry are collected in conjunction with researchers in the ProLearn project. The principle data sources are interviews, learning logs and documents (see Table 6). ProLearn followed a longitudinal design, focusing on professional education and the young professionals' transition to work life. Longitudinal is a broad term, but in this case it means that data are collected at a sequence of time points for each of the targeted groups (nurses, teachers, accountants and engineers). Longitudinal studies of learning at work are not common in the field (Eraut, et al., 2002); nevertheless, this design opens up for the study of learning and social change over time. In the ProLearn project, data collection was divided into two main phases: initially in the context of professional education and subsequently, postgraduate employment. The phases are indicated with brackets in the overall design represented below.

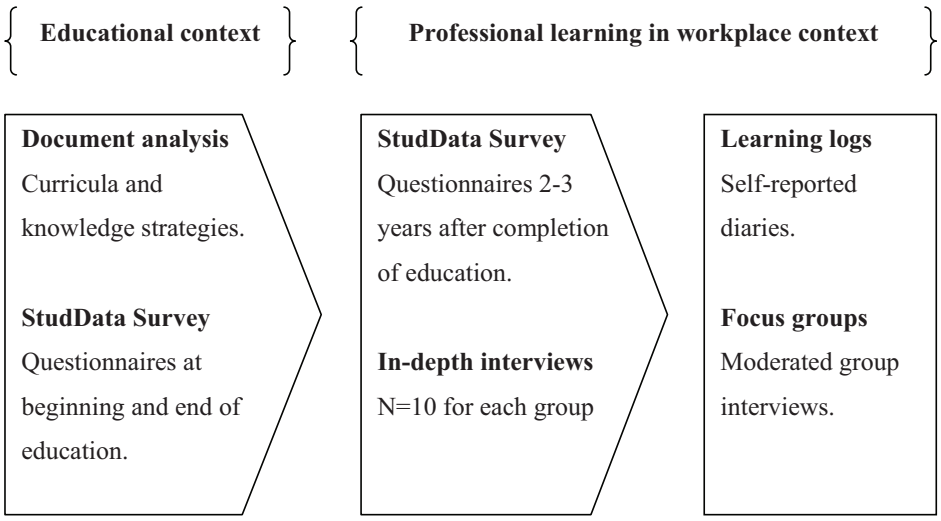


Figure 4: Meta design ProLearn project (adapted from Karen Jensen, et al., 2008:5)

The subjects analysed were the same from each period to the next, making comparison of data among periods possible. Respondents for the interviews and learning logs in the ProLearn project were chosen from the national StudData database. The rationale for this strategy was to acquire a selection of respondents based on the year in which they graduated from Oslo University College (HiO). This provided the involved researchers with the opportunity to study groups of relevant respondents over time, with assorted analytic foci and methods. Of the four professional groups targeted in the overall ProLearn project, the group of computer engineers is focused upon in this thesis. The informants in this group consist of 3 females and 7 males (n=10). They are predominantly under 30 years old (two of them are 32). As a group they represent a rather varied workplace environment; some of them are from very small companies and others from large multinational firms.

The empirical data underpinning the forthcoming analysis is mainly collected jointly in the ProLearn project during its second phase. The context for my involvement in the joint data collection is confined to postgraduate employment (right section of Figure 4). I conducted personal interviews among all professional groups, participated as assistant in various focus group interviews and administered the collection of learning logs. However, for the specific purpose of this thesis, additional data were needed. To differentiate between the data collected

in collaboration with the ProLearn researchers and the data gathered intentionally for this thesis, I will refer to the former as *common data* and the latter as *core data*.

Data source	**Data Type**	**Theme**	**Status**	**Time collected**
10 in-depth semi-structured interviews	Transcribed text	Transitions to work life. Exploration of ProLearn project's key themes.	Common (case I)	2005
Approximately 20 days of learning logs	Self-reported text	Formal and informal learning situations. Ties and alignment to knowledge.	Common (case I)	2005
Focus group interview	Transcribed text	Knowledge cultures. Visions for the future.	Common (case I)	Spring 2006
Documents, artefacts and literature.	Text, graphics and computer interfaces.	Knowledge practices across contexts and artefact mediation.	Core (case I and II)	Continually
Field study of online community for developers.	Message archive from developer forums. Documents.	Processes of intersubjectivity around joint objects of knowledge.	Core (case II)	2008-2009

Table 6: List of data sources

The data sources labelled as ‘common data’ in the above table are shared among the ProLearn researchers and gathered as a collective effort. Note that different subsets of data were also collected among the targeted professional groups in the ProLearn project, but these are not included in the above table. This is due to their limited explanatory value for the research problems probed in this thesis. Excluded sources are the data gathered from the StudData survey in which a population of students from professional educational programmes were examined at different points in time, and the field-specific document analysis focusing on professional knowledge strategies. In the following paragraphs, the different data sources, referred to as *common data* in this thesis, are described shortly. The descriptions draw on project notes and the final ProLearn project report (Karen Jensen, et al., 2008). The core data specific for case II is described in further detail in chapter 6.4.2.

5.2.1 Semi-structured interviews

Personal (face-to-face) interviews were conducted with 10 respondents from each of the four targeted groups in the ProLearn project. Samples from each group were randomly selected from students graduating from the Oslo University College in the years 2001 and 2002, using the StudData database. The questions put to interviewees were relatively informal and aimed primarily to collect information about their own learning and knowledge development at work. A semi-structured interview guide was designed to be applied by the researchers in the interview setting (see appendix 9.1 for personal interview guide). Key topics or issues for the interviews were decided collectively by the researchers involved in the project, reflecting consensus with regard to the main areas for exploration. The loosely structured guide consisted of a set of questions around our key foci, allowing the respondents to talk rather freely about their experiences and the researcher to ask follow-up questions arising naturally throughout the interview. The first part of the interviews concentrated on factual/contextual information; asking questions about educational background, current work situation and activities. The remaining part focused on learning potentials at work, problem solving and learning strategies, especially challenging and interesting tasks, applied knowledge sources and affective relations towards knowledge. The objective was thus to present open-ended questions in order to understand the respondent’s point of view in relation to these issues.

5.2.2 Learning logs

Learning logs were applied as a form of self-report diary for making notes about problems and knowledge sources applied throughout the day. In the ProLearn project, the learning logs were incorporated into the design to complement the semi-structured personal interviews. A logging system was developed to gather data closer to the workplace context, and to provide material for discussions in a later focus group. A central issue was to explore how respondents connect to knowledge in their daily work. What 'triggers' a knowledge seek? What is the nature of their sources? Every respondent interviewed in the current phase of the project was invited to participate in writing learning logs. The selection criteria were hence the same as for the semi-structured interviews. After the personal interview, respondents were asked to participate by writing learning logs in a predefined form at the end of four freely chosen workdays within a two week period (see learning log form in appendix 9.2).

For practical reasons, we prepared a paper version and an online electronic version. Some of the participants did not have easy access to computers and the Internet, and preferred to use pen and paper. The forms included a short introduction explaining the procedure and purpose of the logs. Questions placed on top of the form fields were relatively open and invited participants' own reflections. In the electronic version, we used a *web form system*[6] developed at University of Oslo's Center for Information Technology (USIT). This is a comprehensive tool for administering online questionnaires. An e-mail containing instructions and a link to the form was distributed to respondents who had requested to complete their learning logs electronically. Earlier experience with open-structured learning logs, where the researchers are not present at the site, indicates that a strict follow-up routine is necessary in order to keep the loggers committed to the task (see e.g. Fuller & Unwin, 2003). As a consequence, we sent out frequent reminders during the log period (four times in two weeks). Respondents writing electronic logs were sent e-mail reminders, and the ones using pen and paper were reminded via SMS message.

In line with the longitudinal design of the ProLearn project, and in order to control for seasonal variations, we prepared two rounds of logging activity. The first round proceeded in spring 2005, and the second in autumn 2005. To keep the respondents committed and interested in the project, we offered them a gift certificate valued at 500 Norwegian kroner as

[6] https://nettskjema.uio.no/

a reward for completing both rounds. Overall, 35 respondents from the targeted groups (engineers, teachers, nurses and accountants) agreed to participate in writing learning logs. This resulted in a total of 141 log days. The table below shows computer engineers' participation in the two rounds.

	% response	**Number of log days**
First round	6 of 10= 60 %.	15
Second round	5 of 10= 50 %	20
Total:	11 of 20 = 55 %	35

Table 7: percentage of learning logs completed by computer engineers

5.2.3 Focus group interview

Focus groups were incorporated as a research strategy to complement the other sources in the second phase of the ProLearn project. This element in the design is partly justified in the belief that interaction between focus group members could provide additional information that was currently lacking in the personal interviews and learning logs. Individual interviews and learning logs were thus followed up by focus group interviews during spring 2006. Interesting profiles among the respondents previously interviewed in person were invited to participate in small-group interviews. As far as practical concerns allowed, we attempted to select participants who represented variation in gender and work contexts. The assembled groups each consisted of three individuals from the same profession. For example, the computer engineering focus group had three male participants with equivalent educational backgrounds at the time of the interview, but embodying diverse work experience. In addition, two researchers (one moderator and one assistant) were present during the interview. The group interview sessions lasted about two hours. Discussions were recorded and later transcribed to make them more accessible for analysis. As with the personal interviews, a thematic guide was collectively constructed in advance. This included a set of a priori foci formulated as open-ended questions intended to initiate discussions in the groups (see appendix 9.3 for

focus group guide). The aim here was to corroborate and challenge the other data sources by presenting our preliminary interpretations to the respondents as well as collecting supplementary information about the groups' knowledge cultures. Under the moderator's guidance, discussions were centred on their personal experiences within the knowledge field, focusing especially on interaction with different knowledge sources and exploring potential epistemic trajectories.

5.3 Selecting case for exploration

In accordance with the first aim stated in the introduction, the case study is intended as an explorative device for researching the potential of key artefacts applied as knowledge resources to mediate relational properties associated with knowledge objects. Exemplified by a single-case study, the relation between local problem solving and knowledge development in distributed expert networks is to be investigated. It thus targets and analyses artefact-mediated connections to knowledge cultures across time and space. Furthermore, the notion of epistemic trajectories is explored as an analytic tool for describing the artefact-mediated patterns of movement from local problem solving to distributed knowledge practices. The rationale for selecting a single case is presented in further detail below.

The single case is selected from the sample of 10 computer engineers participating in the ProLearn project. Due to the extensive data sources produced in the project, and the fixed timeframe of my PhD work, I chose to do a 'purposive selection' of a case from which I could learn the most (Gall, et al., 2003). Furthermore, definite purposes guiding the selection of the single case is that it should represent rich data and variation in its orientation towards knowledge sources. The case should reflect the phenomenon of interest by rendering key artefacts applied as knowledge resources within the field of software development. This implies intense studies of one individual developer in his/her naturalistic settings. It follows that the overall aim is not to produce statistical valid data for generalising about key artefacts and object relations. The purpose is rather to select one case that offers intrinsic potential, and employ a mixture of methods to study it. However, this does not necessarily mean that the knowledge obtained will be restricted to the single case (Calhoun, 2002). The results might also be employed in later comparative analysis to test theories and perhaps investigate the relationship between the single case and a population of developers to provide modest analytic generalisations. Yin's (1994) typology of designs (see chapter 4.1) distinguishes

between single- and multiple case studies and two levels of interferences. A case study can accordingly target one single case or multiple cases. In the former, the researcher may embark upon an examination and testing of theoretical propositions if the targeted propositions are defined prior to the study. To paraphrase Yin (ibid.), the single case would then provide the opportunity to substantiate or challenge the set of propositions. Furthermore, he argues that single cases can enable us to generalise to other cases that somehow represent similar theoretical conditions. The following quote illustrates the difference between conventional statistical generalisations and analytical ones:

> In fact, the classic single-case studies are classic in part because of their broad implications or generalizability - even though only single cases were the subjects of study. In other words, generalizing from case studies is not a matter of *statistical generalization* (generalizing from sample to a universe) but a matter of *analytic generalization* (using single or multiple cases to illustrate, represent, or generalize to a theory) (ibid.:239).

In this view, case studies only lead to analytic generalisations. Although, George & Bennett (2005) maintain that there is no such thing as single-case studies. On the contrary, "different case study research designs use varying combinations of within-case analysis, cross-case comparisons, induction, and deduction for different theory-building purposes" (p. 49). For transparency and to prevent misunderstandings, I find it reasonable to continue by defining *the case* as a unit of analysis. The decision stated above is to limit the study to one individual developer and the set of artefacts applied as knowledge sources as the current case. Consequently, the inquiry does not involve replication logic, as proposed by Yin (1994, 1997), in order to draw inference from multiple cases. However, the research question guiding the inquiry implies a unit of analysis that can be sampled from a group; namely *software engineers* within the ProLearn project. Consequently, the next decision is to select a purposeful 'sample' within this group. Among the 10 computer engineers participating in the ProLearn project one individual developer and the set of artefacts applied as knowledge resources are selected as the case. This is partly justified by time/resource restrictions. It would be impractical to embark upon an in-depth study of all the engineers participating in the ProLearn project within the timeframe of this thesis. The study of multiple cases may also decrease the analytic attention that can be granted to each one, thereby preventing an in-depth study. I find it more feasible to consider which of them has the experience and insight that can

provide data sources of special value for the research purposes of this thesis and study only one case in depth.

In sum, the design of this case study is in part *explorative*. Empirical data underpinning the forthcoming analysis were collected jointly in the ProLearn project and the study was undertaken prior to definition of the final research questions. Initial guiding aims were presented in the introductory chapter of this thesis. However, as the study progressed and my understanding of software development practices grew, a more precise research question and issues were presented when introducing this case. To explore these issues and answer the research question, I find it constructive in the first level of data presentation to describe the work context of the case by applying selected themes from the interview guides and learning logs. These themes reflect key dimensions funnelling the data collection and have also been functional codes for organising data. Thereafter, I move on to 'thicker' descriptions of artefact-mediated relations towards objects of knowledge. The inquiry thus progresses in two different phases; first by screening, describing and categorising the knowledge sources applied in the software developers' problem solving. Subsequently, key artefacts are selected for further analysis.

5.4 Screening case D19

The population of computer engineers studied in the ProLearn project represents a rather varied group with regard to work contexts and central tasks. They come from the public and private sector, small and large companies and are involved in both system engineering and software development. Common features in their individual biographies are a genuine interest in computer technology. Most respondents attribute their interest in the field to new technologies. With respect to relations to knowledge, they describe a rather pragmatic orientation and the joys coupled with trying out new solutions and getting them to work. At the group level, they report access to a large amount of knowledge resources, among them workplace colleagues, specialised books, periodicals and various online resources. However, D19 stands out as especially reflexive when describing his relations to knowledge. During the personal interviews, he appears eloquent and meditative, producing rich and relevant empirical data. He also represents a somewhat different identity from the other developers in the ProLearn population, as he indicates a concern for human relations and psychology

together with an experimental approach to knowledge and technology. On this basis, he was also considered a valuable candidate to participate in the focus groups.

In order to provide a basis for understanding the tasks and problems of a professional software developer, I first embark on describing the work context of D19. The data presentation is structured by applying relevant themes from interview guides and learning logs as follows: Individual biography (education, intellectual drive, transition to work life); current work place features (type of enterprise, collaboration pattern); central tasks at work; experienced problems (especially challenging problems/situations); triggers (interesting tasks and problems initiating knowledge seeks) and applied knowledge sources. Next, the set of artefacts applied when searching for knowledge is categorised and screened for features according to the guiding aims and research questions.

5.4.1 Brief individual biography

Prior to his three years of college education to become a computer engineer, D19 spent one year at the university studying elementary level psychology. In our first interview, he conveys that he really wanted to be a psychologist. But, as a freshman directly out of secondary school, it was in his experience difficult to gain admission into the full programme in order to become a professional psychologist. He had two main preferences when choosing further education; the first was to work with technology, and the other to work with people. After some consideration and thought during a year of compulsory military service, he decided to become a computer engineer. This was, in his view, “the best education to work with technology” (Personal interview, March 2005:7). Yet, his experience with programming was limited at the time. After graduation, he had a short period of unemployment. The extra time in between was spent reading and acquiring certifications in various technologies in order to strengthen his ‘market value’ and increase the possibilities of working as a consultant. He explains that “the field is very keen on certifications nowadays” (ibid.:88). Nevertheless, he perceived the transition from an educational context to work life as being especially abrupt. There was a change in pace, more implacable demands and tighter deadlines than those to which had become accustomed. As a new employee, he spent a few months before starting to feel that he was gaining mastery of the new situation.

5.4.2 Current work context

D19 is currently employed as a consulting developer in a multinational corporation which specialises in developing software and hosting solutions for a wide range of fields. Its main markets are large and medium-sized organisations. The respondent expresses that he finds his work tasks in the company to be varied and mostly gratifying: "That's what is exciting about being a consultant; there are lots of different tasks depending on the type of project you are working on" (Personal interview, March 2005:88). During the last few months prior to the interview, D19 has participated in several projects. Projects, he explains "can be everything from developing extensive solutions for large organisations to a week-long assignment for a small firm" (ibid.). Also, he reports that this work as a consulting software developer involves quite a lot of writing of project proposals. When the projects are of a certain scale, calls for tenders are laid open by principals. They then have to compete to obtain the assignments. However, this part of the job, he reports, is not always enjoyable. If he writes a proposal for something in which he does not really wish to be engaged, it can be boring. In his view "it can be slightly tedious to be put on an assignment that involves obsolete technology that you just have to trench through" (ibid.:143).

With regard to collaborating with peers, D19 sees group efforts with other developers as essential in this kind of work. He portrays the climate in the developers' room as follows:

> There are many loud discussions and sometimes a high temperature in the developers' room. It is well known that developers' rooms have a somewhat peculiar milieu. You just have to have a whiteboard in the room, because there will always be drawings and discussions around drawings and stuff like that. But for some odd reason, you always end up with agreement, more or less (ibid.:278).

The issues at stake, he elucidates, can be about superior choice of technology in a project, for instance, Microsoft.net versus J2EE as a platform, down to a more specific level concerning what classes to incorporate and how they are to communicate. Another collaboration practice existing among local developers is referred to as 'code review'. D19 indicates that the developers at his workplace 'make a thing' out of code review sessions. "We go through each other's code, and then it's very enjoyable to bring up the pigtail … then you can bitch a little bit, as long as it has a humouristic tone ... this is how we keep up and perform best practice"

(focus group interview, April 2006:1099-1119). The review practice is accordingly not organised in rigid procedures, but it is up to each developer to book a code review when needed.

The development projects in which he participates are commonly organised as teamwork. Four persons were members of the previous project group in which he was involved. A project, he declares, "typically involves planning as an essential part, departing from some rather loose specifications from principals" (Personal interview, March 2005:105-109). The solutions are then further planned all the way down to the component level; which classes to be included and how they will communicate. This he sees as an incredibly important process, "because the first sketch will always be wrong, and that is the one we are likely to follow if we start programming right away" (ibid.). Some programmers, he claims, have a tendency to be sloppy in this phase and jump right to the programming. This approach is, in his opinion, likely to fail. D19 thus indicates the value of thorough planning in advance. During the lifespan of a project, documentation is also highlighted as being of vital importance. The idea is to get an overview or a map so that it becomes apparent how the classes interact in the programme and what tasks they perform. If documentation is not provided, other developers will later face difficulties; "they will have to read maybe 5000 lines of detail to get the general idea instead of just having an A3-sheet in front of them where they can see the overall picture" (ibid.:305). However, despite the emphasis on documentation, D19 points out that it is not really necessary to comment on the source code. He suggests that "if one knows how to programme, the source code will mostly articulate itself" (ibid.).

Moreover, D19 perceives his work with technology mainly as creative problem solving. However, he sometimes has to take part in the whole process (including economy) because managers are fully booked. The total responsibility can be delegated to single consultants or a group of two to three people. It is then up to managers to read 'Quality Assurance' and verify that the project stands and the pricing is about right. When it comes to technology, he identifies J2EE as his major area. The vast complexity of this technology is expressed in the following quote:

> It's incredibly complex ... a simple acronym, but an enormous technology ... it's apparent that you have to work many years with J2EE before you get an overview of all the different parts of the platform. You will always discover new areas within the

> J2EE platform and then quite a bit of reading is required to get to know the interfaces that are there (Personal interview, March 2005:62).

This platform he assesses as future-oriented, 'living technology'. D19 further explains that it is Sun that owns the J2EE platform, but that they have made it open so that everyone can use it freely. The openness makes it a superior choice in his opinion; "the openness is what makes it preferred compared to other closed platforms" (ibid.:128*).* On the other hand, the open JDEE technology is also coupled with restrictions as Sun Microsystems partly controls its future development.

As indicated above, D19 assesses the tasks related to software programming as creative problem solving. He maintains that most challenging tasks contain elements of innovation. When we asked if there is something in particular with his work that gives him a 'rush', he responds that it is the very process of finding solutions to difficult tasks:

> "When you've been sitting for two days and really scratched you're head, facing ideas that afterwards seem hopeless ... and when it is solved, it's a real kick. This is what I can live on for the rest of the day ... it really motivates me (ibid.:135).

Nevertheless, his daily tasks indicate reciprocal ups and downs. He maintains that this allegedly provides desirable variation to the work. D19 is also rather explicit about his frustrations over problematic integration with old technologies. System integration is about getting different systems to communicate. He sees this as a major challenge. A recurrent problem mentioned is the need to get a sufficient overview of what he refers to as the technology jungle; "you must always expand the horizon by reading about numerous new technologies" (ibid.:237). To know what is going on in the field, he monitors a few technological areas.

> It's more like, you know what's going on. But you don't go deeply into it at the time. You have a little eye on it, but do not go seriously into it; maybe try it a little bit, maybe 10 or 15 minutes, and then put it away. And next time I bump into a similar problem, I will look closer into it. And if it's good, it works and has been applied by others before, received good reviews and so on, I adopt it into my code (focus group interview, April 2006 : 311).

The quote above signifies an orientation towards issues that are not directly related to the tasks on which he is currently working. When finished with the current project and new problems arise, such chunks of knowledge can be useful. The sources for this monitoring practice are found primarily on the Internet. In addition, he subscribes to a single periodical from Sun, which is the provider of Java technology. This brings us to the next theme; exploring the set of artefacts applied as knowledge resources in a local work context.

5.4.3 Set of artefacts applied as knowledge resources

In order to indentify and map the artefacts applied as knowledge resources in a workplace context, this subsection starts with thin descriptions of the various artefacts applied when searching for knowledge. Thereafter, they are mapped and categorised to gain an overview and identify 'key artefacts' with the potential to mediate relational properties associated with knowledge objects.

In the personal interview, the respondent was asked what drives him to search for new knowledge and what he prefers to read in relation to his work. He was also asked to be specific about the content, medium and type of knowledge that he commonly applied when performing his work. In the learning logs, predefined fields included prompts to make notes about technical questions that had appeared during the day, and the knowledge sources (if any) that were applied to find out more about this issue (see appendix 9.2).

D19 reports that he often (almost on a daily basis) uses how-to's or walk-throughs to aid problem solving (focus group interview, April 2006). This is described as hands-on literature, directly applied in local problem solving. It is the "stuff you can't be bothered learning, because you don't need it in the big picture. You just have to pass the obstacle to solve a problem" (ibid.:1530). These sources are mostly associated with reference books. Besides reference books, he prefers articles and technical journals. D19 underscores that educational books become outdated extremely quickly; "When I sometimes read educational books, they are wrong! This is not how we do things nowadays; it's not best practices or anything (ibid.:996)". In D19's experience, academic textbooks in programming lag behind and are sometimes not sufficiently rooted in real life. A programming example is mentioned:

> In order to demonstrate the syntax in a given situation, the educational books apply it in a way that you would never do it in real life. For instance, when you teach JSP programming, you write the code in the JSPs — you say the functional code in JSP, and this you should never never do in practice. But clearly, if you sit there with an educational book, you can become outdated believing that it is okay. So the educational books, at least the one we had at school is wrong. From this outlook, the Internet is a much better medium for us (focus group interview, April 2006:1574).

The above quote indicates that D19 assesses his educational books as rather obsolete and of limited value as sources. The set books from college are still stored in the basement. He claims to know for the most part what is in them. If there is something in particular that he needs to look up in the educational book, he can simply "walk down to the basement and get it" (ibid.: 218). However, this has not yet happened. In the learning logs, specialised books are reported in combination with other knowledge sources in daily problem solving. These books are perhaps more up-to-date, belonging to the 'hands on' category referred to in the personal interviews. For example, the following extracts from the learning logs mention books and internet sources applied as knowledge resources.

> **Experienced problems:** How is an incomplete backup of Oracle 9 database done? How is RMAN set up to automate backup routines?
>
> **Applied knowledge sources:** Internet resources and specialised books
> (Log reference 41972).

The specialised books referred to may consist of printed pages involving detailed and specific knowledge relevant to the experienced problem. On the other hand, they may also be scanned parts of books stored in an electronic format such as Adobe PDF. It is reasonable to assume that a reference book on database technology is applied here in combination with Internet resources. A book on a specialised subject within computer technology and programming is also likely to include references and hyperlinks to Internet-based resources. In this way, the mediating artefact can guide its user towards more dynamic resources online. However, the data do not provide sufficient support to claim the existence of such patterns. The data rather indicate that combinations of non-interactional and interactional artefacts are applied as knowledge resources in immediate problem solving.

In the personal interview and learning logs, Google is repeatedly mentioned as a valued starting point for searching programming-related discussion forums on the Internet. When experiencing a programming problem, there is, in his words, “almost always” somebody else out there who has dealt with the very same or a similar problem that can be applicable as an example. As he puts it:

> Google is just genius for developers. Google has a very fine search functionality that allows you to search discussion forums. Often, if I get a new error message or something like that, it is sufficient to paste the error message in Google and search in groups and then I get someone who has already discussed this among them in these groups, and I’m likely to get the answer (Personal interview, March 2005:392).

New errors can thus be pasted into Google and one can search different discussion groups to see if anyone has experienced and perhaps solved the problem. By using this approach, D19 reports that, for the most part, he gets the answers. It also indicates that a broad range of net-based knowledge resources are applied by searching the global version of Google. As he describes it, “I just search Google and a number of links appear. And then it’s mainly specialised articles that are published at universities or in other forums, seminars and so on. But it is mainly through Google that I stumble upon these” (ibid.:179). In addition, there are a few favourite websites that he visits on a regular basis that also have search functionality. The central role of search engines in the quest for knowledge is also corroborated by the reports written in the learning logs. The extract below gives an idea of the concrete problems experienced in the workplace context and a typical search pattern.

> **Experienced problems**
> Programming specific problem: How to read in a text String to a XML tree (DOM)? Database specific problem: How is the name on a column in an already created table changed? Subversion control system: how is an error report situation solved when something is uploaded?
>
> **Applied knowledge sources**
> Google has led me to different online discussion forums that provided me answers. In addition, colleagues have been helping me out. (Logg reference 51893).

XML stand for “extensible markup language” and is a simplified language especially designed for Web programming (Daintith & Wright, 2008). A subversion control system is an application for managing different file versions and maintaining the development history of software projects (Karl Franz, 1999). As indicated in the above log extract, the Google search engine plays a role in connecting the respondent to different user forums on the Internet as a pathway to his immediate, task-related problem solving.

In the personal interview, D19 explains that forums such as Sun offer incentives in the form of Duke Dollars which are awarded to those who answer postings (Personal interview, March 2005). Participants in the forum get 25 units when they start. If a contributor answers an open question, a few units can be attached to it. In this way, active participants can earn symbolic currency that can be used to attract competent others to answer questions. According to D19, Sun forums also employ their own engineers whose primary occupation is to answer questions posed in the developer forums (ibid.). This is to keep the forums alive. In his opinion, if a forum is not alive, it is of limited value. He concludes that some forums are better than others. Although, “even if forums give an overview of the problem, it still can be necessary to plough into quite a few deeper works in the form of articles or books that I order from Amazon” (ibid.:399). What is more, he mentions that Sun offers an official catalogue for programmers consisting of 24 best practice patterns. These are patterns showing how to build quality programmes.

Other discussion forums that he visits and to which he occasionally contributes, are based on participants’ shared goodwill. As he puts it “I have good use of the forums for my own purposes and if I come by some problems that I can answer, I answer because I know that someone sits there and sweats at the other end” (Personal interview, March 2005:443). Nevertheless, this is how far he has gone as an unpaid helper. D19 underscores that he is reluctant to give away his own source code; “it is stated in my contract that I am not allowed to do that. It’s the company’s property” (focus group interview, April 2006 :694). When it comes to utilising knowledge from open-source development communities and, in turn, contributing to them, he describes a rather pragmatic attitude. On the question of whether he uses knowledge from open-source communities in day-to-day problem solving and integrates code in his own programming, he replies, “to a very large extent” (ibid.:605). The Apache project, consisting of both large and small projects, is mentioned as an example. As a concrete

example, he mentions “to handle XML more easily, we have made a little Java file that you can toss into projects. I can use the functionality that lies there all the time, and save countless code lines every time I perform those operations” (ibid.). On the other hand, it is also indicated that such ‘knowledge machineries’ are driven by commercial agendas. He comments that “open-source is not only shared goodwill and well-meaning souls; it’s the vast and weighty actors who are behind. How shall I put it? If you can’t beat them, join them! It’s according to this principle that they rule (ibid.:533). Additionally, D19 explains that for the commercial actors within server technology, it is important to be part of an ‘idea market’. In this way, they can influence how developers produce the deployment files, job specifications and so on.

In sum, data from interviews and learning logs indicate that combinations of artefacts are applied as knowledge resources in daily problem solving. The importance of consulting colleagues at work is also emphasised. In addition, he sees knowledge acquired during education as the underlying foundation for dealing with programming-related problems. Nonetheless, in line with the research problems and boundaries for exploration, the potential of artefacts applied as knowledge resources to mediate relational properties associated with objects of knowledge is focused upon in the following discussion. The screening of data indicates the combination of the following artefacts applied as knowledge resources by D19 in a local work context: hands-on reference literature, specialised books and articles, course material, Google search engine for groups, various websites, online discussion forums including SUN developer network and code shared in open-source communities.

5.4.4 Discussion

A few ‘sorting categories’ are applied when working close to data in the initial descriptive phase. These were functional as descriptive codes for identifying and organising two major sets of artefacts applied as knowledge resources. Note that these labels are not actually explicitly mentioned by the respondent. Rather, they are constructed as meta-tags based on my own comparison of artefacts when reading through data transcripts. I have labelled these descriptive categories *interactional knowledge-laden artefacts* and *non-interactional knowledge-laden artefacts*. The former are the artefacts with embedded features that allow for a two-way flow of knowledge and information between the artefact and the user. They are dynamic knowledge sources characterised by frequent change and associated with a rather

high level of joint problem-solving activity (i.e., shared source code or discussion threads). The knowledge potential that such artefacts carry is mostly jointly produced. A key parameter distinguishing it from non-interactional knowledge-laden artefacts is its ability to respond (or mediate a response) to a user's input within a short time span; ranging from instantly to within a few days. Examples are searchable databases or threads in discussion forums for software developers. Non-interactional knowledge-laden artefacts are the antithesis to interactional artefacts and lack their dynamic features; they are not able to change or develop as rapidly. Neither can they interact with users by providing instant or slightly delayed responses. Examples associated with the latter category are printed books, articles and plain tools. They are purposeful knowledge sources, but are more or less static in nature.

The combined data sources also indicate that D19 makes use of two different approaches when seeking new knowledge, rooted in the 'associated events' experienced in the work context. The first approach is the knowledge-seeking activity that is detached from the current problem solving in the local work context. Besides coursing and acquiring certifications, the Internet is underlined as a fine and simple way to obtain information to keep updated in the big picture. This *monitoring* approach goes beyond the immediate tasks, not knowing if (or when) the searched chunks of knowledge will be applicable in direct problem solving. On the other side of the continuum, *task-related short cycles* are the short-lived trajectories involved in the immediate task at hand. However, these patterns are not clearly distinguished and may overlap and run simultaneously. Based on the data presentation in the previous chapter and the suggested categories, Figure 5 below maps applied artefacts and their associated trajectories. Drawing on the respondents' practice reports, the knowledge-seeking activity is divided into two main resource categories; local and distributed. The artefact-mediated and knowledge-intense paths, referred to by Lahn (in press) as *epistemic trajectories*, are represented under the latter resource category. D19's task-related problem solving tends to initiate knowledge seeks involving distributed resources. However, in order to explore the key artefacts' potential in mediating relation to objects of knowledge, it can be useful to isolate applied artefacts that provide some kind of interactional features. It is suggested in the theoretical section that along epistemic trajectories, there are 'openings' or invitations for participating in communities that are oriented towards objects of knowledge. It follows that the key artefacts mediating such trajectories have embedded features that allow the knowledge seeker to interact with other participants as well as the knowledge content.

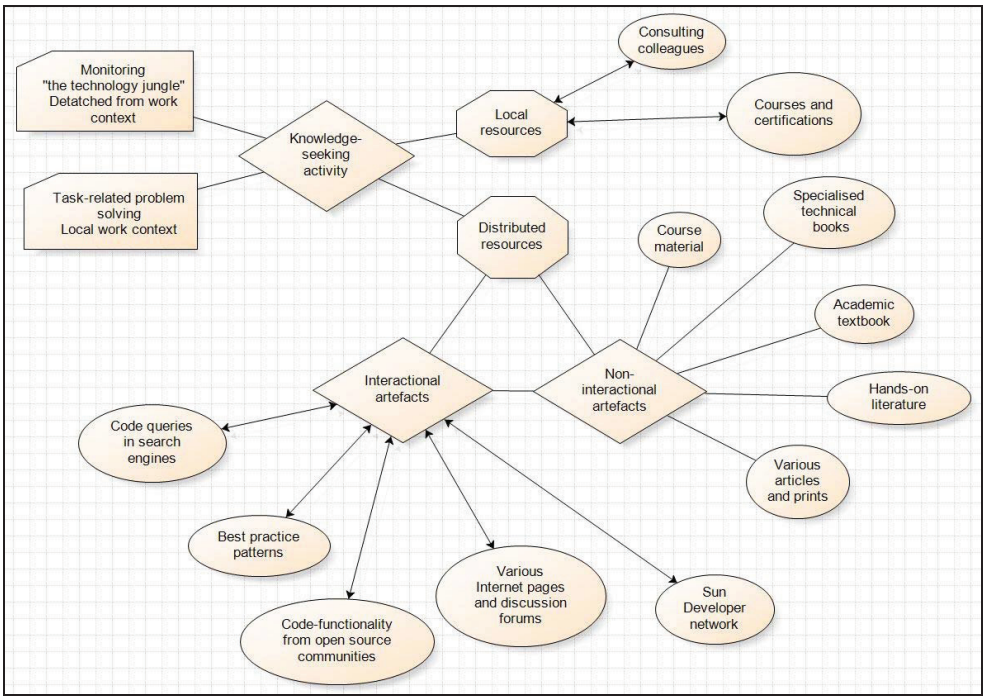


Figure 5: Map of applied knowledge resources and associated trajectories

The sphere-shaped elements in the resource map above are based on D19's reports from the workplace context (see previous chapter). However, the square and hexagon-shaped categories and associations in between are constructed for analytic purposes. The interactional category of artefacts is primarily associated with Internet-based sources, whereas the non-interactional ones are tied to more offline sources. Furthermore, the level of integration with other artefacts may also vary between these two categories. For instance, online discussion threads in developer networks can be seen as integrated artefacts that allow a certain degree of interaction with other people as well as the interfaces tying the networks together. On the other end of the continuum, the possibility of interaction is rather limited. A specialised textbook can be a valuable resource in a local context, but is not necessarily directly integrated with other resources. However, it may contain code sequences or references to online dynamic content. It follows that the distinctions made above are not absolute but rather operate on a continuum from low to high integration and different degrees of interaction. One can argue that both categories embody the necessary linguistic qualities to serve as 'externalised memory systems' and mediate knowledge and ideas between a local and distributed knowledge practice. As Wertsch (1996) noted, their structures may continue to

exist across time and space even when they are not incorporated into immediate activities. For example, a sample of programme code written down and commented on in a specialised book or article can be seen and external reification that continues to exist across time as space. It can be reproduced in its original form or re-written and adapted into a derived version. Furthermore, the same piece of code and attached comments can be mediated by artefacts on various levels of interaction and degrees of integration. At a low level of interaction and low degree of integration it is reasonable that a software developer cannot directly modify and adapt the code without converting it into a digital format. However, if the code sample appears in an integrated set of artefacts as a discussion forum, the developer is presented with more options. For instance, he can enter the forum as a member and participate in the joint problem solving around the code sample, or it can be downloaded and stored for later projects.

As indicated in the data presentation above, sample codes from open-source projects are frequently applied as a knowledge source by D19. Discussion forums and shared or open-source code are associated with a high degree of integration and high level of interaction. From the data presentation and discussion, these appear as key artefacts with embedded features that are relevant for exploring their potential to mediate objects of knowledge. Furthermore, the movements towards distributed discussion forums and shared code online can reasonably be interpreted as an epistemic trajectory (Lahn, in press). Epistemic trajectories are characterised as knowledge-intense paths in relation to mediated expert networks. For example, by posting a search string containing a code sequence in Google and following the links displayed on the results page into a developer community in which the problem is addressed, the respondent connects to distributed expert networks. In the Google Group search engine, the results commonly display hyperlinks to sources that are registered so that these can be indexed by the search robots. One can say that search engines also play the role of mediators or obligatory passage points in the epistemic trajectories towards distributed objects of knowledge. However, to locate relevant object-centred communities through posting queries in a search engine, one needs to formulate rather precise strings. Moreover, search engines such as Google make use of advanced mathematical algorithms to rank their results pages. It follows that an epistemic trajectory involving search engine queries can move in multiple directions. The pathways are thus influenced by several forces, among them what is predefined by the search engine as the most relevant results. When successful search strings are formulated, the subject can move on to relevant expert networks. Integrated

sets of mediating artefacts applied along this trajectory can be interpreted as linked to object-centred communities distant in time and space. Collected partial code can be said to move along rather flexible trajectories across contexts. However, the code bits may also be further adapted in local practice and perhaps transformed to acquire different attributes.

In sum, the key artefacts applied as knowledge resources in this case are interpreted as a set of integrated artefacts operating on a high level of interaction. Examples are search engines, discussion forums and shared source code. The next section adds a layer of description by interpreting a discussion forum's artefact-structure and potential for mediating relations associated with objects of knowledge, drawing on the propositions concerning object relations presented in the conceptual framework (chapter 2-3). Scattered around on the Internet are numerous discussion forums that openly share programme code. However, I find it reasonable to select a forum for analysis that is also referred to by our respondent. As already indicated, D19 uses knowledge from online developer communities in day-to-day problem solving and integrates code in his own programming "to a very large extent" (focus group interview, April 200:605). As the Sun Forums is repeatedly raised as an example, I chose to employ a thread from this forum as an analytic example. When defining the case borders in chapter 5.1, I argued that it can be useful to let certain features from the external environment pass through the boundaries enclosing the case, based on their significance in answering the research question. Now, in order to explore how a developer forum can mediate the relational properties associated with objects of knowledge, an example thread from Sun Forums is included within the case boundaries.

The Sun Forums structure and potential for mediating object relations

The Sun Forums mainly consist of text-based user-generated discussions. The mediating structure is conceptualised above as an integrated artefact that allows for a high level of interaction. From a social cultural perspective, a discussion forum can possibly represent reifications of ideas or function as an externalised memory system (Säljö, 2006). As distributed memory systems, discussion forums may convey "reflexive embodiments of activity" related to Wartofsky's (1979) secondary level of artefacts if the content reflects modes of action and conveys knowledge resources that are close to software developers' productive practices. However, as the text-based discussions are primarily generated by the forums' active users, it is reasonable to assume that the content is also 'flavoured' by off-topic

social talk that is not directly related to productive practice. Furthermore, I have argued that it can be problematic to conceptualise a programme code quoted in a forum as a primary artefact. A source code is not necessarily a simple tool intended to mediate specific activities in productive practices, but is rather technical and abstract in form. In the argumentative space of a developer forum, such dual features of programme code can perhaps represent mediating potential with respect to relational aspects associated with objects of knowledge.

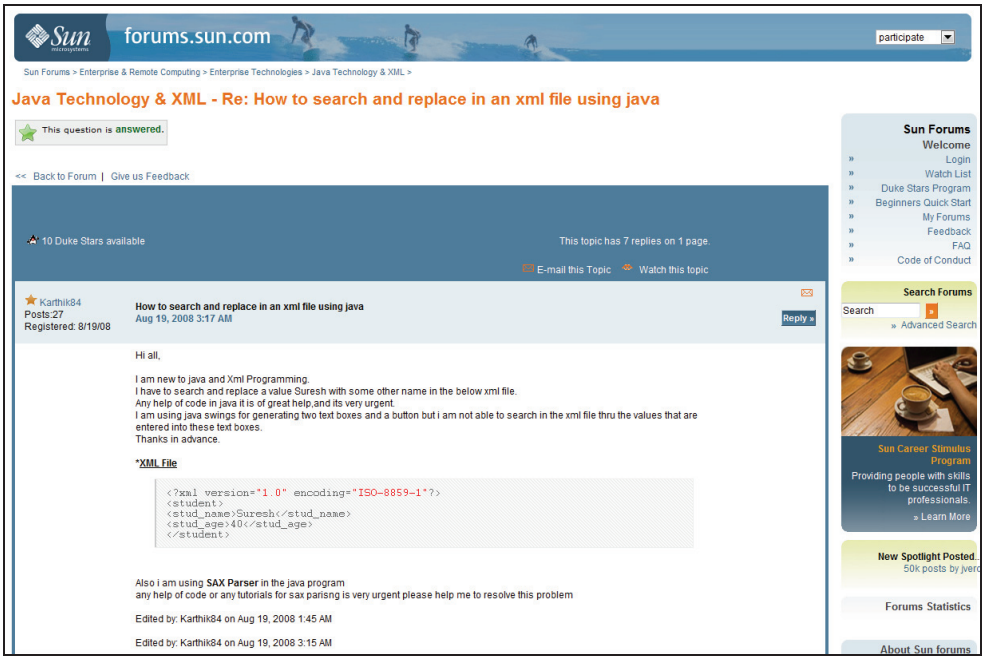
In essence, the forums are intentionally designed to assist developers and other users of Sun Microsystems technology in asking questions, finding answers and participating in discussions (http://forums.sun.com/index.jspa). The overall forum statistics as of Oct. 27th 2009 imply a large and lively resource network:

> Total Threads: 941,241
> Total Messages: 4,245,233
> Total Contributors: 977,834 (ibid.)

As an example of how this resource network is structured, one individual thread is applied in the upcoming analysis. The threads are referred to as 'topics' in the forum and commonly consist of one initial question and subsequent replies. The topics are structured following a standardised template system based on software from Jive Forums (jivesoftware.com). This implies that by looking at one topic, we get an impression of how the overall developer network is structured. The content included in each of the 941,241 topics is likely to vary extensively. However, the interface (or thread structure) tying the vast amount of messages together involves some common features which are of interest for exploring the potential to mediate relational aspects associated with objects of knowledge. Such structural features and potential are explored by using a discussion thread as an analytic example. Note that the forums are in principle available and visible for non-participants. This means that guests who are not registered users are allowed to search and browse the forum. However, to start new threads and post new messages, one needs to sign up as a member. In order to get an impression of how this developer forum is structured as an artefact, I start by describing its main features below. Thereafter, the potential for mediating relations associated with objects of knowledge is discussed by drawing on Knorr Cetina's propositions concerning object relations (see p. 44-45).

Artefact structure

Following the reasoning of Nespor (1994), activity distant in space and time can be transported into particular settings through resources and representations. Sharing programme code in a distributed environment can be seen as interaction with others on a different time-space scale. It follows that online user forums may expand the social organisation of commenting code beyond the immediate local practices by mediating programming skills, ideas and activities of individuals who are not physically co-present. The Sun forums' interface appears as the boundary between the local context of single developers and the system or resources provided within the network. This is the shared surface for interacting with other participants in the forum, as well as reading documentation and administering their membership. A typical forum page consists of various navigational links, a thread title, user posts and replies. Discussions are stored as *topics* for later retrieval.



Screen image 1: Example thread from Sun Forum

We see from this user interface that several options for interaction are provided. First, at the upper right corner there is a pull-down menu labelled 'participate'. The menu is linked to several modes of participation across the Sun network; among them wikis, blogs and forums.

What is more, one can search previous discussions, browse content, give feedback to administrators, e-mail the topic to someone or sign up to 'watch this topic'. Apart from the visible labels, another option may appear for a developer searching solutions to programming-related problems. The shaded area in the screen capture above shows XML functions that can be copied and modified for one's own purposes. This practice is also indicated by D19 when he reports to collect and re-use code from developer networks. One can argue that the code sequences that are purposely displayed within the discussion forum represent an artefact on a somewhat different level from the mediating structure in which they are embedded. According to Wartofsky (1979), artefacts operate at different levels relative to their closeness to action or practice. It is reasonable to interpret programme code as closer to developers' practice than the discussions, explanations and comments forming around them. However, in the conceptual framework adopted for this thesis, I have suggested that the traditional levels of artefacts may be problematic to distinguish and perhaps operate in a more 'blended mode' when it comes to networked, computational artefacts. In order to understand programme code and comments about closeness to practice, we can distinguish between functional and non-functional material as pointed out by Moglen (1999:2) in this quote:

> The function of source code in relation to other human beings is not widely grasped by non-programmers, who tend to think of computer programs as incomprehensible. They would be surprised to learn that the bulk of information contained in most programs is, from the point of view of the compiler or other language processors, "comment," that is, non-functional material. The comments, of course, are addressed to others who may need to fix a problem or to alter or enhance the program's operation. In most programming languages, far more space is spent in telling people what the program does than in telling the computer how to do it. The design of programming languages has always proceeded under the dual requirements of complete specification for machine execution and informative description for human readers (p. 2).

The linguistic practices of commenting code pointed out by Moglen (ibid.) indicate an inherent duality of code objects, consisting of both instructions executing computer functionality and verbal utterances aimed at other developers working on the code. The non-functional comments are thus 'objectified' and distributed as integrated descriptions within the actual programme code. This duality of code object is close to the definition of artefacts

suggested by Cole (2002), and can be seen as an essential dimension in software developers' knowledge practices. Functional or *prescriptive* code and coupled non-functional *descriptive* comments in discussion forums can be interpreted as integrated artefacts with the potential to be incorporated into developers' local practices. However, functional code is not material or physical entities in line with Wartofsky's (1979) primary level of artefacts. Their attributes are of a technical and abstract nature, specifying commands for machines' execution. On the other hand, functional material may be directly applied in the production of software programmes and can be seen as a primary concern in developers' creative practices. Non-functional material in the form of comments in source code and discussion forums are not completely detached from a primary level (or functional material) as the comments are also integrated in the programme code; although, together the functional and non-functional aspects of code seem to blend into one hybrid resource that displays both ideal and material attributes. In the discussion forums, source code and comments appear as a set of integrated artefacts with the potential to assume different roles depending on to how they are oriented in productive practices. This brings us to the next issue, namely their potential for mediating relational objects of knowledge. A potential for mediating relations in this context implies an artefact structure that somehow 'triggers' individual developers' urge to participate in further development of joint objects of knowledge. In addition, according to Knorr Cetina (2001a), for an object relation to form, it should be sustainable over some time and the subject ought to have agency with regard to the evolving object. These issues are discussed in the following paragraphs.

Potential for mediating relations associated with objects of knowledge

So far, we have seen that the discussion forums within the Sun Developer Network provide an infrastructure for displaying and storing discussion and comments on development as well as code samples. When code samples are shared in online discussion forums, they can be applied by peers as purposeful knowledge resources. This practice is also indicated by the data presented before. The previous discussions around the shared code in the developer forums are likely to be stored in a discussion thread for later retrieval. Non-functional comments can then be useful in understanding how the code has evolved in previous joint efforts. If the knowledge seeker has agency towards the shared code, it may also embody the potential to transform into an object of knowledge. For example, if a developer stumbles upon a source code rendered visible in a discussion forum that appears attractive and especially relevant to

his own work, he may sign up as a participant in the community and contribute to the further development of this piece of code. In this way, the S-Ko dynamic suggested by Knorr Cetina and Brugger (2002) can potentially be initiated. The discussion forum postings may signal what is still lacking in the code, i.e., by displaying software malfunction or faulty syntax. If the participation persists over a time period, an artefact applied as a knowledge resource can hence potentially transform into an object of knowledge. However, the time interval of the example thread showed above stretches from Aug 19, 2008 to Feb 16, 2009 and is currently tagged ‘answered’. A small programme is posted as a solution to the problem stated in the beginning of the thread:

```
import java.io.File;

import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.transform.Transformer;
import javax.xml.transform.TransformerFactory;
import javax.xml.transform.dom.DOMSource;
import javax.xml.transform.stream.StreamResult;
import javax.xml.xpath.XPath;
import javax.xml.xpath.XPathConstants;
import javax.xml.xpath.XPathFactory;

import org.w3c.dom.Document;
import org.w3c.dom.NodeList;
import org.xml.sax.InputSource;

public class BasicXMLReplaceWithDOM4J {
  static String inputFile = "C:/student.xml";
  static String outputFile = "C:/studentRenamed.xml";

  public static void main(String[] args) throws Exception {
          // Read xml and build a DOM document
          Document doc = DocumentBuilderFactory.newInstance()
                          .newDocumentBuilder().parse(new
InputSource(inputFile));

          // Use XPath to find all nodes where student is named
'Suresh'
          XPath xpath = XPathFactory.newInstance().newXPath();
          NodeList nodes = (NodeList)xpath
                  .evaluate("//stud_name[text()='Suresh']", doc,
XPathConstants.NODESET);

          // Rename these nodes
          for (int idx = 0; idx < nodes.getLength(); idx++) {
                  nodes.item(idx).setTextContent("Suresh-
Renamed");
          }

          // Write the DOM document to the file
          Transformer xformer =
TransformerFactory.newInstance().newTransformer();
          xformer.transform(new DOMSource(doc), new
StreamResult(new File(outputFile)));
  }
}
```

Table 8: Sample XML programme from forums.sun.com

From this sample, we see that the author included explanatory comments in between the lines of code by using double slashes (//). For example, the comment on line 18, “Rename these nodes”, invites other developers to customise the programme for their own purposes by

specifying a local node list. Besides the explanatory comments, parts of the code are also readable for other knowledgeable developers as the programme's markup language (XML) uses English words in its syntax (i.e., 'import' and 'outputFile'). However, for the constructs to appear in meaningful sequences, programmers need basic knowledge about the grammatical rules of the applied markup language. Subsequent replies confirm that this programme helped to solve the problem. This means that the object towards which the thread is oriented is considered a closed topic by the moderators of the forum. In Knorr Cetina's terms, the object is no longer open-ended, signalling deficiencies for the participants to interpret and act upon. One can also argue that the time span in which participants engage in the open-ended problem solving around this object is too short to fall under Knorr Cetina's definition of an object relation. Yet, if this programme code were incorporated into one of the participants' local development projects, it may still function as a piece of another knowledge object.

In the topics heading (Screen image 1), it appears that "10 Duke Stars" are available for participants of the Duke Stars developer programme (developers.sun.com/forums/). This is a form of symbolic currency or points provided as an incentive to encourage forum users to answer questions. The intention is for users to assign a 'reward' to their initial problem formulations and thus motivate other forum members to help them to find solutions. When a question is answered, Duke Stars are granted to participants contributing with the best answers. In this way, active contributors can reach different Duke Star levels (Bronze, Silver, Gold and Platinum) and receive 'Star Badges' which are attached to their user profile. To my understanding, the points that contributors earn by sharing their technical knowledge can later be applied to obtain help from other developers. A symbolic currency thus circulates in the system, stimulating knowledge sharing among peers. This suggests that the 'wanting structures' that trigger participation in this forum are not necessarily inherent in the shared code itself. In order to attract developers' participation in joint problem solving, an external incentive system is created. It follows that the structural affinity between the elements in object relations suggested by Knorr Cetina (2001b), do not readily apply in this context. I do not find it reasonable to interpret the incentive-triggered participation in discussion threads to be a result of open-ended structures that 'uniquely match' the unfolding characteristics of knowledge objects. However, we cannot rule out the possibility that the programme code embedded in discussion threads will signal the deficiencies associated with objects of knowledge that somehow match the contributing developers' structure of desire. Nonetheless,

as I read Knorr Cetina's theoretical propositions (see chapter 2.3.1), sustaining object relations is rather characterised by experts' personal drive which is founded on extended engagement with open-ended objects, and not by brief participation in solving other developers' technical problems by means of a symbolic incentive system. These issues point towards potential limitations with the notion of knowledge objects and related problems with adopting a one-sided interpretation of such entities. In the expert systems of a knowledge economy, there might be various incentive systems outside the objects themselves that interact and influence the 'desire' to participate in developing joint objects of knowledge.

5.5 Transition to the next study

As this thesis incorporates another thematically-related empirical study, I find it appropriate to provide an overall concluding discussion after the next case is presented. The final included discussion in chapter 7, examines the degree to which the aims of this thesis have been achieved, assesses its original contribution to knowledge and reflects on its limitations.

Considering the field of software development's emphasis on argumentative problem solving and knowledge sharing in distributed expert networks, joining the active meaning making and thereby contributing to processes of intersubjectivity might be important constituents in the collective knowledge machinery. It is reasonable to assume that the progression along epistemic trajectories also involves sense-making aspects. Perhaps participation in the joint construction of code in developer networks represents potential for stepping into an expanded space of intersubjectivity. This brings us to the central theme investigated in the next case; processes of intersubjectivity in open-source developer networks. As a transition to the next study, let us spend a few lines reflecting on the possibility of expanding intersubjectivity on D19's epistemic trajectory. The respondent reports that it is "sufficient to paste the error message in Google and search in groups and then I get someone who has already discussed this among them in these groups and I'm likely to get the answer" (Personal interview, March 2005:394). This quote indicates that local problem solving may intersect with (or be informed by) the knowledge development of competent peers in mediated expert networks. Moreover, a related problem may previously have been discussed *among* other developers and he is *likely to get the answer*. This can be interpreted as an example of a distributed form of Rommetveit's (1979a) notion of *shared prolepsis*; anticipations serving as a catalyst for sense-making across contexts. Faith in his peers leads D19 to search for solutions in mediated

expert networks. The knowledge-seeking actor thus enters intersubjective spaces situated in online discussions among peers on different time-space scales. If he finds a previous discussion thread online, in which the respective problem is discussed and solved, he can read through the argumentation. Moreover, the stored communication may perhaps include 'proleptic incidences' that occurred as a part of the other developers' active meaning making; embodying the potential to initiate sequences of anticipatory comprehension for the knowledge seeker. In this way, what is previously made known within the expert network can transcend what was said 'there-and-then'. Furthermore, the stored argumentation between other developers may be proleptic in omitting pieces of information and implying solutions that invite the knowledge seeker into an expanded intersubjectivity. In this process, the local developer has the opportunity to creatively infer information from other people's semiotic interactions, even when not taking part in the actual conversation. Bringing mediating technology and object-oriented peer interactions on different time-place scales into Rommetveit's framework for intersubjectivity represents uncharted territory (see chapter 2.1). The next study inquires into aspects of these processes in the distributed communication among participants in an open-source development project. These issues also represent indications for further research and development of a framework for understanding processes of intersubjectivity in the technology-mediated contexts of a future knowledge society.

6 Processes of Intersubjectivity in open-source development networks

[Empirical study II]

6.1 Introducing open-source development networks

A vast body of research over the past few decades has contributed to the understanding of information networks. For example, through his trilogy on the information age, Castells (2000, 2004; 2006) provides comprehensive insight into the networked interrelations characteristic for a wide range of contemporary states of affairs. A central factor in the new society described by Castells is the information technology revolution. Within this new social order, network relations are seen to transform modes and processes of sociality, knowledge production, communication, power structures, culture and so forth. It follows from the extensive analysis of Castells (vol. 1) that major functions in our society are increasingly dependent on, and organised into networks. However, notions as production, resources and property may appear ambiguous in this discourse. Benkler (2006) maintains that in the networked society, "a new model of production has taken root: one that should not be there, at least according to the most widely held beliefs about economic behaviour" (p. 59). Moreover, he suggests that networked environments allow for innovative modalities in relation to manage production. Effective practices of transforming resources within these social structures are identified and portrayed as rather decentralised, based on shared collaborative efforts and non-proprietary. This is what Benkler (2006) refers to as *commons-based peer production*:

> Any production strategy that manages its inputs and outputs as commons locates that production modality outside the proprietary system, in a framework of social relations. It is the freedom to interact with resources and projects without seeking anyone's permission that marks commons-based production generally, and it's also that freedom that underlies the particular efficiencies of peer production (p. 60-62).

Networked commons are thus resources quite opposite to properties, being essentially open for access and free to use by anyone. Free or open-source software projects are frequently used as examples of projects that do not rely on monetary markets and traditional organisation patterns to produce high-quality output. Of concern for the research theme of this thesis are the processes 'in between' actors in networks, and the joint understandings forming around the shared resources. Do the shared resources take on social properties in interaction and

become open-ended, dynamic objects of knowledge or more restricted carriers of technical expertise in the field?

The open-source model in software development is considered by Miettinen (2006) and others to be a relatively new form of community-based, distributed knowledge creation. According to Von Hippel (2005), such projects are relatively well developed and represent a flourishing form of Internet-based innovation communities. An example is the Apache mod_perl project, consisting of a worldwide network of more or less dedicated collaborators. The objective to which they are committed is the development of a software module that integrates Perl programming with the Apache HTTP (Web) Server. This module is constantly changing and unfolding as new patches and upgrades are released at a high frequency. Developers spend a tremendous amount of time and effort on this activity as unpaid volunteers. Numerous skilled programmers and users scattered around the globe work jointly on an advanced object via internet technology. In order to interact and work on a common set of objects in a distributed software developing community, it can be argued that the actors must share some categories of understanding. The analysis conducted in chapter 5.4 indicates that open-source development projects are dynamic and rich sources of knowledge adapted and applied across contexts in the field of software development.

What then is 'open-source'? Is it a philosophy, process, fact or occurrence that can be observed and explained? The term *open* is associated with a form of access as opposed to being closed or restricted. Furthermore, the concept 'open-source' is commonly associated with a trend in software development, promoting access to the programmes' source code. In his book *The Success of Open Source*, Weber (2004) explains open-source projects as innovations in social organisation around a distinct notion of property. Property in conventional terms is associated with having the right to use it, although its use may perhaps lead to the exclusion of other people. Property and property rights are also regarded as fundamental cornerstones of capitalist societies (John Scott & Gordon Marshall, 2005). However, property in open-source communities is of a somewhat different nature and is said to be constituted by the right to distribute, not to exclude (Weber, 2004). In this context, internet technology has shown a powerful infrastructure for interactive communities to form around diverse modes of production and knowledge processes (ibid.). However, the ideas of open-source are not restricted to software production. Examples of application in other areas that has been influenced by open-source philosophy are the free online encyclopaedias (wikis)

and the Biological Open-source (BiOS) movement. Nevertheless, in such communities and among researchers, there has been some confusion about what open-source really means (Feller & Fitzgerald, 2000). Open-source Software is not to be seen as freeware or shareware products. It is not necessarily available free of charge. However, the free software tradition has a long history and is regarded as an origin of the more recent Open-source movement. As a clarification, and to provide a wider context or a macro-perspective for the Apache mod_perl case, I describe the underlying ideas of the open-source movement in the next paragraphs.

The concept of open-source as applied in software development can be seen as methodologies and forms of practice that maintain open access to source code. As a moving force in software production, open-source is characterised by a standard-promoting body such as the Open Software Foundation (Ince, 2001). The underlying idea is that the code of software systems ought to be obtainable to everyone. In this way, skilled individuals and groups can modify, write add-ons or support software for the product. Functional applications are to be released under a licence in compliance with the Open-source Definition (OSD), expressed by the Open-source Initiative[7]. Accordingly, the programme distributions should not only be in compiled, executable form, but also include the actual source code.

What are the criteria with which a source must be in compliance to be distributed under open-source licence? As a brief introduction, I present a condensed form of the definitions and their implications, adapted from Bruce Perens (1999) and the current 1.9 version (Appendix 9.4). The criterion of *Free Redistribution* entails that the licence should not restrict sales or free distribution. The software may be copied any number of times, but the source code has to be included in the distribution or made freely available. This is regarded as necessary for further development (i.e., repair or modification). In addition, the licence must allow for free redistribution of all modifications and derived works. No discriminations may be stated in the licence, no groups or people can be locked out, neither may the licence restrict application in a specific field of endeavour (i.e., commercial use cannot be excluded). The rights associated with the programme should affect all actors to whom it is redistributed, with no need for subsequent licensing. A programme claiming to be open-source must not be licensed as a

[7] The Open-source Initiative is a non-profit corporation that manages the Open-source Definition (OSD) and the community-recognised body for reviewing and approving licences as OSD-conformant (http://www.opensource.org/).

dependent component of a particular (larger) software distribution (i.e., an Open-source product cannot be given away free on the premise that it is applied on an IBM computer); neither should the licences require that other software distributed along with it be required to be open-source.

6.2 Issues and borders for exploration

> Very often, there is someone out there who has solved a similar problem before. Then it is handy to see how they were thinking, even if you may not do it in the same way yourself ... it's about getting some ideas ... to see how his programme is working, you get the source code. Then you can see how it's done (Computer engineer participating in focus group interview, April 2006, ProLearn project).

The screening of artefacts applied as knowledge sources in the previous case, indicates that shared source code plays a role in connecting software developers to distributed expert networks. The above extract points to relevant issues to be investigated further in the present case; namely, how partially shared understandings form around shared source code. "Someone out there" with his/her conscious mind has embedded his/her ideas and problem solving into shared or open-source code. What is the role of intersubjectivity in a distributed expert network? In the previous study, it appears that code-sharing scenarios rely extensively on integrated sets of mediating technologies. Participants discussing code in an online forum for developers are decentred, but in one way or another "their thoughts and experiences are dialogically interwoven with those of their other" (Crossley, 1996:12). This connection is partly achievable by means of shared code samples and written comments (i.e., utterances as comments in the code, in mailing lists or web-based discussion forums). Moreover, shared codes are commonly developed in a particular programming language (i.e., java), by means of a compiler or interpreter. It follows that the inscriptions here serve as common bases for sharing ideas and thoughts. In this way, ideas can be passed among developers who are distant in time and space. Transcending different private worlds in this context is thus likely to include a set of mediating artefacts. For example, open-source programmers may collaborate by using asynchronous discussion forums in distributed expert networks in which individual participants are not physically co-present and enter the shared space on different time-place scales.

For the purpose of a more detailed analysis of intersubjective interactions, I find it practical to demarcate a few aspects for exploration. In order to explore and analyse processes of intersubjectivity in an open-source developer network, it can be useful to screen participants' communication in the developer forum for breakdowns; where something goes wrong. This may involve selecting 'hotspots' in the interaction among different sub-groups within the network, where the focus is on 'restoring order' to continue with their joint problem-solving activity. The theoretical assumption sketched out at the beginning of this thesis suggests that partial intersubjectivity around shared objects of knowledge is important in order to move forward with joint problem-solving activities in expert networks. This entails the construction, development and repair of a (partial) common ground. The rationale for selecting sequences following breakdowns in the communication as foci for analysis is the assumption that such breakdowns have significant consequences for the learning processes involved and how actors make sense of other participants' incomplete utterances (shared prolepsis). As Matusov (1996) also points out, it is not necessarily consensus and agreement that is the key factor in understanding intersubjectivity, but the disagreements and 'broken' communication are central aspects to account for when studying the phenomena.

The main issues addressed in this case can be summarised as follows:

A. Topical issue: processes of intersubjectivity in open-source software development.
B. Research question: Which participants belong to the central and peripheral sub-groups in the distributed developer network? What is the role of prolepsis in relation to shared objects of development in the mediated communication among participants in the sub-groups?
C. Assertion: Proleptic instances trigger the construction of new understandings. The associated sense-making of incomplete utterances varies according to participants' relative placement in the core and peripheral regions of the overall network structure.

6.3 Social network approach to computer-mediated communication

For structuring purposes and to seek out more formal descriptions of the set of social relations characterising the mod_perl community, this section provides grounding for the network analysis conducted. Taking into account the computer-mediated environment connecting the

globally dispersed participants, a meaningful analysis of interactions may involve seeing beyond dyads of single developers and examining patterns of social relations or what is commonly referred to as Social Network Analysis (SNA) (Garton, Haythornthwaite, & Wellman, 1999; Scott, 2000; Wasserman & Faust, 1994). This approach can be informative in describing the network of communication among participants in the project and exploring its structure; is it a dense or more sparsely knit network? Are there a central core and peripheral sub-groups present? What are the consequences of participants' location in the network for processes of intersubjectivity? I regard these issues as relevant for the overall theme and research questions of this thesis.

SNA is a collection of methods derived from mathematics and graph theory aimed at analysing relations among interacting units (Hanneman & Riddle, 2005; Scott, 2000; Wasserman & Faust, 1994). Analysts applying this approach are usually committed to analysing social relations between actors rather than attributes of individual actors. In this chapter, I focus on the basic relational concepts and models associated with SNA relevant for describing the network structure of the mod_perl developer community. Subsequently, I discuss how this approach can complement an ethnographically informed content analysis of sub-groups' messages in the project's e-mail archive.

Let us go through the basic assumptions underlying the SNA perspective. According to Wasserman and Faust (1994), the unit of analysis is not single individuals, but rather structural entities composed of a set of nodes (e.g., a collection of individuals) and the links between them. Essential components of a social network are accordingly the actors or nodes and their relations. Structures of relations are commonly represented in both graphical and tabular form in an *adjacency matrix* (Wasserman & Faust, 1994:21-35). The matrix is rather simple, containing as many rows and columns as there are actors in the data set. Depending on the scale of measurement (nominal, ordinal or interval) the cells can describe a variety of relations. As with conventional statistical analysis, different levels of measurement determine what operations we can perform on data and, consequently, the inferences to be made.

For descriptive purposes, simple calculations and with respect to the research questions of this thesis, a binary matrix might be the best choice. The matrix format is also well-suited to performing computer-assisted calculations. I will return to discuss this issue at the end of the

chapter when defining the actors, relations and boundaries to be studied in the mod_perl project.

It is important to note that network data are not necessarily atypical. The datasets may also be explained by methods commonly used in statistics (Hanneman & Riddle, 2005). However, the data sets developed using the linking procedures of network theory have quite a different appearance from the more conventional rectangular data. This implies that as analysts, we observe and interpret data from a different viewpoint. The network stand leads us towards a relational perspective, seeing actors and their relations instead of actors and attributes. According to Hanneman & Riddle (ibid.), a basic network data set consists of "a square array of measurements" (p. 39). A key feature is that the rows and columns of the array are the same cases or subjects. Furthermore, the cells of the array are functional in describing a defined relationship (relational tie or link) between the actors. These links between actors are regarded as important channels for transferring resources in the network (Wasserman & Faust, 1994). Analytic tools for displaying and understanding such relations among actors can be useful for describing and structuring data for a more informal analysis of intersubjectivity among participants in the mod_perl project.

Garton et. al (1999) identify two main approaches from which to choose when studying social networks; ego-centred and whole networks (pp. 81-83). The prior analyses the network from the perspective of single actors, placing them as centres of their networks. The focal point is thus on the social network of particular persons. The latter approach is more inclusive in modelling the ties between all actors in the bounded system. This means the "whole network based on some specific criterion of population boundaries" (ibid.:82). Examples can be various interest groups, academic peers within a specific field, bilateral relations among firms and so on. Furthermore, both the occurrence and non-occurrence of relations among all members of the population are considered. The resulting network structure represents the ties (or relations) that all participants sustain with all other participants within the defined population. The above represents two modes of examining communication links among actors in the mod_perl project. According to Garton et al. (1999), a whole-network analysis is the most reasonable choice if one is to identify members of the network who are less connected and those who act as central figures. It is argued that such positions emerge through analysis of the whole network. However, to my understanding, the two ways of studying networks are not mutually exclusive. The ego-centred network can be examined as structures embedded in

whole networks. The network characteristics of the mod_perl project and the rationale behind formal measures are further developed in relation to data in chapter 6.5.1.

6.4 Adapted ethnography

In a 'classic' sense, the term ethnography is often coupled with anthropological research and studies involving modes of participant observation in which culture and social organisation of groups (or communities) are of key concern. An *ethnography* could also connote the written product of such study (Calhoun, 2002). What distinguishes ethnographic inquiries from field work and other participant observation studies is, according to Lapan (2004), the prominent focus on culture. It is reasonable that object-oriented communities operating in 'virtual space' can be seen as particular cultural sites. For instance, in open-source projects, cultural integration and social bonds may perhaps form around the collaborative creation of programme code. On the other hand, as Grenfell (2006) notes, these sites can also become "fields of struggle in which the production, legitimation and circulation of particular forms of knowledge and experience are central areas of conflict and resistance" (p. 415). As ethnographers, one might "seek to identify the tensions which exist within particular social and cultural group(s) and the ways that these are resolved or continued" (ibid.).

Ball and Ormerod (2000) conduct a supportive meta-study investigating a range of current synopses of ethnographic research and, on that basis, suggest ten features of what they portray as "a prototypical case of ethnography" (quoted from ibid.:150):

- *Situatedness:* data are collected by a participant observer who is located within the everyday context of interest (e.g., a community of practitioners).
- *Richness:* the observer studies behaviour in all manifestations, such that data are gathered from a wide range of sources including interviews, team discussions, incidental conversations, documents as well as non-verbal interactions.
- *Participant autonomy:* the observees are not required to comply in any rigid, pre-determined study arrangements.
- *Openness:* the observer remains open to the discovery of novel or unexpected issues that may come to light as the study progresses.
- *Personalisation:* the observer makes a note of his/her own feelings in relation to situations encountered during data collection and analysis.

- *Reflexivity:* the observer takes a reflective and empathetic stance in striving toward an understanding of the observee's point of view; the observer taking account of, rather than striving to eliminate, his/her own affects upon the behaviour of the observees.
- *Self-reflection:* The observer must acknowledge that any interpretative act is influenced by the tradition to which he/her him/herself belongs.
- *Intensity:* Observations must be intensive and long-term, such that the observer should become immersed in the ongoing culture of the observee's environment.
- Independence: The observer must not be constrained by pre-determined goal set, mindset or theory.
- *Historicism:* The observer aims to connect observations to a backdrop of historical and cultural contingencies.

The features above can be helpful as 'loose guidelines', but need to be adapted to an online study. According to Grenfell (2006), ethnographers typically aim to "interpret the cultural events, patterns, symbols and meanings of social groups by paying particular attention to the temporal and spatial context in which each group is situated (p. 416)." Moreover, its occupation is the production of professed thick descriptions, accounting for "the behaviour of the groups and to make explicit what may be tacit or intuitively understood or only partly realized" (ibid.). However, inquiries into cyber cultures or virtual communities imply paying special attention to somewhat unorthodox arenas. The field site is likely to be coupled with constraints, lacking the properties associated with physical space and the presence of real life face-to-face interactions. As a consequence, innovative and adaptive methods of data collection have evolved over the past decade, applying the techniques of anthropology and ethnography to the online world (John Scott & Gordon Marshall, 2005). With respect to the technology-mediated nature of the new field sites, 'thick descriptions' can possibly be constructed by utilising (and adapting) some classical notions of ethnography. Let us have a closer look at variations of online ethnography.

The rapid growth of internet communities over the last decade is pursued by researchers in many fields (ranging from anthropology to informatics). New crossbred methodologies challenge, extend and make use of traditional concepts. Some researchers call for a *virtual* or *cyber ethnography* and the study of *virtual communities* (Crichton & Kinash, 2003; Rheingold, 1998). The shared endeavour is to conduct qualitative studies of online communities and their interactions. Inquiries of this kind may involve:

> participant observation in chat rooms, multi-user domains, email distribution lists, forums, and bulletin boards, as well as techniques of online interviewing, conferencing and other types of computer-mediated communication. The overall aim of cyber-ethnographic studies is to immerse oneself in the virtual world that the participants have created, in order to understand how they experience social interaction and devise ways of regulating social order … cyber ethnography has created an original and effective way of finding out about interaction online, and so has increased our understanding of a relatively new phenomenon of social life (J. Scott & G. Marshall, 2005: "cyber ethnography").

It appears that the new and probing word that needs further explanation is *virtual*. What does it connote, and what are the consequences for the inquiry? In modern computer language, virtual is described as something not physically existing but made by software to appear to exist (Allen, 1999). Rather than being directly engaged and observing social groups' behaviour in the material world, virtual ethnographies process the socio-cultural behaviour of online communities (ibid.). As portrayed earlier in this thesis, open-source communities encompass several unique features, making them attractive cases to study knowledge practices and processes of intersubjectivity around joint objects. Being distributed and virtual, they operate on different time-place scales with no "physical location or centre" (Grenfell, 2006:421). Membership is, for the most part, fluctuant (participants may enter and disappear), and based on free will, distributed peer review and transparent development processes (opensource.org). Nevertheless, the communities appear to generate rich knowledge sources in the form of code objects and comments that are freely available, adaptive and transferable across various settings.

When describing the mod_perl scenario, the centre of attention shifts, to use the words of Attila Bruni (2005), "from Place to Space" (p. 359). The online arena for open-source development is previously described in this thesis as a distributed or virtual community in which human interaction is made possible by technologies. To grasp the complex dynamics of a technology-mediated online community, I observe the mediated communication of the specific group of developers in a limited timeframe, and produce a description. As an observational field study, it seeks to collect first hand information about social processes in a naturally occurring context (Silverman, 2001). This approach also opens the way for a more

flexible research design, increasing the chance of coming across unforeseen issues (ibid.). However, the specific online community that is focused upon involves important distinguishing features that have to be accounted for in the research methodology. First of all, the targeted community is not physically co-present. Interaction and communication is mediated by information technologies. Traditional concepts of ethnography may not apply to this context. When conducting observations in an online environment, observations are unlikely to gather data from co-located, face-to-face interactions. The field site hence provides a ground for somewhat limited observations, as one cannot observe 'real people' in virtual communities (Wittel, 2000). It is reasonable to say that collaboration and interaction of open-source developers are in a physically distributed mode, and can hardly be observed by merely applying traditional concepts of ethnography. In order to understand intersubjective interactions in this environment, I will argue that it is adequate to engage in ethnographically informed observations which also accounts for the text-mediated communication necessary for the functioning of the open-source development project. Other studies concerned with interaction patterns in open-source communities often refer to project mailing lists and the actual source code as the key artefacts (Lanzara & Morner, 2005). According to these studies, the source code and mailing lists are found to be 'dynamic vehicles' assisting distributed communication, organisation and coordination. Participants' typical activity in the projects is to write programmes and discuss code functionality (ibid.). Project-related discussions are commonly documented in mailing lists or discussion threads on websites. However, methodological approaches to study processes of intersubjectivity in this context are not extensively developed in previous work.

6.4.1 Approach to text-mediated interactions online

In the following paragraphs, I account for a few relevant approaches to *text* as data. The specific kind of written text that I am addressing here, are words that have been recorded in electronic databases without intervention from researchers (including me). Textual data in this case is thus drawn from what developers 'do' with words rather than from accounts they give of themselves. Analysis is hence coupled with some restrictions. First, these data will not be treated as depictions of subjects' inner experience nor with some true correspondence to reality. As Silverman (2001) notes, "the role of textual researchers is not to criticize or access particular text in terms of apparently 'objective' standards. It is rather to analyze how they work to achieve particular effects to identify the elements used in the function these play" (p.

121-122). A few potential advantages of textual data are also mentioned (ibid.); *Richness;* close analysis of written text can reveal presentational subtleties and skills. *Relevance and effect;* written texts are likely to influence how we see the world, other people and how we act. *Naturally occurring;* texts can document participants' activities without researchers interfering questions. *Availability;* texts can be easily accessible and are not necessarily closed by access or ethical constraints.

In the case of open-source development projects, texts are important foundations for the community and are in principle obtainable for anyone. However, even if they are accessible, some documents may still be restricted by the nature of the content as they are written in highly advanced technical terms and require some background knowledge to be able to be read and understood.

The term ethnomethodology is often associated with alternative sociological analyses and approaches to the study of human communication, focusing on the exchange of utterances 'in-situ' and the use of participant observation. As Roulston (2003) notes, "rather than study retrospective accounts of members' actions (as is common to interview studies), researchers using approaches informed by ethnomethodology seek to study the ongoing achievement of social practices" (p. 141). Of primary concern are the routines and details structuring human conduct. By means of phenomenological awareness and a concern for customary social practice, it is a study of methods, asking "not why, but how" people get things done (Watson & Goulet, 1998:97). How do members of a group or community make sense of their social world? An underpinning postulation within this perspective is that some joint understandings are taken for granted by group associates situated in social contexts, and since they are so profoundly (tacitly) understood "people don't even think about why they do what they do" (Patton, 2002:111). Conducting research on processes of intersubjectivity informed by this framework might focus on rendering the group's tacit knowledge and partially shared understandings explicit. I assume that participants in an internet-based community for software developers have some 'unspoken' knowledge prior to the interaction. Without any shared fragments of socio-cultural biographies, such as a sufficient understanding of the programming language, the object of activity and the ground rules for posting messages in the discussion forum, conversations and other facets of sociality may perhaps break down. If we apply Rommetveit's (1974, 1979a) notion of intersubjectivity, the situational joint understandings are grounded in the social realities that participants bring to the situation,

forming a partly shared social world. The discussion suggests that intersubjectivity is a premise to engage in meaningful communication. How to elucidate and make explicit such presupposed, taken-for-granted understandings is a key issue for analysis. The 'micro cosmos' of social life around shared objects of knowledge in the open-source project can then be seen as continuous accomplishments through the usage of language. Furthermore, the application of linguistic utterances in joint activities is underlined by Patton (2002); through language, we learn, seek abstract knowledge, act jointly to solve problems and so on.

Two recurring and basic notions in ethnomethodology are *reflexivity* and *indexicality*. The former is "one of the constitutive problems of modern philosophy and social science, rooted in the question of whether and how persons can know the world with any certainty" (Callon, 2002). Garfinkel (1967) diminishes the problem by conceptualising reflexivity as an integral process in social actions. Social order is seen as a result of conversational processes; reflexive and constituted in talk. For instance, as analysts, we tend to assume that we observe social realities from outside and thus describe a pre-existing social order. On the other hand, for ethnomethodologists, to describe a social context is also to create it. The same argument holds for individual partakers in social interactions. Through participation in conversation and direct interactions, social realities are seen as continuously constituted. The process of reflexivity also articulates ethnomethodologists' awareness of the ongoing construction and re-construction of meanings sustained in their interpretive work. When a 'label' or a category is assigned to a segment of social life, our understanding may be transformed by the very category constructed. Consequently, such categories are, according to Pfohl (1994), creations of situational features rather than pure interpretive processes.

When analysing social interaction, indexical terms (i.e., this/that, here/now) are in general considered problematic as their referents and accuracy vary with circumstances (Heritage, 1984; Roulston, 2003). What is commonly known about such terms is relative to social context. However, Garfinkel (1986) suggests a slightly different approach, which is to investigate indexicality not as a problem to be solved, but rather as productive phenomena when studying people's methods. From this point of view, sociality depends upon indexicality; the capacity to provide a jointly understood context for speech or actions. How, for example, do developers of an open-source community make use of indexical terms as a resource to ground their communication?

It follows from the above that all-embracing and general definitions are problematic. Sense and meanings rather emerge "from reference to other words and to the context in which the words are spoken" (Marshall, 1998a:1). For example, if you ask 'What do you mean?' about another person's statement, the same question can be asked indefinitely to whatever answer is given. Language, in this sense, is indexical, closely coupled to applied situations.

6.4.2 Textual data sources

Primary textual data collected in the community of mod_perl developers are summarised as follows :

- The project mailing list's archive, containing discussions related to development of the mod_perl core. A text-based archive from January 2001 to present day is available from http://mail-archives.apache.org/mod_mbox/perl-dev/. Information letter about the research project was sent to the mailing list on July 2009 (Appendix 9.8).
- Various written documentation openly available in the community (http://perl.apache.org/).

Data source	Data type	Time collected	Comment
Mailing list archives	Text	2008-2009	Discussion forum for users and developers.
Documents and literature	Text and graphics	Continually	Community-written documentation

Table 9: List of data sources

The textual data sources in the above table are publicly available on the Internet. The mod_perl archive consists of accumulated asynchronous communication. This means that message threads are stored in a database as a shared knowledge resource. During the period of

observation, I was thus able to read through previous postings without disturbing the 'naturally occurring' communication in the project. Mann and Stewart (2000) refer to such a practice as *lurking* and suggest that it as an advantage of internet research; though ethical issues may arise when collecting online communication as the research involves human subjects. Researchers posting or 'lurking' in online communities might be perceived as intruders by the participants (Eysenbach & Till, 2001). When conducting research on people, ethical aspects are primarily associated with informed consent, privacy and confidentiality. According to Eysenbach & Till (ibid.), in order to determine if informed consent is needed, researchers should "decide whether postings on an internet community are private or public communications" (p. 3). This distinction is considered important since informed consent would be required if the interaction of participants takes place in a 'closed room' or private context. On the other hand, research in open communities can, according to Miller (2002), be justified without obtaining informed consent when using "publicly available information" (p. 109) . Still, with respect to the transparency and public nature of the mod_perl project, I decided to send an informative letter introducing myself and explaining the study's purposes (Appendix 9.8). In order to verify that the above issues have been handled correctly, the Norwegian Social Science Data Services (NSD) was consulted. The data gathering, privacy and research ethics of the mod_perl case were reported to NSD as of 07.11. 2008 and found adequate by their advisors (project nr. 21718).

6.5 The mod_perl developer scenario

The inquiry into the world of mod_perl starts with a 'thin' case description of the developing project, and subsequently adds a layer to the description by interpreting the overall network structure and identifying possible clusters or sub-groups within the network.

The mod_perl project is licensed under the Apache Software Licence (http://www.apache.org/licenses/). The 2.0 licence is distributed along with the source in a text file, defining the terms and conditions for use, reproduction and distribution. Mod_perl is an optional module that can be installed on Apache web servers (Spainhour & Eckstein, 2002). This means that it is intended for servers running the Apache application. It follows that the module cannot run as a 'stand alone' application, but is dependent on other pre-installed software. The prerequisites to run the mod_perl module are the Apache bundle installed on a Linux or Windows server and version of Perl (Bekman & MacEachern, 2009).

The Apache bundle is a highly configurable open-source internet server. Since its first release in April 1995, it has become the most popular server software of the World Wide Web. According to the survey conducted by Netcraft (2006), the vast majority (currently 62 percent) of Web servers in the world use Apache. The project is described as "a collaborative software development effort aimed at creating a robust, commercial-grade, featureful, and freely-available source code implementation of an HTTP (Web) server" (Apache.org, 2006). However, modern Apache distributions mostly depend on modules for providing dynamic content and functionality (Lerner, 1999). When Apache is installed on a new system, one can choose which modules to install. One such module is mod_perl.

What functionality does mod_perl module add to the Apache web server? An important feature mentioned in the computer literature is the implementation of the Perl programming language into the Apache server (Spainhour & Eckstein, 2002). This should make it easier for users to programme and configure the Apache server and integrate it with databases for dynamic content. Bekman & Cholet (2003) suggest that "with mod_perl, Apache is not only a web server; it is a complete programming platform" (p. 15). The module thus provides a programmable interface to the Apache server, opening it up for customisation and new features by using Perl language rather than C. Perl is the acronym for Practical Extraction and Report Language (Daintith & Wright, 2008). It is a high-level, interpreted programming language that is one of the main languages (besides Java) for internet scripting (ibid.). Scripting, in this context, means that a small programme scans text files in order to extract and manipulate information. As Labrinidis and Roussopoulos (2000) note, "web servers are increasingly being used to deliver dynamic content rather than static HTML pages" (p. 26). In order to generate web pages dynamically, servers somehow need to execute a script that connects to electronically stored information. For example, if a web developer wishes to create a template system for displaying dynamic content on an internet site, modules can be written in Perl. Mediated by Perl, dynamic content can then be extracted from a relational database and displayed on a webpage. However, other scripting languages can also provide similar functions.

6.5.1 The overall network structure and identification of sub groups

The aim of this section is twofold: initially to get an overview of the structure of interactions in the mod_perl project; second, applying formal methods to see how participants are located

and embedded in the overall structure. The latter seeks to identify central and more peripheral sub-groups of actors within the network to serve as contrasts for a more detailed content analysis of processes for maintaining intersubjectivity.

The population for social network analysis is somewhat restricted in terms of actors, but to include all members of the project as units of observation is too comprehensive for the timeframe of this thesis. Since the first version of mod_perl was released by Gisle Aas on March 25, 1996, thousands of individuals have contributed to its current state (http://perl.apache.org/about/history.html). In order to strike a good balance between a manageable and informative data set, I choose to include all members within a timeframe of six months. Another reason is that in a global open-source project of this magnitude, over time, new developers tend to join and others leave. By including all actors within a time interval, we get a 'snapshot' of participation in this period. The population studied is thus defined by these boundaries:

- Naturally occurring clusters of individual contributors in the mod_perl mailing list.
- All active participants in the project within a six month period are included to conduct a census; meaning that all elements within the defined population are included as units of observation (Hanneman & Riddle, 2005).

A brief summary of the population is presented in the table below:

Timeframe	01.01.2008 – 30.05.2008
(n) contributors	215
Number of postings	1154
Mean	6,2

Specifying the relations among actors

The set of actors studied in the mod_perl project might be connected by multiple ties and relations of different types. For instance, participating developers may also be personal friends or perhaps co-workers in a different context. They conceivably have negative or positive impressions of each other and so on. For the initial descriptive purpose of this study, I have selected one specific type of relation as the basic units for constructing the network; namely communicational ties among contributors on the mod_perl developers' mailing list. In accordance with the boundaries stated above, all of the communicational ties among the selected nodes are studied to provide a census for a full network analysis (Hanneman & Riddle, 2005). It follows that the adjacency matrix is to include information about each participant's communicational ties with all other participants in the population. The rationale is to collect information about all communicational dyads to construct an inclusive overview of the network and provide parameters to calculate centrality and density in communication. In this way, sub-groups of core and peripheral contributors can be identified for further content analysis. Such full network data are also recommended in the literature in order to take advantage of key structural concepts of network analysis (Hanneman & Riddle, 2005; Wasserman & Faust, 1994).

Hence, what should be counted as a valid communicational tie among participants? Due to the focus on interacting participants, I will suggest to only count mutual communication as ties. Related network studies show that this can be purposely conducted by dichotomising post and reply messages in online communication (see e.g. Willging, 2005). Dyads of participants can be measured by a binary scale where the presence of communicative interaction in the form of post and reply sequences is coded as '1' whereas non-interacting dyads are coded '0'. Consequently, messages that are not replied to are unaccounted for in the analysis. This leaves us with a standard asymmetric binary network matrix to perform basic operations. The matrix thus consists of cells assigned with either '0' or '1' (see Table 10). However, one can argue that potentially valuable information is lost by not using a higher level of measurement (e.g., ordinal or interval level). For the descriptive and sorting purposes of providing contrast groups for content analysis, I do not regard the strength of ties as significant. Actually, the simplicity of handling binary data may be worth the information lost. One may also consider that many of the algorithms for measuring properties of actors and networks have been developed for binary data. Binary data are so widely used in network analysis that it is not

unusual to see data that are measured at a 'higher' level transformed into binary scores before analysis proceeds (Hanneman & Riddle, 2005:11).

To sum up, interactional relations among pairs of actors (or dyads in network terminology) are to be measured following a binary scale. In the adjacency matrix, contributions in the form of reply messages to other developers are assigned as present by the value '1' and '0' for non-present. An extract of the matrix[8] is represented as follows:

	dev-1	dev-2	dev-3	dev-4	dev-5	dev-6	dev-7	dev-8	dev-9	dev-10	dev-11	dev-12
dev-1	0	0	0	0	1	0	0	0	0	0	0	0
dev-2	0	0	1	0	0	0	0	0	0	0	0	0
dev-3	0	1	0	0	0	0	0	0	1	0	0	1
dev-4	0	0	0	0	1	0	0	0	0	0	0	0
dev-5	0	0	0	1	0	0	0	0	0	0	0	0
dev-6	0	0	0	1	0	0	0	0	0	0	0	0
dev-7	0	0	0	0	0	0	0	0	0	0	0	0
dev-8	0	0	0	0	0	0	1	0	0	0	0	0
dev-9	0	0	1	0	0	0	0	1	0	0	0	1
dev-10	0	0	0	0	0	0	0	0	0	0	0	0
dev-11	0	0	0	0	0	0	0	0	0	0	0	1
dev-12	0	0	1	0	0	0	0	0	1	0	1	0

Table 10: Extract from adjacency matrix showing network data

The elements (cells) display the relations among participants. In accordance with graph convention, the direction of communication is indicated by placing the senders' ties (or messages) in the rows and the targets in the columns (Hanneman & Riddle, 2005). The rows thus represent the source of messages on the mod_perl e-mail list while the columns represent the targets.

Density, centrality and *cliques* are regarded by theorists as important conceptual ideas derived from SNA (Garton, et al., 1999; Wasserman & Faust, 1994; Willging, 2005). In the context of the mod_perl project, some participants may appear as central and others more isolated or peripheral. These ideas are explored further in the next paragraphs.

Density and Centrality measures

According to Scott (2000) in the *Handbook for Social Network Analysis*, "concepts of density and centralization refer to differing aspects of the overall 'compactness' of the graph" (p. 89). As the former expresses cohesion and the latter the extent to which the cohesion is organised

[8] The complete adjacency matrix in Ucinet 6 format is available at http://dl.dropbox.com/u/674266/sna/ucinet_matrix.zip

around particular focal points, these two measures can be complementary in analysis. Of principal importance for the research questions posed in this thesis, centrality scores also allow the least central participants to be identified. Nodes in the network with low centrality can be regarded as the peripheral participants in the mod_perl network. Furthermore, visual inspections of the graph and formal calculations are undertaken in the next subchapter in order to look for clusters or cliques around the centre and peripheral regions of the network.

First, let us consider the overall cohesion in order to obtain a general impression of the communication patterns in the mod_perl project. How tightly connected are the participants? By examining the patterns of relation among the participants, the overall structure may appear as more or less dense. Density is a widely used measure of a social network's structure, and is defined by "the number of actually occurring relations or ties as the proportion of the number of theoretically possible relations of ties" (Garton, et al., 1999:84). In a theoretically complete graph, each node is connected to all other nodes. When measuring density, we get indicators on how much the graph differs from completion, describing the general level of cohesion in the graph (ibid.). Thus, densely knit networks involve considerable direct communication among most participants in the network. The descriptive measures follow below. Calculations are supported by the computer programme Ucinet[9] for Windows. The density of the whole network is simply calculated by summarising the number of communicative connections between members in the adjacency matrix and dividing it by the number of possible connections. The overall density score together with the total number of ties in the mod_perl dataset is shown in the following table.

	Density	No. of Ties
dataset	0.0127	584

Table 11: Overall density score mod_perl project

[9] A specialised programme for handling and assisting analysis of social network data developed by Borgatti, Everett and Freeman (2002). The software can be obtained from Analytic Technologies http://www.analytictech.com/ucinet6/ucinet.htm

The density score ranges from 0-1, describing the general level of linkage among the actors in the network. A score of 0.0127 indicates a rather low density. The points in the graph are far from completion, implying a rather loosely knit structure. Another implication is that we can expect relatively few participants in the mod_perl project to communicate directly and frequently with each other. The relative number of communicational ties among actors is only a small fraction of all possible ties. However, network density provides only a rough impression as it describes the general level of cohesion in the overall network. To explore if this cohesion is organised around particular focal points, centrality measures are more informative.

Centrality measures of the relational data in the matrix are probed in the following paragraphs to spot key participants in the network. The idea of point centrality originates in the sociometric concept of 'the star'; highly connected or popular individuals form star structures in the network (Scott, 2000). It thus concerns the relative centrality of points in the graph. The most basic method of measuring point centrality is to estimate the degree of various points. In the present dataset, this is done by summarising the number of other participants each participant is adjacent to. A point in the graph is regarded as central if it has a high degree and, to use Scott's words, "the corresponding agent is central in the sense of being 'well-connected' or in 'the thick of things'" (Scott, 2000:83). An important distinction is made in the literature on SNA between 'local' and 'global' centrality (ibid.). Points that are locally central have extensive connections in the nearby environment, whereas globally central ones occupy regions that are strategically positioned in the overall structure of the network. Here, the distance between various points is also accounted for. Globally central points are relatively close to many others. A useful calculation in this regard is *degree centrality*, developed by Freeman (1979), which shows the overall network activity of individuals. Borgatti (2005) describes it as "the number of ties incident upon a node" or "the sum of each row in the adjacency matrix representing the network" (p. 62). Participants with the highest degree centrality score are listed in Table 12. See appendix 9.5 for respective scores for all nodes in the dataset and appendix 9.6 for the whole network sociogram.

		OutDegree	InDegree	NrmOutDeg	NrmInDeg
12	dev-12	62.000	40.000	28.972	18.692
29	dev-29	29.000	19.000	13.551	8.879

1	dev-1	25.000	21.000	11.682	9.813
81	dev-81	24.000	23.000	11.215	10.748
23	dev-23	22.000	12.000	10.280	5.607
16	dev-16	18.000	12.000	8.411	5.607
3	dev-3	17.000	16.000	7.944	7.477
14	dev-14	16.000	11.000	7.477	5.140
28	dev-28	13.000	14.000	6.075	6.542
52	dev-52	11.000	4.000	5.140	1.869
40	dev-40	11.000	12.000	5.140	5.607

Table 12: Freeman's (1979) degree centrality measures indicating most active participants in the mod_perl community

The above measures point towards dev-12 as the most active and central participant. As this is a directed graph, it is also relevant to assess OutDegree and InDegree. An Out/InDegree of 62/40 means that the participant sent 62 communicational ties and received 40. A high centralisation score would indicate that this participant has a greater number of connections than others. In an asymmetric expert network such as the mod_perl community, a high InDegree may indicate centrality in the sense that they are consulted more often by other members. High OutDegree centralisation could mean that the participant is influential in doing the majority of consulting in the project.

To obtain a visual impression of the connectedness of dev-12, I extracted his ego network from the overall network. Note that the sizes of nodes are weighted according to degree centrality.

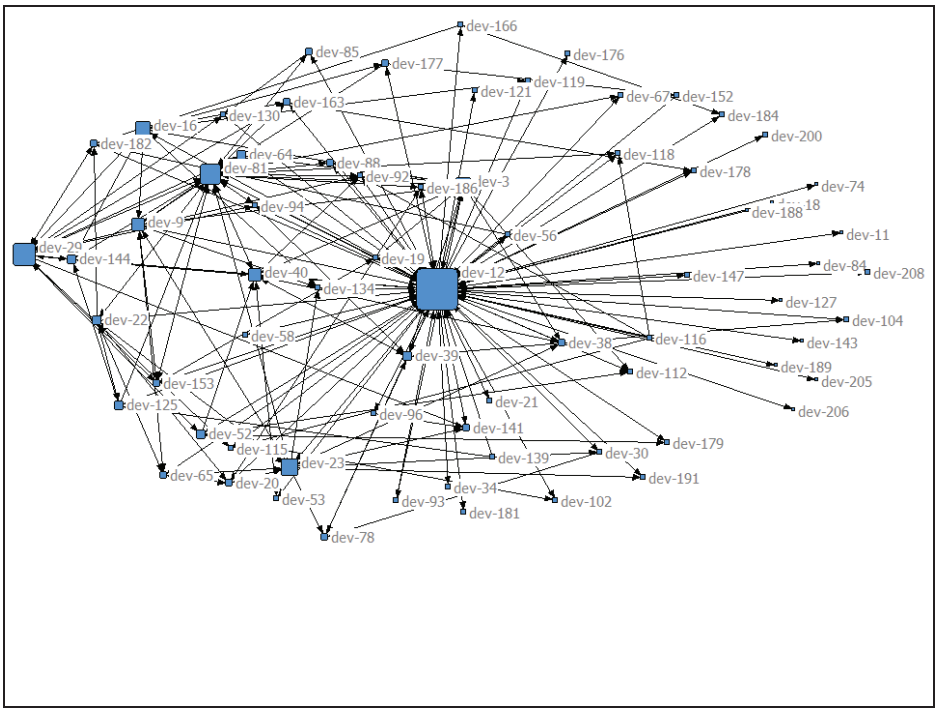


Figure 6: Sociogram ego network for dev-12

The sociogram of dev-12 forms a rather clear star structure, indicating a highly central participant. It also emerges that this central participant is connected to other central ones (i.e., dev-29 and dev-81). However, these measures with corresponding visual representations only signify centrality around the most central points in the network; this does not, however, tell us whether the central points cluster in sub-groups (cliques) around the structural centre. For this we need to add a few 'group conceptualisations' to the analysis. The identification of possible clusters or cliques in the central and peripheral regions of the network is probed further in the next chapter.

In the more peripheral regions of the network, there are participants with a low centrality score. In the period of observation, they contributed once or twice, forming dyadic or triadic relations. An example is dev-209 (Figure 7 below). The participant posted one question and received two replies, giving an OutDegree 1 and InDegree 2. The ego network of dev-209 thus forms a triadic relation:

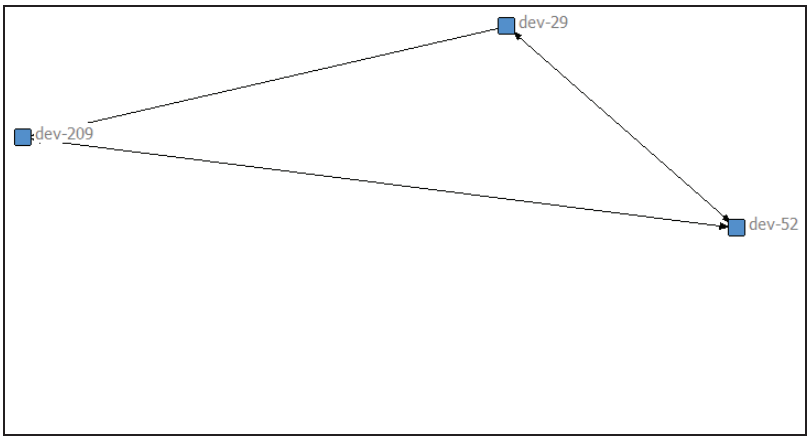


Figure 7: sociogram ego network for dev-209

The degree centrality measures conducted above give us only 'thin' initial descriptions of communication patterns in the mod_perl network. This approach might be criticised as it simply accounts for the actors' immediate, local ties rather than incorporating indirect ties to other participants in the network (Hanneman & Riddle, 2005). For example, a single mod_perl developer can be connected to many peers who are rather peripheral in the network as a whole. According to Freeman's degree centrality measures, the participant would appear to be quite central. However, the participant is only locally central. In the next section, I continue exploring whether the network tends to form sub-groups or clustering nodes around a structural centre and in the more peripheral regions.

Identifying clusters/cliques in the structural centre and peripheral regions

The analytic potential associated with identifying sub-structures, groups or cliques within the network is a widely used and powerful feature of SNA. In the context of mod_perl, this can be a useful tool for understanding the social structure of the project and paving the way for content analysis. For the latter purpose, I intend to scrutinise the network and single out contrast groups along the central and peripheral regions. An underlying assumption is that individual participants' relative placement or integration in the overall structure impacts processes of intersubjectivity. Considering the population size (n=215) of the network, it might be difficult to know and maintain communicational ties with all the other developers. It

was also indicated earlier that the overall density is at the lower end of the scale. As Hanneman (2005) puts it:

> Size is critical for the structure of social relations because of the limited resources and capacities that each actor has for building and maintaining ties. As a group gets bigger, the proportion of all of the ties that could (logically) be present -- density -- will fall, and the more likely it is that differentiated and partitioned groups will emerge (p. 41).

Thus, a potential consequence of a loosely knit large expert network may be the emergence of smaller sub-structures within the overall network. Will connections among participants within the network compound, extend and develop larger cliques or sub-groups? If so, can we distinguish clusters in the core and peripheral regions in the network? In the following paragraphs, I explore the presence of sub-groups or cliques within the network to provide contrasts for further content analyses targeting processes of intersubjectivity.

Hanneman (2005) suggests a clear-cut, general definition of a clique as "a sub-set of actors who are more closely tied to each other than they are to actors who are not part of the group" (p. 77). However, to perform meaningful operations on data, more accuracy is required. Initially, I explored algorithms operating on a strict definition of cliques. This meant that I looked for fully connected subgraphs in the network where *n* actors have all possible ties present among them. Furthermore, the adjacency matrix is symmetrised as the clique algorithm applied operates on binary symmetric data. Thus, the direction of links among nodes is ignored and only reciprocal ties are accounted for in order to explore strong ties among central sub sets of participants. For the smallest group to be considered a clique, I applied n=5. The following four cliques were found:

1: dev-12 dev-29 dev-40 dev-81 dev-92
2: dev-12 dev-29 dev-40 dev-81 dev-144
3: dev-12 dev-20 dev-22 dev-23 dev-65
4: dev-3 dev-12 dev-38 dev-39 dev-40

There are thus four maximally complete subgraphs present in the network. The overlapping relations in the four sets of cliques are displayed in figure 9;

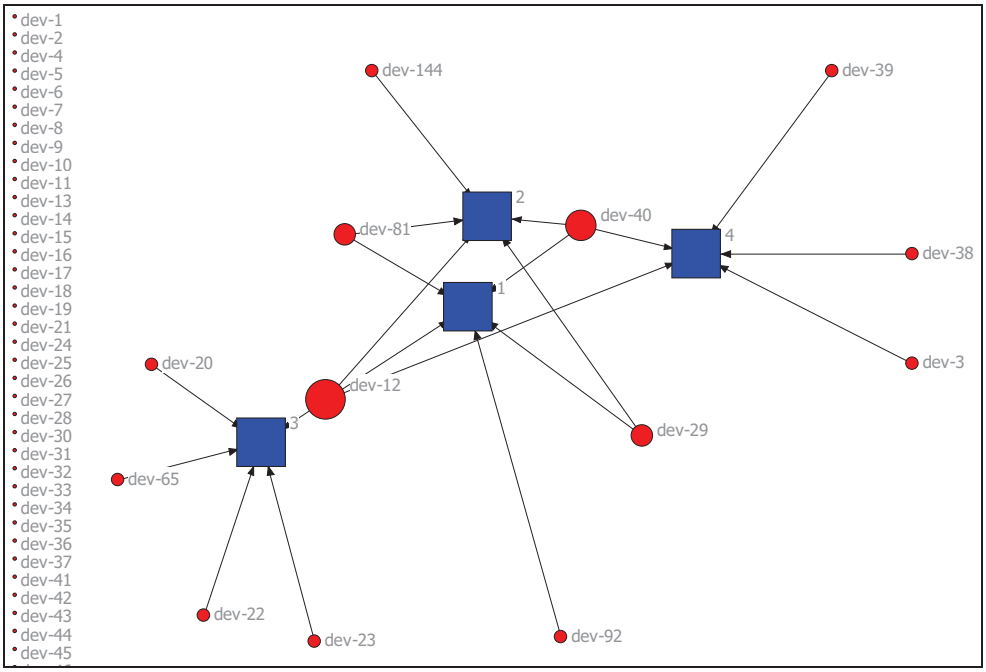


Figure 8: Clique sets showing central participants' co-membership

From the clique sets above, we get an impression of central participants' co-membership in these sub-groups. For instance, actors dev-12 and dev-40 share membership in three of the four groups. On the other hand, dev-3, dev-20, dev-22, dev-23, dev-38, dev-39, dev-65, dev-92 and dev-144 are more isolated, since they belong in only one of the cliques.

Recalling the last chapter, we see that dev-12, dev-29, dev-40 and dev-81 also have the highest scores with respect to centrality measures. From the model above, we see that the most central participants are overlapping co-members in the cliques. It follows that these actors form a central sub-group around the structural core in the network. Given that the most central nodes in the network are co-members in clearly defined cliques, I find it reasonable to regard them as the structural centre in the network. A closer inspection of the whole network sociogram (appendix 9.6) also reveals the four central participants positioned around its core. The well-bounded, central position of this sub-group is corroborated by a less strict clique concept called *K*-plex (Seidman & Foster, 1978). K-plex is an algorithm in which each

member of a K-plex group of a specific size N has N-K ties to other members. In the dataset, a total of four 1-plexes were found (N=5):

1: dev-3 dev-12 dev-38 dev-39 dev-40
2: dev-12 dev-20 dev-22 dev-23 dev-65
3: dev-12 dev-29 dev-40 dev-81 dev-92
4: dev-12 dev-29 dev-40 dev-81 dev-144

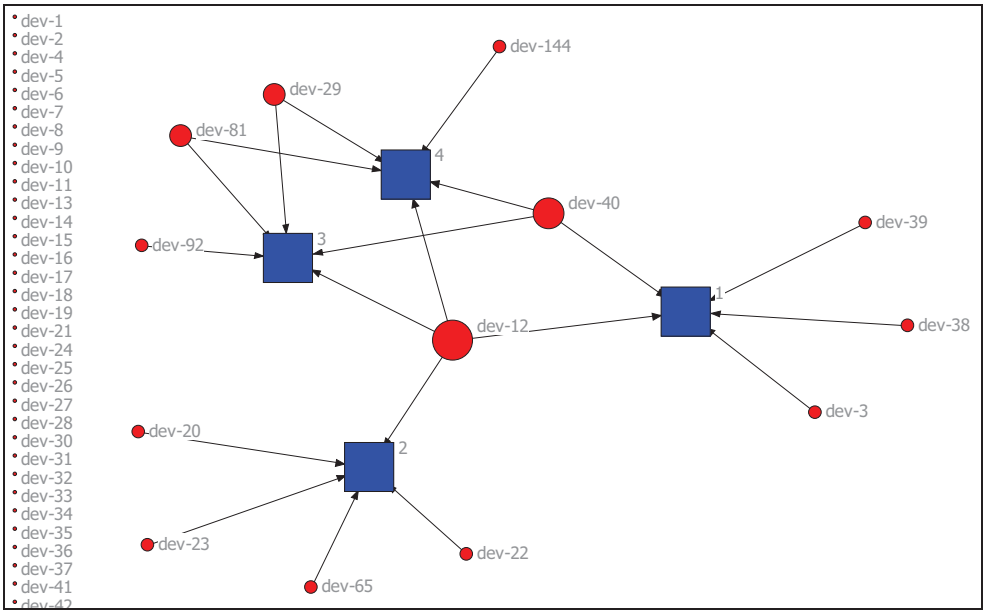


Figure 9: Clique sets showing central participants' co-membership by applying the K-plex algorithm

In these groups, we can also spot overlapping representation of the core participants; dev-12, dev-29, dev-40 and dev-81.

Now, the participants' positioning in areas of the overall network with high or low cohesion can be identified by a criterion of minimum degree as suggested by Seidman (1983). K-cores describe a maximal group of actors who are connected to *k* other members in the group. The criterion is that each point in the maximal subgraph is adjacent to at least *k* other points. For instance, to identify a 3*k*-core, all points with a degree of 2 or less are deleted (Scott, 2000). K-cores thus identify areas of the graph with clique-like structures. The K-core values

indicate that the sub-group identified above containing the core actors is present in the 5k cluster; the relatively dense area in the middle of the circle representing high cohesion:

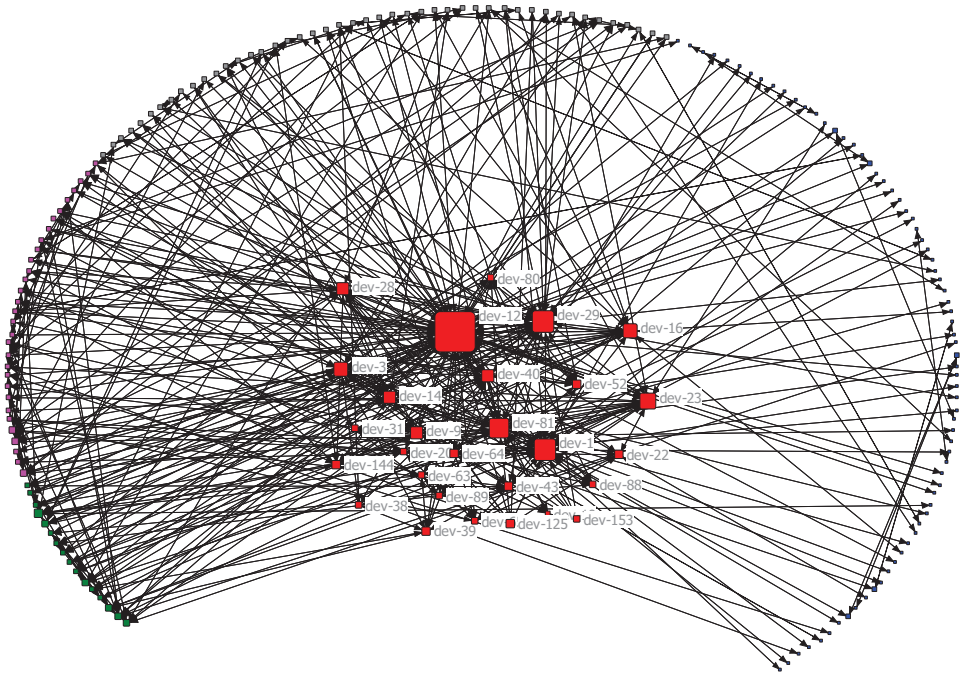


Figure 10: Areas containing clusters of high and low cohesion (K-cores)

The model is arranged so that regions with low cohesion values are displayed at the lower right (1k), whereas the more cohesive clusters are to the lower left (4k) and highest in the middle (5k). A closer reading of the measures and the sociogram displayed above also reveals dev-1 to be a rather central participant. However, it appears that dev-1 is not a member of the cliques forming around the core of the network. In other words, his communicative relations are not tightly bound to the central core of developers. It is possible that he has a specific brokering role in the forum, i.e., guiding newcomers or providing some kind of support.

In summary, so far I have identified a sub-group of 4 central participants operating in the structural centre of the network, namely dev-12, dev-29, dev-40 and dev-81. These will be referred to as *the nucleus group* in the content analysis that follows. The next problem I face is to identify a cluster of peripheral participants to serve as a contrast to the nucleus. A distinguishing feature emerging from the overall network structure is the relatively high

number of peripheral participants in the population. A simple dichotomised core-peripheral model based on density measures indicates that 202 of 215 participants are members belonging to the peripheral class. What is more, the low density scores point towards a different communication pattern in the peripheral region of the network:

	1	2
1	0.436	0.070
2	0.053	0.005

Table 13: Density matrix showing scores in core regions (1) and peripheral regions (2) of network

From the matrix, we see that intergroup communication among peripheral participants is scored 0.005. This means that the numbers of communicational ties in between peripheral actors are very few opposed to a theoretically complete graph (Garton, et al., 1999:84). On the other hand, among the core participants, the density measure of 0.436 points towards a considerable amount of direct communication among most participants in this region of the network. Almost half of the possible ties are present.

In order to identify a cluster of participants in peripheral regions of the network, I removed the core participants from the adjacency matrix and searched for clusters forming among the remaining participants. Subsequently, clique measures are applied to the matrix consisting of peripheral participants in order to identify sub-groups. Freeman's degree centrality measures (appendix 9.5) and a categorical core-periphery classification (appendix 9.7) were applied for corroboration of the participants' positions in core and peripheral regions. Operating on a strict clique definition, no cohesive sub-groups consisting of more than three members were found in the peripheral regions of the network. This indicates that there are no clearly defined sub-groups of participants forming in the peripheral region of the network. A closer inspection of the overall network sociogram corroborates these findings (appendix 9.6). In the peripheral regions, we see that communicational ties mainly appear as rather loose dyads and triads among actors with low centrality scores. According to Borgatti and Everett (1999), social networks with a core-and-periphery structure involve a cohesive core of participants and a more sparsely knit periphery. I have defined the cohesive core of the mod_perl network

as the tightly interconnected nucleus group. However, due to the high number of peripheral participants in the project, some kind of data reduction is needed prior to the content analyses in the next chapter. My practical solution to this problem was to randomly extract a selection from the total population of peripheral participants. As a starting point, I used the simple core-periphery model (appendix 9.7) and randomly selected 10 participants from among the 202 who belong to the class of peripheral participants. The core-periphery categorisation uses a generic algorithm developed by Borgatti and Everett (ibid.) to fit a core-periphery model to the network data in order to identify which participants belong in the core and which belong in the periphery. The Ucinet software was applied to categorise the participants according to the core-periphery model (Borgatti, et al., 2002). In the upcoming content analysis, the 10 random participants in the peripheral region are referred to as *the peripheral group* representing the participants with low centrality and density measures. In comparison with the nucleus group, they contribute and interact infrequently on the project's distribution list. To sum up, the contrasting groups and distinguishing features are displayed in the table below:

	Participants	Characteristics
Nucleus group n=4	dev-12, dev-29, dev-40, dev-81	High centrality and density. Densely knit around the structural centre of the graph. Frequent contributors on the project distribution list.
Peripheral group N=10	dev-19, dev-30, dev-33, dev-95, dev-131, dev-142, dev-153, dev-169, dev-186, dev-207	Random participants with low centrality and density measures. Sparsely knit in the peripheral regions of the network. Infrequent contributors on the project distribution list.

Table 14: Contrasting features of nucleus and peripheral group

From the social network analysis conducted in this chapter, we get an impression of how the overall mod_perl network is structured as a set of relations during the period of observation (01.01.2008 – 30.05.2008). The type of relation defined as the basic unit for constructing the

network is reciprocal communicational ties among contributors on the project's mailing list. The overall network's density score (0.0127) signifies a rather loosely knit structure. This means that the numbers of communicational ties in the network are far from completion; only a small part of all possible ties are present. The first part of the analysis concludes that the general level of cohesion in the overall social network is low. However, this is as expected in a network with a large number of participants (Scott, 2000). The next step in the social network analysis applied centrality measures to explore whether the cohesion is organised around 'focal points' or highly central participants. In accordance with the second aim stated in the introduction of this thesis, formal measures were applied to identify central and peripheral sub-groups of participants. The rationale was to provide contrast groups for a more detailed content analysis targeting processes of intersubjectivity. This brings us to the third aim of this thesis; namely, the investigation of how processes of intersubjectivity evolve around shared objects of development in the mediated communication among actors in the identified sub-groups.

6.6 Analysing intersubjective interactions

In the current chapter, the two sub-groups of participants identified as *nucleus group* and *peripheral group* are investigated further in order to follow up on the final aim stated in the introductory part. As the thesis has progressed, a more precise research question was formulated that targets the role of prolepsis around shared objects of development in the mediated communication among participants in the sub-groups (p. 112). Additionally, the underlying theoretically driven assumption is that proleptic instances trigger the construction of new understandings among the participants and that the associated sense-making of incomplete utterances varies according to actors' relative placement in the core and peripheral regions of the overall network structure. This is where the social network approach and the 'crossbred' virtual ethnography described above are combined to inform the research question. The formal methods of social network theory serve as tools for rendering the project's communicational structure visible and reducing the e-mail archive's content into manageable sub-groups for a more detailed content analysis. Consequently, we also get a purposeful data reduction as the content analysis is narrowed down to messages between defined clusters of participants forming around the structural centre and peripheral regions of the network.

When preparing the binary dataset for network analysis, only mutual communication among participants was counted as valid communicational ties. From this scale of measurement, the overall structure of the computer-mediated communication network was rendered and regions with high and low cohesion were identified. However, to understand processes of intersubjectivity in these regions, we need to trace and analyse *the content of messages* flowing between the nodes of the network. Each node represents one individual participant and the communicational ties in between represent message content among participants (see overall network structure in Figure 11). In order to understand how intersubjectivity develops around shared objects of development in the mod_perl project, it is thus reasonable to analyse the content of project-related messages occurring on the mailing list. Now, departing from the sub-groups identified following a social network approach, I have focused on analysing the content of messages from participants belonging to the *nucleus group* and the *peripheral group*. This means that I have coded and retrieved post-reply sequences from each cluster within the period of observation (01.01.2008 – 30.05.2008). To probe the research question, the exchanges of utterances among these participants are analysed for processes of intersubjectivity with special attention to one aspect; namely, the role of prolepsis around shared objects of development. This implies that we search for interactional 'hotspots' in the project's message archive that involve sense-making of incomplete utterances and shared attention towards some feature of the mod_perl code or functionality. In order to achieve a balanced view of the targeted processes, I will also investigate data, for example, where the associated sense-making breaks down, is not picked up in subsequent postings or somehow fails to be addressed.

When taking into account the message content flowing between the different regions of the network, what characterises the two sub-groups? Do they represent different functions or roles in the project? Prior to analysing processes of intersubjectivity on an 'utterance level', it can be useful to summarise the group's main characteristics and clarify the differences in message content. The central and peripheral regions of the overall social network are rendered as follows:

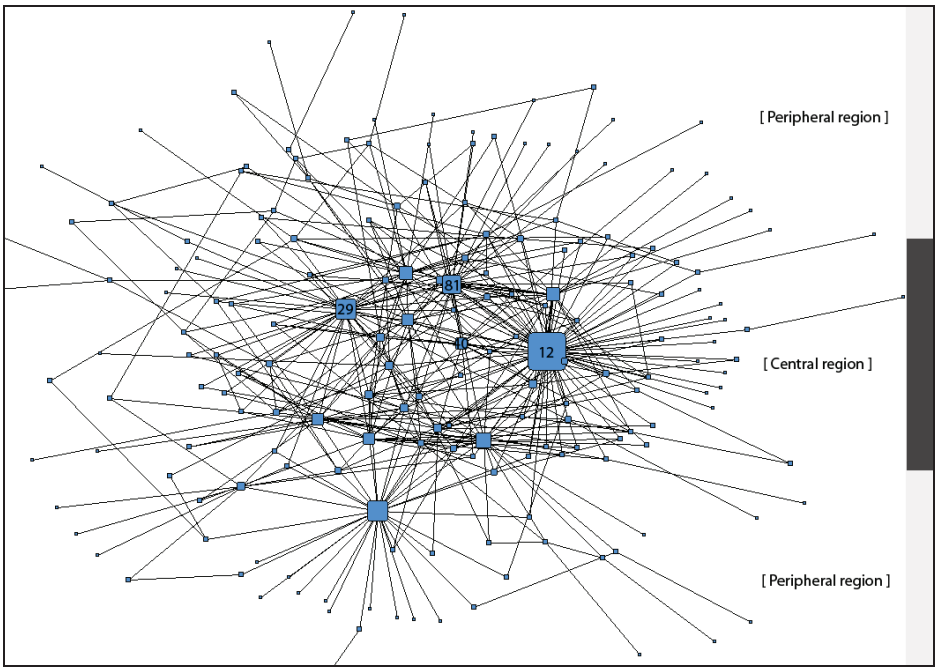


Figure 11: Central and peripheral regions of social network

In the whole network structure above, the region with a high degree centrality is indicated with a darker colour (right column). This is the area where the cluster of participants defined as the nucleus group is represented. From the previous social network analyses, we have seen that dev-12, dev-29, dev-40 and dev-81 represent central nodes around the core of the network. They have the highest scores on centrality measures and are also found to be co-members in overlapping clique-sets forming in the structural core. If we look at the general level of linkage between the points in the graph, representing participants in the nucleus group, all points are directly connected to every other point:

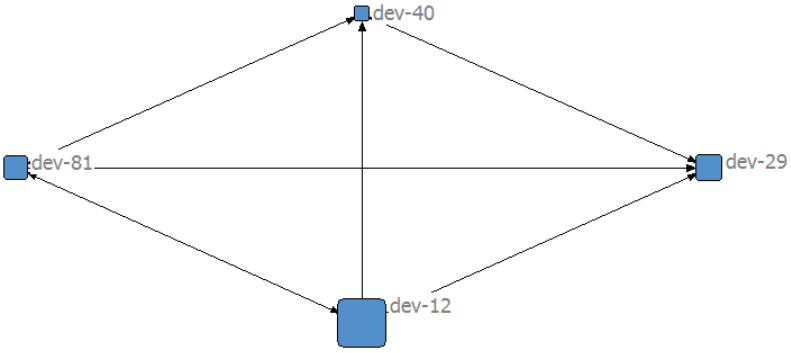


Figure 12: Intra-group sociogram nucleus group

According to Scott (2000), such completion is relatively rare, even in small networks. However, the above sociogram only shows a small part of the whole network structure (see Figure 11). It was indicated earlier that the overall network has a rather loosely knit structure. It follows that within the overall network, the nucleus group is atypical in the sense that the subgraph among the actors is complete. This means that all possible communicational ties are present, indicating a high level of cohesion among central participants. During the period of observation, multiple messages flow between these participants and they are also frequently consulted by other participants across the social network. Moreover, the most central participant, dev-12, is commonly referred to by other participants as *the expert*. For example, when taking part in an exchange with a novice participant, dev-81 utters: “you should probably pay more attention to what dev-12 tells you, rather than to what I tell you below. He is the real expert, I just dabble” (17 Mar 2008, 13:13). On the other hand, as one moves out of the structural core to the peripheral regions of the network in Figure 11, senders of messages are more likely refer to themselves as *beginners, newbies* or *novices*. For example, one peripheral participant posts a message with the following introduction:

[Line]	**Message from dev-153, Tue, 29 Jan 2008 10:51 GMT**
1.	Hello all Apache mod_perl2 module experts (I am a newbie with Apache).
2.	
3.	Hope I am clear in my explanations (my English is not so good and I had a
4.	lot of problems explaining my needs by mail. I am not sure that everybody
5.	will read entirely this mail ;-)

mod_perl forum extract 1: Introductory message from peripheral participant

This peripheral participant inquires of the "mod_perl2 module experts" and points out that he is a novice with the technology (line 1). Moreover, he claims to have experienced problems with formulating his needs in an e-mail message. As a "newbie" with limited language skills, dev-153 seems to be inhibited by what Clark and Brennan (1991) refer to as *production costs* associated with the text-mediated communication. The activity of producing interpretable utterances requires a great deal of effort. Another peripheral participant includes the quote "Beginner question" in the message heading and starts the message body with: "I'm just trying to understand the basics, having set up my server to run mod_perl" (dev-142, 17 May 2008 08:04). In the overall network structure, this participant appears in a dyadic relation with one other peripheral participant in the period of observation:

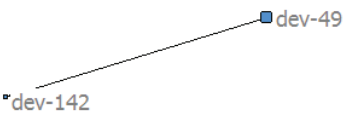


Figure 13: Dyadic relation dev-142 and dev-49

When inspecting the central and peripheral regions of the whole network structure showed in Figure 11, we see that the outermost borders contain mostly interacting dyads of peripheral

participants. The post-reply sequences at the edges of the network predominantly involve one or two messages among peripheral participants. The message structure in this region is thus different from the threads appearing in the central region. The latter often spans several days, involving multiple central participants. Moreover, the message content sent among participants belonging to the peripheral and central regions of the network reveals a characteristic communication pattern. Participants belonging to the former cluster tend to formulate initial problems based on their experience from local practices, whereas participants in the latter category are more likely to post advisory replies and contribute to the distributed sense-making (exemplified in extracts 2-4 below). It follows that some of the 'introductory problem definition posts' sent from the peripheral participants during the period of observation are picked up and submitted to active meaning making among central participants. What about the intersubjective content of these messages? Do peripheral participants communicate deficiencies or missing functionality in the joint knowledge object that 'triggers' the experts to participate in the active meaning making? What about processes of shared prolepsis?

As indicated above, participants in the peripheral regions are more or less *novice users* of the software module that is developed in the project. This means that they are unlikely to contribute directly to the joint development of code, but rather provide bug reports and test new releases. From a 'grounding perspective', in keeping with Clark and Brennan (1991), one can argue that the limited participation in the peripheral regions provides less shared information or common ground among these participants. For example, in the period of observation, dev-49 participated in a total of three message exchanges and dev-142 in only one. On the other hand, in the central regions of the network, interactions are more frequent and perhaps more content is added to these participants' common ground. The most central participant (dev-12) posted 62 messages and received 40 during the period of observation. The dense interaction patterns in the central region also indicate that these participants are more likely to participate in extended sense-making (threads spanning several days involving multiple participants) whereas 'dyadic participation' in the peripheral regions is commonly short-lived. The latter may occur when a peripheral actor introduces a problem that has previously been addressed and solved, and the reply message includes a link to the relevant thread in the e-mail archive. For example, dev-169 writes; "I don't know where to find answers to the following question, so I better ask this list" (20. Feb 2008 18:08). The reply message from a central participant includes the following utterances: "I believe the following

message from this month's archive is still applicable [link] (dev-3, 20. Feb 2008 12:35). However, sometimes peripheral participants 'hit the right spot' in identifying an issue that triggers a sense-making discussion among competent peers around the structural core of the network. Drawing on the theoretical propositions deduced from Knorr Cetina's theory (see chapter 2.3.1), we can say that a deficient or incomplete feature in the distributed project's joint knowledge object 'triggers' the experts' participation. Such instances might be significant constituents for understanding the partially shared knowledge content among participants in the project. Are peripheral participants invited into the active sense-making around the structural core of the network? What characterises instances of shared prolepsis between these different regions of the social network? Let us now look more closely at the message content among participants identified as the nucleus group and peripheral participants.

6.6.1 Processes of intersubjectivity and shared prolepsis between the structural core and peripheral regions of the network

In the following paragraphs, I present and discuss the content of selected message exchanges between nucleus group participants and peripheral participants that involve active sense-making concerning some aspect of their joint knowledge object. The first extract below shows a sequence of post and replies spanning five days. The initial problem formulation triggering the 'sense-making' discussion involving two members of the nucleus group (dev-12 and 81) was introduced in a message issued by the peripheral participant, dev-186:

[Line]	**Message from dev-186, 13 Feb 2008 14:09 GMT**
1.	Hello @all,
2.	
3.	we are just transfering our Intranet Server which used Apache 1.3 and
4.	mod_perl1 from SLES8 to SLES10 wiht Apache2 and mod_perl2.
5.	
6.	The switch has to go quick and we can't redesign all of our old mp1
7.	programs to mp2, so wetry to use the compat module. But I have no luck

8. with it. I tested my config with SELS10, Opensuse 10.3 and Ubuntu but i hit
9. everytime the same problem. I searched the Mail Archvie and the
10. Documentation, but no luck.
11.
12. [code]
13.
14. I get the following output..
15.
16. Software error:
17. Can't locate object method "request" via package "Apache" at /usr/lib/cgi-
18. bin/test1.pl line 4.
19. For help, please send mail to the webmaster (webmaster@localhost), giving
20. this error message and the time and date of the error.
21.
22. It seems, he can't find the requested methods.. How can i validate that the
23. compat module is loaded and working?

mod_perl forum extract 2: Initial problem formulation from peripheral participant

When switching to an upgraded intranet server, dev-186 experiences problems with his earlier mod_perl programmes (mp1). The previous server configuration has been tested on different systems by using a compatibility module, but with no luck. He also searched the project's message archive and documentation without finding an adequate solution to the problem. In order to assist mod_perl peers in understanding this problem, a piece of code is posted along with the message (line 12) and the software error that appears when running the system (line 16-20). Following Rommetveit's (1979a) framework for intersubjectivity, the sender of utterances is in a privileged position to control what features of the shared object "to enter the field of shared attention" (p. 95). If we interpret the mod_perl programme code as the joint knowledge object attended to in the forum, the message sent by dev-186 points out a specific feature of the object to penetrate the shared space; namely, the "compat module" (lines 7 and 23). According to the documentation, this is a module that provides a compatibility layer when transferring server programmes written for mod_perl 1.0 to an upgraded mod_perl 2.0 system (http://perl.apache.org/docs). In the problem formulation, he also refers to several

different server operating systems (SELS10, OpenSuse 10.3 and Ubuntu) that are not integrated parts of the mod_perl development project. However, a partial shared understanding of how these systems integrate may be important premises for rendering the current message meaningful. What shared information can be tacitly assumed or presupposed by the receivers of the introductory message? Dev-186 explains that prior to sending this message, he searched the mail archive and documentation of the project (line 9-10). The previous postings stored in the mail archive and the project's documentation can be seen as knowledge resources that are mutually accessible and part of participants' partially shared social realities. The contributors involved in the sense-making that follows in the subsequent postings can then assume that dev-186 brings some level of shared understanding about the joint object of knowledge to the situation. Now, will there be a reciprocal tuning of perspectives on the specific aspect of the knowledge object introduced to the shared space?

As the initial message is addressed to all subscribers of the mailing list (Hello@all, line 1), the potential 'listeners' or receivers of utterances are numerous. They are dispersed all over the developer network, ranging from novices to experts. It follows that 'the listener' to be invited into a partial shared intersubjective space is not a single individual, as in Rommetveit's framework, but rather a *generalised other participant* (Mead, 1962). This implies that the sender somehow has to adapt the message content to a potentially large number of receivers. However, once the request is picked up by another participant, the situation changes. In headings of reply messages, it is more common to address the sender of the original message, for instance by introducing it with "Hi [name of participant]". A member of the nucleus group (dev-81) replies to the problem introduced by dev-186 a few hours later on the same day. It appears that he has experienced "the same kind of problem" (line 10-16) at an earlier point in time:

[Line]	**Reply from dev-81, 13 Feb 2008 21:49 GMT**
1.	Hi.
2.	
3.	About your problem below...
4.	
5.	I am not sure that this is going to help, nor even if it is really relevant to your

specific problem. Just trying to give you ideas, because it reminds me of something.

Below are two configurations, of two of our systems which have slightly different versions of mod_perl and the rest. I went through the same kind of problems you're going through, a while ago when going from Apache 1.x to Apache 2.x, and then again more recently when there were some changes in the mod_perl naming of modules between mod_perl 1.9xx and mod_perl 2.0, and then some additional things happened during a Linux Debian upgrade from Sarge to Etch, which changed the location where some Apache2 stuff was installed.

And there is still some difference to this day in some Perl scripts or modules between these two systems, which greatly bothers me, but it basically works.

The main difference is in the two first "use" statements in our "startup.pl" script (which I assume you are familiar with, as a way to pre-load some perl stuff when the server starts).

I do not remember precisely which problems we had, but they were of the same "general gist" as yours, so maybe this helps. [code]

mod_perl forum extract 3: Reply from member of the nucleus group (dev-81)

In the reply message above, dev-81 indicates a shared understanding of dev-186's specific problem that is incomplete and the suggestions for further problem solving might not even be relevant. Nevertheless, based on his interpretation of the problem formulation, dev-81 temporarily de-centres and discusses features of the shared knowledge object in the light of 'the others'' challenging position. Furthermore, dev-81 shares related system configurations and points out the main differences among them (line 21-23). Note, in the latter utterances, an explicit assumption about a piece of shared knowledge is induced. The presupposed information concerns a script called "startup.pl", which is applied to pre-load Perl code during

server startup. Dev-81 thus indicates that he found differences in their configurations of the startup.pl script's "use" statements (line 21). However, what these differences are comprised of is not expressed clearly or explained in further detail. The developer also notes in parentheses that he assumes that the receiver is familiar with the script (line 22-23). More specific information about the statements is thus purposely omitted from the message content; their functions are only implied and not spelled out in further detail. Can these utterances be interpreted as proleptic incidences where the less competent is invited into sharing with the more competent? Do they initiate sequences of anticipatory comprehension? In the subsequent replies, the incomplete utterances are submitted to shared attention and sense-making among nucleus group participants.

[Line]	
	Reply from dev-81, 13 Feb 2008 22:00:56 GMT
1.	Of course, what I forgot to mention below - and sorry if you know that
2.	already - is that whichever perl modules you pre-load in your main Apache
3.	server config via the startup.pl script, you do not need to "use" anymore in
4.	all your perl scripts or Apache/mod_perl handlers. [..]
5.	
6.	**Reply from dev-19, 14 Feb 2008 14:13**
7.	This is the first time that NOT using "use" because it was preloaded is
8.	mentioned. In fact, how would the modules compile(while testing for
9.	example)? Wasn't preloading suposed to do the exact opposite? [..]I'm at a
10.	loss here.
11.	
12.	**Reply from dev-12, 14 Feb 2008 15:51:22 GMT:**
13.	[..] I think that was just a wording mistake. When the process forks, the
14.	loaded modules are shared by the operating system's copy-on-write feature. I
15.	believe that's what he was trying to say.
16.	
17.	**Reply from dev-81, 18 Feb 2008 09:40**
18.	Well, it was not exactly what I was trying to say, but undoubtedly this
19.	reflects my incomplete personal understanding of the process anyway, and I

stand ready to be educated.

What I have until now believed is that perl "code" is in fact "data" for the perl interpreter, and that as such it cannot really be "shared". What I mean is that, as soon as some bit is changed in a "page" of any perl module, that "page" is dirty and must be copied and made private to the one child process. And since there is (in my understanding) not such a clear separation as to which parts in "perl code" are data and which are code, after a while one ends up with a full duplicate in each child anyway. Probably badly explained, but not so in the general sense ?

P.S. What I really meant originally, is that if the speed to make it work was of the essence, it might be easier to (find/grep) and remove the various use Apache-x() from the multiple modules or cgi scripts, and put them all in the startup script. Then later one could go back and refine things, if it makes a difference.

Reply from dev-12, 18 Feb 2008 19:55:03 GMT

Your technical understanding is correct, but in practice most pages remain shared. You can help this by using a tool like Apache::SizeLimit that kills off processes after a while.

mod_perl forum extract 4: Sense-making of incomplete utterances involving nucleus group participants

In the above replies, the general and intended functionality of the startup.pl script's "use" statement is the subject of shared attention among two nucleus group members (dev-81, dev-12) and another semi-central participant (dev-19). With reference to previous postings, dev-19 notes that the statement's preloading function is intended to be applied in the reverse manner (line 7-10). So far we have seen that a specific aspect of the joint knowledge object is attended to by participants representing different regions in the social network. However, it appears that the incomplete utterances concerning the "use" statement are not sufficiently understood and a temporary communication breakdown occurs among nucleus members. A

central nucleus group member (dev-12) attempts a repair sequence, suggesting that this is only a wording mistake (line 13). The expert, or more competent other, thus steps in to assist the sense-making around the incomplete utterances concerning the "use" statement. Dev-12 also indicates that he temporarily adopts another nucleus group participant's perspective in the communication, in asserting "I believe that's what [dev-81] was trying to say" (line 14-15). In the follow-up message, dev-81 points out that "it was not exactly what I was trying to say, but undoubtedly this reflects my incomplete personal understanding of the process anyway, and I stand ready to be educated" (line 18-20). At this stage in the communication process, it is not reasonable to interpret the shared 'here and now' as expanded, reflecting an instance of prolepsis. The partakers may have searched tacit and hidden information for clues to assist their current interpretations, but the deficient and fragmented sense-making do not provide us grounds to claim progress in joint understandings.

On lines 22-29 of mod_perl forum extract 4, dev-81 utters that his understanding of the shared knowledge object (perl code) is improved. The creative assumptions provided when attending to the specific feature of the object ("use" statement) in previous postings have enabled him to fill out the missing pieces. Furthermore, the nucleus group participant explains what he originally meant (lines 31-35) as well as his developed understanding of the process. The most central participant and competent member of the nucleus group confirms the progression of their (partial) joint understanding in replying; "your technical understanding is correct, but in practice most pages remain shared" (line 38-39). An expansion of intersubjectivity among the two nucleus participants (dev-81 and dev-12) is thus confirmed and information can, according to Rommetveit's (1979a, 1979b) framework, be nested to what is already partially known in common. However, the mediated communication quoted in the extracts above also involves a peripheral participant. The initial problem formulated by dev-186, is picked up by central nucleus group participants and triggers a discussion with reference to a feature of the shared knowledge object. Moreover, the incomplete utterances concerning the problematic "use" statements of the startup.pl script are presented in the reply message from a member of the nucleus group (mod_perl forum extract 3, line 21-23).

How can we understand the peripheral participant's (dev-186) role in the development of partial intersubjectivity in the above message exchange? As the initial sender, he was assumed a privileged position in controlling what aspect of the shared knowledge object to penetrate the intersubjective space. However, did he develop a sufficient understanding of the issue to

continue with his local problem solving? Was he invited by the more competent nucleus group participants to step into an expanded intersubjectivity? In the message thread above, he only participated by raising a problem but did not participate in the active meaning making following the initial problem definition. There are hence no observable cues or utterances confirming that his inadequate understanding of the object's functionality is enhanced. Nevertheless, a part of the problem experienced in his 'private world' is brought into focus in the preceding communication among nucleus group participants. Dev-81's reply message can be seen as inviting the co-construction of a greater intersubjectivity in omitting information combined with providing minimal clues about the aspect jointly attended to. Yet, the inviting proleptic incidence was answered by another nucleus member (dev-12). The sense-making following the proleptic incidence did not involve the peripheral participant who identified the original problem. Assuming that dev-186 had a sufficient understanding of the aspect of the shared knowledge object focused upon prior to the interaction, he may have 'attuned' his perspective to the others by reading the replies from the nucleus members. However, there is no evidence in support of these assumptions as he did not participate in further sense-making around the subject. According to the extended contribution theory suggested by Clark and Brennan (1991), "all collective actions are built on a common ground and its accumulation" (p. 222). From this point of view, participants are assumed to contribute and update their common ground moment by moment in order to progress with their joint activities. The grounding model also asserts that partakers in communication tend to seek positive evidence of understanding and try to confirm that the collaborators understood the content well enough for the current purpose. Furthermore, both the presentation and acceptance phase have to be completed for valid contributions to be added to their common ground. In the e-mail exchange above, one can, according to this model, expect that the issuer of utterances will monitor the receivers' understanding in order to 'check' whether the content can be added to their shared understanding. The initial problem formulation issued by dev-186 in mod_perl forum extract 2 was presented for the entire network of participants to consider (Hello @all, line 1). When first picked up by dev-81, a partial understanding of the initial problem is signalled by describing two related system configurations (mod_perl forum extract 3). However, no evidence is provided by dev-186 if the content of the reply is sufficiently understood for carrying on with his local problem. On the other hand, if the utterances were not understood, one could expect some kind of negative evidence and the initiation of repair sequences.

In the discussion thread, the peripheral participant is not contributing any further to the active sense-making. We may see this unbalanced involvement as a result of variations in the two main factors that are assumed to 'shape' grounding processes; namely *purpose and medium*. The former relate to Clark and Brennan's (1991) postulate concerning the *grounding criterion*; stating that a sufficient condition for continuing the collective activity is that partakers mutually believe that the receivers "have understood what the contributor meant to a criterion sufficient for the current purpose" (p. 223). In the case of dev-186, reply messages provided by nucleus group members were perhaps sufficient to carry on with his local problem solving. However, as the message thread evolves (based on the peripheral participants' local problem) into a discussion among more competent participants, the collective purpose of communication seems to change. In the second wave of replies (mod_perl forum extract 4), the sense-making moves beyond the local problem mentioned by dev-186, triggering expert participants to discuss the functionality and semantic meaning of a specific feature of their joint knowledge object (the "use" statement). This again leads dev-81 to fine-tune his understanding of the knowledge object itself (perl code). As he puts it: "What I have until now believed is that perl code is in fact data for the perl interpreter, and that as such it cannot really be shared" (line 22-23). In the final reply message, dev-12 provides positive evidence that the content was understood sufficiently for the current purpose in uttering that the "technical understanding is correct" (line 44). The information can thus be added to the nucleus group members' common ground. However, when interpreting the sequence of messages shown in the extracts above from a grounding perspective, it is unclear what to consider as the collective purpose of the communication. The message thread, which spans several days, involves multiple participants from different regions of the social network. In the first part of the thread, the exchange of utterances mainly addresses the local problem specified by a peripheral participant, whereas the next part focuses on sense-making around a feature of the joint knowledge object attended to in the distributed expert forum. Even if dev-186 did not actively contribute in the consecutive sense-making, he nevertheless played a role by sharing the local problem that triggered the construction of (partially) shared understandings around the structural core of the network. This indicates an intersection of purposes among the local and distributed contexts. The local problem experienced and shared by dev-186 somehow activated the distributed 'knowledge machinery' involving competent participants from the innermost circle of the developer network. When posting the initial problem description, dev-186 may have mediated deficiencies associated with the unfolding knowledge object that matched the experts' wanting structures (see chapter. 2.3.1 on object

relations). The complete message thread can perhaps be seen as a gradual movement from a local problem experienced by a peripheral actor, towards an expansion of intersubjectivity in a distributed expert network.

From the above discussion, it is reasonable to conclude that dev-81 constructed a proleptic incidence based on an interpretation of the initial problem issued by a peripheral actor (dev-186). Moreover, assisted by the more competent dev-12, he did see something that was not yet added to their common ground. This signifies that new information about the shared knowledge object was added to what was already (partially) known to the nucleus group participants. However, apart from specifying the initial problem to enter the intersubjective space, the data do not provide us with sufficient evidence to assess how the peripheral participants' understanding of his local problem develops. At this point, the research design is coupled with limitations. In order to make claims on how distributed processes of intersubjectivity inform and intersect with the local contexts of peripheral participants, a more comprehensive dataset would be required. For example, in a follow-up study on related themes, one can supplement the research design adopted for this case with a logging system that captures the local screen activities from selected participants and video-based observations of localised peer interaction.

Below is another example of sense-making around the structural core of the network. The forum extract is a reply message in which a member of the nucleus group (dev-81) reflects on his understanding of a peer developer's (dev-29) explanation of a new interface invented for enhanced performance on the Linux platform. Note that dev-29 is also an active member of the nucleus group.

[Line]

1. **Message from dev-81, 03 Jun 2008 15:11**
2. I find it interesting that the answers are all different on system (1), but all the
3. same on system (2). I don't have a clue as to what it means, or what it does to
4. my systems, but I trust you do.
5.
6. **Reply from dev-29 03 Jun 2008 16:48**
7. Just to enlighten you. Originally linux on x86 used to use a software

interrupt 0x80 to implement syscalls. With newer processors this is a performance bottleneck. So a new interface (sysenter/sysexit) has been invented to switch from user space to kernel space and back.

This new interface uses a special page that is mapped in the virtual address space of each process the so called Virtual Dynamic Shared Object (vdso). It was originally located at a well known place in the address space of a process because the C library needs to know where it is to make syscalls....

From dev-81, 03 Jun 2008 19:43 GMT

many thanks for the time taken for the explanation.

I still understand only slightly less than 50% of it, but then I don't really need to understand more either. I am just very glad and thankful that there are people such as you who apparently do understand it, upon which we can rely to give us a good perl when we need it, a good explanation when we ask for it, and upon which we might need to call some day when we are really desperate.

And, one other good thing about the explanation below, is that it will undoubtedly provide a number of people like me with some esoteric new vocabulary for the next time the marketing guys ask us why this damn project isn't ready in time. I can just see their faces when we talk to them about boosting performance and avoiding shell code injection by taking advantage of the latest processors and glibc features, and now switching from user to supervisor mode by means of a Virtual Dynamic Shared Object instead of the old pass software interrupt 0x80. Great stuff !

Now if I could just understand why the different choices of a perlio layer influence the location of this VDSO, I feel that I would instantly become a much better perl and mod_perl programmer. Or was it just the fact that I ran the command 3 times in a row ? Hmmm. But don't ! I really do not need to know about that one.

41.
42. P.S. Now that I knew what to look for, there seems to be a good explanation
43. here :
44. http://manugarg.googlepages.com/systemcallinlinux2_6.html
45. Now I feel that my understanding has climbed to at least 55%.

mod_perl forum extract 5: Contributions to partial understandings among nucleus participants

The initial problem introduced to the intersubjective space was issued by dev-81 earlier the same day. He points out a discrepancy in previous answers related to different system configurations. In his own words, "I don't have a clue as to what it means, or what it does to my systems, but I trust you do" (line 3-4). Dev-81 thus indicates a trust in his peers, assuming that they are really trying to provide an answer to the problem, even if, at this stage in the process, he has no clue what the reply means. Now, how can the reply message from dev-29 trigger anticipatory comprehension for dev-81's sense-making of not-yet-provided information? Was dev-81 in any way challenged to construct a new understanding of the intended meaning? According to Rommetveit (1979a), the sender (I) of proleptic utterances may search the receiver (you) for some potential taken-for-granted aspect of his partially shared 'here and now' to confirm this faith. By reading past messages and quoting dev-81's problem, dev-29 can assume that some information is already partially known in common. It follows that when constructing the reply message, dev-29 can draw on elements that are already part of the intersubjective space and leave out redundant information. However, dev-29 also indicates that he brings some new information to enrich the understanding of the problem in order to 'enlighten' the receiver (line 7-15). The information concerns the development of a new Linux interface (sysenter/sysexit) intended to provide better performance for running the system configurations on new processors. What are the cues or prompts included in the message for expanding intersubjectivity? Dev-81 expresses gratitude for the explanation and claims that he still only understands less than half of the content (line 20). Furthermore, he may have learned some "esoteric new vocabulary" that can be useful when briefing the marketing guys in his company (line 27-34). The level at which he understands these processes and related terminology (e.g., shell code injection, glibc features and Virtual Dynamic Shared Object) is not clear. In the last part of the message (line 36-40), he continues to articulate his lack of understanding but encourages dev-29 not to follow up on

this matter. Dev-81 then closes the message thread by uttering “P.S. Now that I knew what to look for, there seems to be a good explanation here [link]” (line 42-44).

It seems as if the explanation provided by dev- 29 consisted of many details that were not directly relevant to the problem at hand. Considering dev-81’s response, “I don’t really need to understand more either” (line 20-21) and “I really do not need to know about that one” (line 39-40), the message was perhaps too saturated to serve as a catalyst for the co-construction of an expanded state of intersubjectivity. Nevertheless, his understanding might have ‘climbed’ slightly as he received a prompt on what to look for and found a satisfactory explanation elsewhere on the Internet. However, I do not find it reasonable to interpret the explanatory message issued by dev-29 as proleptic, ‘boosting’ the construction of a greater intersubjectivity. The message is rather informative on a subject that is not directly relevant to dev-81’s problem. In issuing information that the receiver does not really need to know, the utterances do not necessarily invite the receiver into active sense-making and the co-construction of a greater intersubjective space. The receiver was in this message exchange not sufficiently triggered to participate in filling in the gaps in their shared understanding. On the other hand, the additional information included in the explanation implies that dev-29 is a knowledgeable peer on this matter who can be useful to consult in the future. As dev-81 puts it:

> I am just very glad and thankful that there are people such as you who apparently do understand it, upon which we can rely to give us a good perl when we need it, a good explanation when we ask for it, and upon which we might need to call some day when we are really desperate (mod_perl forum extract 5, line 21-26).

This quote may also point towards another ‘sustaining facet’ of the intersubjective space; namely, encouragement and mutual recognition. According to Matusov (1996), processes of intersubjectivity transcend specific and time-limited joint activities and are “not only the basis and derivative of the joint activity but also the social glue of different sociocultural activities” (p. 30). The praising of peers, as indicated by dev-81, may perhaps be significant for continued participation in the open-source development project.

When interpreting the message content of these extracts, the concept of ‘here and now’ introduced in Rommetviet’s (1979a, 1979b) framework for intersubjectivity may appear

problematic. Since the discussion between dev-29 and dev-81 is electronically stored in the message archive for later retrieval by other participants, we cannot completely rule out that the content of dev-29's explanation can initiate processes of shared prolepsis on a later occasion. This issue points towards a potential development in the framework for intersubjectivity that also accounts for mutually accessible and electronically stored communication. However, if we see the message thread as the distributed form of 'here and now', it is reasonable to conclude that dev-29's utterances did not trigger processes of shared prolepsis.

When looking at the message sequences presented in this chapter, the active meaning making commonly occurs between central participants. However, peripheral participants contribute by posting initial problem formulations based on their experience as users of the software in their local practices. Peripheral participants thus play a role in the intersubjective space by formulating initial problems experienced in their local work contexts. In this way, they contribute to the joint meaning making by describing problematic features of the knowledge object to enter the field of shared attention. However, the proleptic incidences 'boosting' progression towards a greater intersubjectivity rather occurred in the second wave of replies among more competent participants situated in the core regions of the network. As a consequence of peripheral actors' limited participation in the active meaning making following initial problem definitions, the data do not provide us with sufficient evidence to infer that this cluster of participants advances intersubjectivity as a result of processes of shared prolepsis. Nevertheless, the problems experienced and shared by peripheral participants are brought into focus in the intersubjective space and may serve as 'raw material' for the co-construction of a greater intersubjectivity among central participants. In this sense, we can perhaps designate a *pre-proleptic stage* in the threaded communication of the mod_perl network. This can be seen as an early step in the development of shared prolepsis in which a problematic feature associated with the joint object of knowledge is introduced into the intersubjective space by a less competent participant. As it is addressed to a 'generalised other participant' represented by a large number of receivers, it may prepare the conditions for the further development of shared understandings by pointing out which aspects of the object are brought into focus in reply messages. It follows that the pre-proleptic stage contains utterances that might nurture instances of shared prolepsis in subsequent replies, even if the initial sender does not actively contribute to the sense-making. For the pre-proleptic stage to evolve into an instance of prolepsis, the data suggest that the local problems

shared by peripheral participants need to be picked up by more competent central developers and re-defined into assumptions about a piece of shared knowledge. The original local problem may then influence processes of shared prolepsis and contribute to the construction of partially shared understandings around the structural core of the network. This suggests that the distributed process of intersubjectivity involving participants from the different regions of the network is enhanced in two steps; a pre-proleptic stage issued by a peripheral participant, and a further developed proleptic instance from central participants. The complete message threads can then be seen as a gradual movement from experienced local problems towards an expansion of intersubjectivity in a distributed expert network.

According to Rommetveit's (1979a, 1979b) framework, human dialogue transpires from the partly shared 'in between' social reality. The coordinates displayed in Figure 1 predict that between the 'I' and 'You', potential states of intersubjectivity can emerge along three dimensions. Throughout the theoretical discussion in chapter 2.1, Rommetveit's original propositions are modified and re-contextualised to a technology-mediated scenario involving joint object-orientation. The first proposition suggests that discussions involving shared attention towards objects of knowledge are characterised by an "attunement to the attunement of the other" and that partial intersubjectivity can develop among the participants, but are reliant on reciprocal 'tuning' of perspectives and 'fixation of perspectives' on shared objects (p. 13). Based on synthesis of key points made in the data presentation and discussion above, this proposition can be differentiated further by incorporating the role of shared prolepsis. Figure 14 below summarises different findings and puts forward a framework for understanding development of intersubjectivity in mediated expert networks. The previous discussion suggests differentiated processes of shared prolepsis in two phases or steps as the proleptic utterances approach the more competent others situated in the core regions of the network. The pre-proleptic stage in the model is set exterior to what is partially known in common (intersubjective space) in order to represent the 'spots' brought into shared attention that address some feature of the knowledge object. However, for the utterances to develop into proleptic instances and boost processes of intersubjectivity, it is suggested that they need to be adapted and re-defined by *a more competent participant* (CoP) to reflect assumptions about shared knowledge. The latter adaptation presupposes shared information about an aspect or deficient feature of the joint object that is not yet introduced in the expert forum. The developed proleptic incidence is represented as a 'spot of shared attention' moving along the horizontal axis of the model below, on a continuum from local experience towards a

shared knowledge object in distributed time and space; a prompt that challenges anticipatory comprehension among more competent participants and *boosts the construction of a greater intersubjective space*. The role of shared prolepsis can then be seen as a catalyst for object-oriented discussions that originate from local experience and 'foresee' further development of the joint knowledge object.

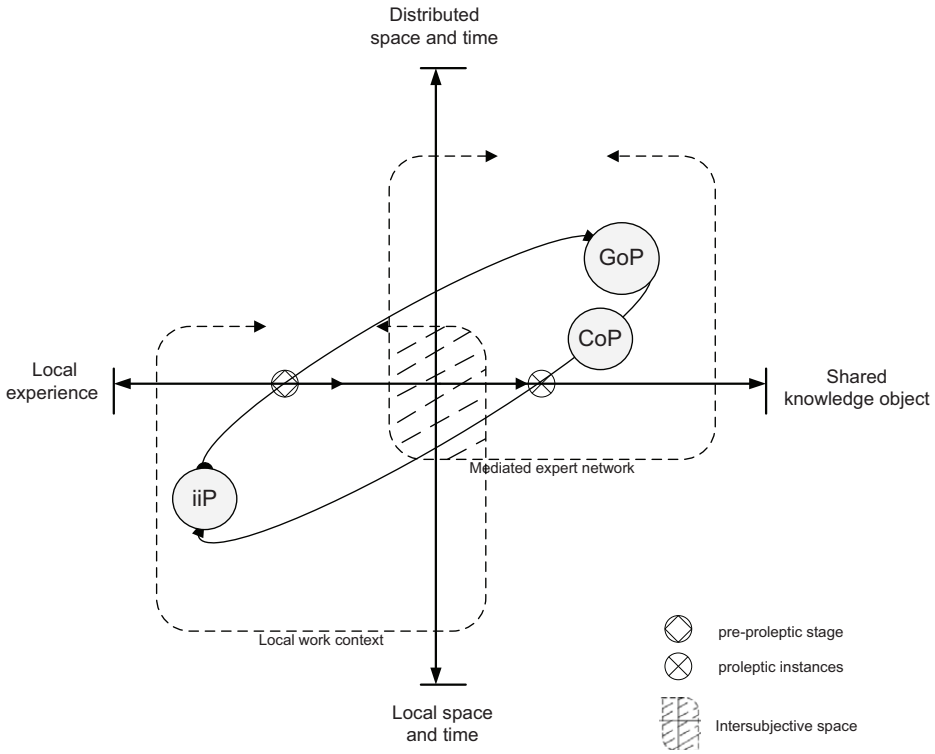


Figure 14: Framework for developing intersubjectivity and shared prolepsis in mediated expert networks

Departing from the time and place matrix developed in chapter 5.1 (Table 5), the vertical axis in the above model represents the transitions in time and space; from a local work context to joint sense-making in mediated expert networks. The two intersecting contexts are thus indicated using dotted lines in the lower left and upper right squares of the coordinate system. In this intersection, the coordinates 'I and You' in Rommetveit's original model are modified to *issuer of initial Problem* (iiP) and *Generalised other Participant* (GoP) and represented along the diagonal path. The data presentation indicates that iiPs have to adapt the message

content to a potentially large number of receivers (GoPs). However, once the request is picked up by another participant, the relation can change into a two-way interaction sequence.

The development of shared prolepsis across contexts in the framework above, suggests that the iiP is in a privileged position in controlling what feature of the joint knowledge object should enter the field of shared attention. However, one can argue that this control is only temporary and potentially constrained by the iiP's ability to formulate an adequate description of the local problem. The threaded message extracts discussed in this chapter point towards the conclusion that peripheral participants issuing iiPs are not likely to engage in the active sense-making as the utterances evolve into a proleptic instance issued by a more competent participant (CoP in the upper right square of the model). It follows that the set of aspects to enter the field of shared attention are influenced by the CoP's interpretation, adaptation and re-definition of the original problem. In this process, the purpose of conversation may also change in the second or third wave of replies (see e.g., mod_perl forum extract 4). The expansion of intersubjectivity can then move beyond the initial local problem, triggering expert participants to further discuss functionality of a specific feature of their joint knowledge object. It is thus reasonable to infer that the set of aspects jointly attended to subsequent to an instance of shared prolepsis in the mod_perl project is not fully controlled by either iiPs or CoPs. The data presentation and discussion rather indicate the former having a pragmatic role in providing the pre-proleptic utterances that serve as the 'raw material' for CoPs to construct instances of shared prolepsis.

7 Concluding discussion

In this chapter, the key points made in this thesis are summarised and synthesised into a more coherent understanding. Departing from the initial aims stated in the introduction and research questions for the case studies, the main threads of the argument are drawn together. Then, I examine the extent to which the aims have been achieved. Finally, the thesis' original contribution to knowledge and its limitations are assessed.

In order to follow up on the first explorative aim, the artefact-mediated relation between the local problem solving of one individual software engineer and distributed expert networks was investigated in chapter 5.4.3. By sharing programme code and using resources within developer networks (e.g., discussion forums), software programmers can meaningfully interact with other programmers who are separated in time and space. However, a common strategy reported by the respondent is to locate and assemble code from developer networks by posing software-specific search strings in Google. In the discussion that follows, it is suggested that the respondents' artefact-mediated approach to attaining new knowledge follows two main patterns. The first is characterised as the *monitoring approach*, in which the knowledge-seeking activity is not directly relevant for daily problem solving. It is described as a forward-oriented trajectory, used to know what is occurring in the field. The latter approach is characterised by *task-related short cycles* involved in daily problem solving. The short-lived trajectories are associated with the task at hand. A related looping pattern has been identified by Nerland (2007) in the article "Knowledge Cultures and the Shaping of Work-based Learning: The Case of Computer Engineering", which draws on empirical data from several engineers participating in the ProLearn project. In her words,

> The problem-solving activities [of computer engineers] are restricted in terms of time. The working days are often characterized by a series of 'short-term loops' where processes of inquiry and specific problem-solving coalesce. The quest for quick solutions and knowledge application may give priority to surface forms of learning and undermine possibilities for more profound engagements with knowledge (p. 62).

Less restricted in time and space, she designates a parallel monitoring activity. The computer engineers seem to monitor the technological advancements in their areas of interest in a broad sense. However, the case of D19 indicates that the two patterns operating on different time

scales are not mutually exclusive or even clearly distinguished. They may overlap and operate simultaneously. Informed by the notion of epistemic trajectories suggested by Lahn (in press), the movement from problems experienced in a local work context towards mediated expert networks is interpreted as a knowledge-intense path where the content consists of chunks or fragments derived from distributed objects of knowledge. The partial knowledge content linked with this category of trajectories is also implied by the second study. The sequence of messages presented in mod_perl extracts 2-4 indicates that the problem experienced in a local context triggered distributed sense-making within a mediated expert network; a part of the project's joint knowledge object is moved across contexts and brought into shared attention. The case studies included in this thesis thus imply that the local problem solving of software developers intersects with knowledge development in mediated expert networks. This interdependency implies that snippets of code and 'chunks of knowledge' traverse rather flexible epistemic trajectories. It is along such pathways that search engines, discussion forums and shared or open-source code are found to be key artefacts with the potential to mediate objects' knowledge.

In the article "Knowing in a System of Fragmented Knowledge", Bruni and Gherardi, et al. (2007) describe a case of remote cardiological consultation "characterized by technologically dense practices in a setting in which human actors and technological objects work together" (p. 1). The study suggests that information and communication technologies (ICT) restructure workplaces into *systems of fragmented knowledge* consisting of reciprocal relations among heterogeneous elements. They maintain that these elements must somehow be held together in order for knowledge workers to master the knowledge needed to perform in a professional practice. The incomplete knowledge objects that traverse across contexts in software development emerge as open to modification (reusable/adaptable) and less dependent on a pre-defined system of heterogeneous elements. Moreover, the data suggest that practitioners do not necessarily need to fully understand the knowledge objects shared in distributed time and space in order to transform what is 'known' into 'knowing' in their local practice. By connecting to the argumentative space of distributed expert networks, software developers can, in a short time, acquire access to knowledgeable peers, get assistance to resolve a local problem and then leave without further obligations. In this way, reusable small parts (or scripts) of an open source code can be separated from the original knowledge object and given a new meaning outside the system of relations in which it first appeared. Under such circumstances, the ideas that professional knowledge is structured by authorised access can be

disputed. The tendency to consult and participate in open source developer networks represents alternative paths to acquiring practice-related competency and recognition in the field. Self-made experts and individuals from other vocational groups may also join and contribute to these distributed arenas, blurring the distinctions between professions and other forms of work organisation.

The second phase in the exploration of D19's knowledge sources identifies a discussion thread from the Sun Developer forum as an analytic example of a key artefact. The forum's user interface provides several options for participation in the expert network. For example, it allows programmers to contribute to and comment on source code samples that are included in the message bodies of postings to the forum. In this thesis, programme codes are previously conceptualised as relational objects with reference to contemporary theories of object relations (chapter 2.3). Yet, the shared code's potential to initiate the S-Ko dynamic suggested by Knorr Cetina (1997, 1999, 2001a) is not straightforward in this context. If the knowledge seeker has agency towards the shared code, such a dynamic might be set in motion. However, in the example thread, reply messages indicate that the problem is solved and the topic is considered closed by the moderators of the forum. It follows that the object is no longer open-ended, mediating 'deficiencies' in order for the relation to continue over time. Another problematic issue is that the structures that may trigger participation are not necessarily built into the software program (shared code) but are partly associated with an external incentive system (Duke Dollars). According to the theoretical propositions deduced from Knorr Cetina's theory (chapter 2.3.1), nourishing object relations is rather based on personal desire and extended engagement with open-ended objects, not on brief participation through symbolic incentive systems. These issues point towards potential limitations of the notion of knowledge objects and related problems with regard to adopting a one-sided interpretation of such entities. In the distributed expert systems of a knowledge economy, there might be multiple incentive systems outside the objects themselves that interact and influence the motivation to participate in developing joint objects of knowledge. Otherwise, why do developers volunteer their time and effort to free and open source projects? There is something about these objects that initiates and gives direction to powerful incentive structures. A comprehensive survey on these matters conducted by Lakahni & Wolf (2005) suggests *intrinsic motivation* as "the strongest and most pervasive driver" behind developers' contributions in free and open source software projects (p. 3). The incentives to participate are partly attributed to emotional factors within the individual, such as creativity, intellectual

stimulation and improving his/her own programming skills. On the other hand, the majority of respondents were skilled and experienced professionals within ICT-related jobs, and about 40% of them were paid to participate in free and open source projects. It follows from this that the will or desire to participate and learn from these scenarios can entail several partly interdependent incentive systems in which the relational features associated with objects of knowledge may be one of several influential factors.

The second research aim of this thesis conveys an intention to describe the network structure of a distributed open source development project and identify central and peripheral sub-groups within the network (p. 3). To what extent were the elements included in this descriptive aim achieved? In a retrospective light, I regard this aim as partially achieved. The formal methods associated with a social network approach were functional in order to obtain an overview of the communication structure in the mod_perl project during the six-month period of observation. Concepts of density and centralisation gave a general impression of the overall network's cohesion. However, with regard to the second element of the aim—to identify central and peripheral sub-groups of participants—the inquiry process did not quite proceed as expected. Various formal methods were triangulated to identify sub-structures or cliques. In the context of the mod_perl project, clique measures proved especially useful for defining the most cohesive central sub-group forming around the structural core of the network. By using a strict clique algorithm, four well-bounded cliques were found. Furthermore, the overlapping sub-graphs (Figure 8) provided a visual aid to trace central participants' co-membership in cliques and to demarcate a group around the central core. This cluster is referred to as *the nucleus group* in the content analysis targeting processes of intersubjectivity in chapter 6.6. Problems emerged when searching for a sub-group of peripheral participants to serve as a contrast group in the content analysis. First of all, density measures indicate that most participants can be defined as peripheral actors in the social network. When performing the formal group measures in the peripheral regions, no cohesive sub-groups were found. This means that the intersubjective content of messages is not directly comparable with that of the nucleus group, as its members are not likely to communicate frequently with each other. On the other hand, nucleus group participants correspond relatively frequently with each other. In network terminology, the former represents 'disconnected nodes' and the latter 'connected nodes'. As a consequence, the shared understandings of peripheral participants might be restricted due to a history of limited inter-group participation and not having the opportunity to sufficiently ground their communication

(Clark & Brennan, 1991). Another problem is that the peripheral participants tend to be novice users of the mod_perl software, whereas the nucleus participants mainly are core developers. As such, they represent different roles and can be argued to relate to the joint knowledge object differently. This may again influence what content or 'aspects' are brought into joint focus in the discussions. These are limitations that are further reflected upon in the next chapter.

Network-level studies of open source development projects report that interaction patterns among participants vary depending on their roles, with core developers as the most active contributors (Crowston, Wei, Li, & Howison, 2006; Long, 2006). For example, making use of a social network analysis of three projects from SourceForge.net, Long and Siau (2007) suggest that the interaction pattern within open source projects evolves from a single hub at the beginning of a software project's life cycle into a core-periphery model as the project matures. In this thesis, another approach is proposed while using the core-periphery pattern as a network-level contrast. The task has been to analyse message content involving processes of intersubjectivity around shared objects of development. While studying the message content in between the central and peripheral regions of the mod_perl network, constructive aspects of a mediated intersubjective space emerged. The relative density of interactions among participants from different regions reflects the transition from problems experienced in a local context towards experts' distributed sense-making. Namely, peripheral participants tend to introduce preliminary problems that serve as building blocks of initiation for the (co)construction of a greater intersubjectivity among central participants. As the initial problems become shared with a large number of 'generalised other participants', an important implication is that the constructive utterances should induce or hint towards some lacking feature in the joint knowledge object that has not yet been introduced into the shared space. The inducing content of such initial problems posed to a large number of 'generalised other participants' may allow more competent peers to fill in the blanks (i.e. tacit and hidden information) and creatively reformulate these utterances into 'proleptic instances' that not only are more complete in terms of resolution here and now but also foresee further development of the project's joint knowledge object.

In the analysis related to processes of intersubjectivity in the mod_perl project, the key findings indicate that participation in active sense-making discussions and the development of shared prolepsis diverge in the peripheral and core regions of the network. The extracts

presented in chapter 6.6.1 involved participants from both peripheral and central regions of the social network. A closer inspection of the message structure points towards variations in post and reply sequences in the contrasting network regions. The peripheral participants tend to issue initial problem descriptions, whereas the nucleus group is more likely to follow up with advisory replies and contribute to more extended sense making, sometimes spanning several days. When the proleptic instances evolve into sense-making discussions among more competent central participants, the collective purpose of communication may also change. For example, in mod_perl forum extract 4, a set of aspects is brought to shared attention, triggering expert participants to engage in joint meaning making around the semantic significance of a particular function to extend the local problem issued by the peripheral actor. By interpreting the locally experienced problem issued by the peripheral participant, more competent central participants were able to construct proleptic utterances and catalyse the progression towards a greater space of intersubjectivity. Yet, the conception of 'here and now' suggested in Rommetveit's (1979a, 1979b) framework for intersubjectivity appears insufficient to explain how these processes unfold across the mediated contexts of software development. Since the discussions are electronically stored, they serve as common reference for later use, but in spite of this, we cannot completely rule out the possibility that current explanations can initiate processes of shared prolepsis on later occasions. As a consequence, a new framework for intersubjectivity is proposed which takes into account technology-mediated communication by incorporating a modified (distributed) notion of time and space along the vertical axis (Figure 14). In order to account for the intersections between a local work situation and distributed expert networks, the model also suggests a reformulated continuum along the horizontal axis, from problems experienced in a local work context towards joint sense making in mediated expert networks. The data presentation and analysis suggest that the issuers of initial Problems (iiPs) are not likely to engage in the active sense making following proleptic instances. However, iiPs appear to assume a pragmatic role in providing pre-proleptic utterances that inform further development of instances of shared prolepsis among more Competent other Participants (CoPs). The developed proleptic incidence is therefore represented as a 'spot of shared attention' moving along the horizontal axis in the model, a prompt that challenges anticipatory comprehension among more competent participants and stimulates the construction of a greater intersubjective space. An important implication of this is that the developed instances of shared prolepsis serve as a catalyst for object-oriented discussions. This, in turn, implies that problems experienced in a local context can contribute to knowledge development in the distributed expert network.

Therefore, the vast number of peripheral participants in the network creates an advantage when they 'feed' the central core with pre-proleptic utterances. As pragmatic elements in a 'collective learning system', the swarm of actors on the outer edges of the developer network monitors the functionality of the shared knowledge object and contributes with prompts and cues that might trigger further development cycles in the core region. However, for pre-proleptic utterances to progress into an instance of shared prolepsis, the data suggest that the local problems shared by peripheral participants need to be re-defined among more competent, central developers into assumptions about a piece of shared knowledge that has not yet been introduced into the intersubjective space. This implies that pre-proleptic utterances should be sufficiently 'imprecise' and provisional, suggesting some lacking feature in the knowledge object that invites re-formulation and further sense-making discussions among competent peers.

The significance of constructing reply messages with content that is relevant for the receivers' problem and at a sufficient level of detail is another point made in the data presentation (mod_perl forum extract 5). The former implies the capacity to de-centre and temporarily adopt the other's perspective. By reading the initial problem formulation and previous postings stored in the mutually accessible database, the replier can make creative assumptions concerning what is already partially known in common and about projections towards the needs of local development. On this basis, participants constructing reply messages can assume that some elements are already part of the intersubjective space and can leave out redundant information. On the other hand, explanations provided in replies will often contain excess information and address issues that are of limited value for the problems experienced in a local context. This points towards the conclusion that saturated messages containing too much presupposed shared information and failing to address relevant aspects of the joint knowledge object may inhibit processes of shared prolepsis. In issuing detailed information that the receiver does not really need to know, the utterances are unlikely to invite the receiver into the co-construction of a greater intersubjectivity. However, even if saturated messages are not closely connected to the problem at hand, they may contain links or pointers as to where one might look further. Hence, the further expansion of an intersubjective space can still function on the basis of saturated messages by referring to other explanatory sources for resolution, even if the expansion was not enabled by a proleptic incidence. It is beyond the scope of this thesis to deduce the reasons for this, but it identifies another area for further research.

7.1 Thesis contribution and evaluation of limitations

In terms of originality, a potential contribution lies in the conceptual framework developed for studying processes of intersubjectivity around shared objects of knowledge; the thesis suggests an analytic lens for understanding these phenomena in the intersection between local experience and mediated expert network. Furthermore, when discussing the notion of epistemic trajectories in chapter 2.4, I added a dimension to the concept in order to reflect on the knowledge content. I put forward that the content moved along these paths can be understood as partial objects of knowledge derived from mediated expert networks.

With respect to research methodology, the combination of a social network analysis and content analysis for studying processes of intersubjectivity online can be seen as an innovative approach. By rendering the structure of communicational ties visible, we obtain a useful overview of the density of interactions that constitute an online expert community. Moreover, the combination of a social network approach and a content analysis of text-mediated interactions provides a tool for studying the relation between processes of intersubjectivity and participants' relative placement in the overall network structure. In identifying tightly bound sub-groups in the core regions and peripheral participants on the outermost borders of the overall network, one can acquire useful contrasts for a more detailed content analysis. By tracing and analysing the content of messages at an 'utterance level', one can study the 'hot spots' in which the active sense-making occurs and if/how the level of integration in the overall network structure influences processes of intersubjectivity and shared prolepsis. However, to what extent does the account given of the objects and social processes throughout this thesis accurately represent the phenomena to which they refer? How are the conclusions warranted and what are their limitations? As a critical remark, the following paragraphs look back at the research process, considering the thesis' limitations and how it could have been strengthened.

When deciding on the initial research aims and overall theme of this thesis, the first drafts targeted only the software engineers participating in the ProLearn project. However, as the ProLearn project progressed, I realised that additional data were needed to pursue my interest in object-oriented expert networks and processes of intersubjectivity. The empirical evidence collected jointly in the ProLearn project did not provide adequate 'interactional data' for

researching these issues. On this basis, a second case study was included in the thesis (chapter 6). This involved a restructuring of the research process; I needed to perform some kind of data reduction in the first case due to lack of time and resources. In accordance with the first explorative aim stated in the introduction, I suggested in chapter 5.3 that a single case study would be functional. One software engineer among the 10 participating in the ProLearn project was thus purposely selected as the explorative case. In a retrospective light, the inferences made from this case might have benefited from drawing more extensively on the reports given by multiple respondents from this group. This could perhaps have provided analytic generalisations about the range of artefacts applied as knowledge resources in software development and how they integrate. Furthermore, drawing on evidence from multiple respondents would afford contrary cases to improve the quality of inferences about epistemic trajectories leading to distributed expert networks. For example, a form of pattern-matching could be applied to compare empirical patterns with a predicted one. According to Yin (1994), the internal validity of the study would be enhanced if the patterns coincided in multiple cases. The case of D19 indicates that such knowledge intense paths can move in many different directions. Drawing on reports from several software engineers might have provided additional evidence to warrant analytic inferences about artefacts' potential to mediate relational aspects associated with objects knowledge and a stronger thesis on epistemic trajectories. However, the results might be employed in later comparative analysis to investigate how the indentified knowledge seeking pattern matches a population of software developers (e.g., by multiple case study design). The single case of D19 can then provide modest analytic generalisations to other cases representing somewhat similar theoretical conditions. Even so, I will emphasise that the case is not sampled from a larger universe of cases to provide generalised conclusions. The notion of external validity does not carry the same connotations in the case study research design adopted for this thesis as in conventional quantitative research. To infer wide-ranging conclusions about general knowledge seeking patterns from this case would be inappropriate.

What steps are carried out to ensure accuracy and rigour of interpretations in this particular explorative case study? When describing the artefacts applied by D19 in chapter 5.4.3, different data sources were triangulated to complement and verify each other. Semi-structured interviews, self-reported learning logs, various documents and computer interfaces were applied in the analysis to validate findings. Furthermore, during ProLearn project seminars, these sources were interpreted by several investigators and the findings discussed in the light

of various theoretical perspectives. A form of *respondent validation* was also carried out in the focus group interview. Throughout the group interview, we asked the respondents to verify and reflect on our preliminary findings from their earlier participation in personal interviews and learning logs. In this way, the organised discussions (under the guidance of a moderator) elaborated on central issues and challenged our interpretation of the other data sources. The idea was to strengthen the empirical grounds for analysing epistemic trajectories by focusing the discussions on central knowledge objects. The focus on knowledge objects partly verified our preliminary findings and produced supplementary data about software engineers' 'knowledge profile' and preferred knowledge resources.

The second case study included in this thesis started out by applying theoretically derived counting techniques informed by a social network approach (chapter 6.3). Post and reply messages in the mod_perl project's online communication were dichotomised and converted to a binary data matrix. This offered a means to organising the corpus of data and getting a sense of the whole network structure and its sub-groups. As the empirical analysis progressed, decisions were made to render selected message extracts as a form of data reduction. A relevant issue is how the inferences made from these data are warranted. Can we say that the inferences drawn from the selection of message extracts are based on a critical investigation of the whole data corpus? Or, do the results, as Silverman (2010) notes, "depend on a few well-chosen examples"? (p. 276). This problem is also referred to by Bryman (1988) as anecdotalism; the tendency to apply brief conversation 'snippets' of data to provide evidence of a particular argument. Thus, the representativeness of the discussion threads applied in the argumentation on shared prolepsis in chapter 6.6.1 can be questioned. They are only fragments of a large message archive. However, in order to obtain a balanced description of the targeted processes, I also applied data extracts showing 'contradictory cases' in which the sense-making associated with shared prolepsis malfunctions and fails to be addressed in subsequent postings. Although formal methods informed the selection of participants (from the core and peripheral network regions) to be included in the content analysis, I do not find it reasonable to claim that findings are representative of the whole population of participants. The rationale here was not to provide statistical sampling, but rather an organising structure for a theoretical discussion. I will suggest that this case can be assessed for its power to explain theory rather than its typicality. The inferences on shared prolepsis can then be seen as significant in the particular context studied and within the conceptual approach adopted for

this thesis. Perhaps also the logical connections synthesised in Figure 14 can be transferred to related research contexts.

In the social network analyses structuring the data set, measurement errors may have influenced the study's reliability. The measurements in this case are attached to the collection of ties among dyads of interacting participants. According to Wasserman & Faust (1994), "these measurements may differ from the "true" structure of the network" (p. 59). They maintain that social networks can operate on different levels depending on what relational properties one defines as units of analysis and what portion of the network one is targeting (i.e. whole network structure or subsets of participants). For the descriptive and sorting purpose of the second case study, one specific type of relation was selected as the basic unit for constructing the network; reciprocal communicational ties among contributors on the mod_perl project's mailing list. All the ties within the specified time slot were counted in order to provide a census for a full network analysis. However, this procedure may not have measured *all possible ties* among the participants. The binary measurements are assigned by using one of several mailing lists in the Apache open source community. It follows that contributors on the mod_perl project's mailing list may also be co-developers in bordering open source projects and form communicational ties that are not represented in the social networks rendered in chapter 6.5.1; although it remains difficult to assess the difference between the 'true network structure' and what is observed using a single message archive. Constructing a socio-matrix consisting of all possible ties would be far beyond the restricted format of this thesis. Despite this concern, I still regard the restricted social network study as reasonably accurate for the particular organising purposes of this thesis. The social network data provided a sufficient level of precision to construct an overview of communicative relations in the targeted development project according to the intention; namely, identifying central and peripheral participants for a further content analysis targeting processes of intersubjectivity.

7.2 Indications for further research

Throughout this thesis, several issues and directions for future research are pointed out. In the following, these are summed up in two interrelated tracks:

1. Comparative studies on the mediation and transformation of fragmented content associated with distributed objects of knowledge in professional learning.
2. Further research on how the expansion of intersubjectivity in object-oriented expert networks intersects with practitioners' localised problem solving.

Track 1 is grounded on the insufficient data and inherent limitations in the conceptual framework of this thesis. When discussing notions of epistemic trajectories in chapter 2.4, I suggested *fragments of knowledge objects* as the characteristic content moved along the 'knowledge intense pathways' forming in between local practitioners and distributed expert networks. However, we still lack adequate indicators for observing how such fragments are transferred across context and further adapted in professionals' local practices. Are there several intersecting trajectories operating simultaneously on the knowledge content? Will the pieces 'spilled over' from expert networks acquire new properties based on local experience and be brought back to the distributed communities in which they first appeared? The data collected for the initial case (chapter 5) were insufficient to explore these issues. Additional data would be needed in order to observe the dynamic and partial knowledge content mediated along these paths. For example, a combination of video observations, logging of screen activity and personal interviews in a local work context can be used to capture potential transformations in knowledge content and how practitioners relate to partial objects of knowledge.

Track 2 points towards areas for future work on object relations, shared prolepsis and processes of intersubjectivity in software development. For the purposes of the second case study, I have described a conceptual approach (chapter 3) and a combined research framework (chapters 4 and 6) for studying these phenomena across contexts. The suggested framework can be seen as a simplified core-periphery model of how shared prolepsis and process of intersubjectivity develop in technology-mediated social networks. Thus, a spatial metaphor is applied in order to analyse and explain the structural relationship between

participants' local experience and the active sense-making around joint objects of knowledge in distributed time and space. An underdeveloped issue in this thesis concerns how the expansion of shared understandings in object-oriented networks intersects with the localised problem solving of professional practitioners. In order to gain an understanding of how the 'silent observers' of an expert forum enhance their partial understandings and perhaps develop new knowledge in a local context, a richer dataset is needed. The communicational dyads accounted for when constructing the social network in the second case study did not include non-interacting participants. Only reciprocal communicational ties were collected from the project's message archive. This indicates a potential direction for a follow-up study, to trace non-interacting subscribers on the mailing list and analyse how/if proleptic instances issued by competent participants influence their partial understanding of the joint knowledge object. The expansion of intersubjectivity in the tacit regions of the network (following proleptic utterances) could thus be pertinent to explore further. Another relevant theme for further research along this track is to explore how the senders of initial problem formulations benefit from the preceding discussion among core participants. When the former shares a problem experienced in a 'private world' and an aspect of his/her joint knowledge object is brought into focus, how can the proceeding sense-making 'boost' local processes of intersubjectivity? Further theoretical and empirical work is needed to elaborate on the relation between instances of shared prolepsis constructed in distributed expert networks and the localised process of intersubjectivity.

8 References

Allaby, M. (Ed.) (1999) A Dictionary of Zoology. Oxford: Oxford University Press.

Allen, R. (Ed.). (1999). *Pocket Fowler's Modern English Usage*. Oxford: Oxford University Press.

Apache.org. (2006). The Apache HTTP Server Project. Retrieved Okt. 3, 2006, from http://httpd.apache.org/ABOUT_APACHE.html

Appleton, B. (2000). Patterns and Software: Essential Concepts and Terminology. Retrieved Sept. 29, 2008, from http://www.cmcrossroads.com/bradapp/docs/patterns-intro.html

Baker, M., Hansen, T., Joiner, R., & Traum, D. (1999). The Role of Grounding in Collaborative Learning Tasks. In P. Dillenbourg (Ed.), *Collaborative Learning. Cognitive and Computational Approaches*. Amsterdam: Pergamon.

Ball, L. J., & Ormerod, T. C. (2000). Putting ethnography to work: the case for a cognitive ethnography of design. *Int. J. Hum.-Comput. Stud., 53*(1), 147-168. doi: http://dx.doi.org/10.1006/ijhc.2000.0372

Bekman, S., & Cholet, E. (2003). Practical mod_perl. Retrieved Feb. 11, 2008, from http://modperlbook.org/

Bekman, S., & MacEachern, D. (2009). mod_perl User's guide - Installing mod_perl 2.0. Retrieved Nov. 3, 2009, from http://perl.apache.org/docs/2.0/user/install/install.html#Prerequisites

Bell, J. (2006). *Doing your research project: a guide for first-time researchers in education, health and social science*. Maidenhead, Berkshire, England: Open University Press.

Benkler, Y. (2006). *The wealth of networks. How social production transforms markets and freedom*. New Haven, Conn.: Yale University Press.

Blackburn, S. (1996). *The Oxford Dictionary of Philosophy*. Oxford: Oxford University Press.

Blackburn, S. (2008). *The Oxford Dictionary of Philosophy*. Oxford: Oxford University Press.

Blakar, R. M. (1979). Preface. In R. Rommetveit & R. M. Blakar (Eds.), *Studies of Language, Thought and Verbal Communication* (pp. 1-1). New York: Academic Press.

Bober, M. J., & Dennen, V. P. (2001). Intersubjectivity: Facilitating Knowledge Construction in Online Environments. *Educational Media International, 38*(4), 241-250.

Borgatti, P. (2005). Centrality and network flow. *Social Networks, 27*(1), 55-71.

Borgatti, P., & Everett, M. G. (1999). Models of core/periphery structures. Retrieved from www.analytictech.com/borgatti/peripheries.doc

Borgatti, P., Everett, M. G., & Freeman, L. C. (2002). UCINET for Windows: Software for Social Network Analysis. Harvard, MA: Analytic Technologies.

Bourdieu, P. (1977). *Outline of a theory of practice*. Cambridge: Cambridge University Press.

Bourdon, S. (2002). The Integration of Qualitative Data Analysis Software in Research Strategies: Resistances and Possibilities. *Forum: Qualitative Social Research,* (2). Retrieved from http://www.qualitative-research.net/index.php/fqs/article/viewArticle/850/1846

Bridgeman, T. (2005). Thinking Ahead: A Cognitive Approach to Prolepsis. *Narrative 13*(2), 125-159.

Brown, J., & Duguid, P. (2002). *The Social Life of Information*: Harvard Business School Press.

Bruni, A. (2005). Shadowing Software and Clinical Records: On the Ethnography of Non-Humans and Heterogeneous Contexts. *Organization, 12*(3), 357-378.

Bruni, A., Gherardi, S., & Parolin, L. L. (2007). Knowing in a System of Fragmented Knowledge. *Mind, Culture, and Activity, 14*(1), 83-102. doi: 0.1080/10749030701307754

Bryman, A. (1988). *Quantity and quality in social research*. London: Unwin Hyman.

Calhoun, C. (Ed.). (2002). *Dictionary of the Social Sciences*. Oxford: Oxford University Press.

Callon, G. (Ed.) (2002) Dictionary of the Social Sciences. Oxford: Oxford University Press.

Castells, M. (2000). *The rise of the network society* (2nd ed.). Oxford ; Malden, Mass.: Blackwell Publishers.

Castells, M. (2004). *The Network society a cross-cultural perspective*. Cheltenham: Edward Elgar.

Castells, M., & Cardoso, G. (2006). *The Network society from knowledge to policy*. Washington D.C.: Johns Hopkins Center for Transatlantic Relations.

Clark, H., & Brennan, S. (1991). Grounding in Communication. In L. B. Resnick, R. M. Levine & S. D. Teasley (Eds.), *Perspectives on socially shared cognition* (pp. 127-149). Washington DC: American Psychological Association.

Clark, H., & Schaefer, E. F. (1989). Contributing to discourse. *Cognitive Science, 13*, 259-294.

Cole, M. (1996). *Cultural psychology : a once and future discipline*. Cambridge: The Belknap Press of Harvard University Press.

Cole, M. (2002). Culture and development. In H. Keller, Y. H. Poortinga & A. Scholmerich (Eds.), *Between Culture and Biology: Perspectives on Ontogenetic Development* (pp. 303-320). Cambridge: Cambridge University Press.

Cole, M., & Derry, J. (2005). We Have Met Technology an It Is Us. In R. J. Sternberg & D. D. Preiss (Eds.), *The Impact of Tools on the Nature and Development of Human Abilities*. Mahwah, NJ: Routledge.

Cole, M., & Wertsch, J. V. (1996). Beyond the Individual-Social Antimony in Discussions of Piaget and Vygotsky. Retrieved Aug. 18, 2003, from http://www.massey.ac.nz/~alock//virtual/colevyg.htm

Colman, A. M. (Ed.) (2009) A Dictionary of Psychology. Oxford: Oxford University Press.

Creswell, J. W. (1998). *Qualitative inquiry and research design choosing among five traditions*. Thousand Oaks, Calif.: Sage.

Crichton, S., & Kinash, S. (2003). Virtual Ethnography: Interactive Interviewing Online as Method. *Canadian Journal of Learning and Technology, 29*(2). Retrieved from http://www.cjlt.ca/content/vol29.2/cjlt29-2_art-5.html

Crossley, N. (1996). *Intersubjectivity : the fabric of social becoming*. London: Sage Publications.

Crowston, K., Wei, K., Li, Q., & Howison, J. (2006). *Core and Periphery in Free/Libre and Open Source Software Team Communications*. Paper presented at the Proceedings of the 39th Annual Hawaii International Conference on System Sciences - Volume 06.

Daintith, J., & Wright, E. (2008). *A Dictionary of Computing*. Oxford: Oxford University Press.

Darvill, T. (2002). *The Concise Oxford Dictionary of Archaeology*. Oxford: Oxford University Press.

Eisenhardt, K. (1989). Building Theories From Case Study Research. *The Academy of Management Review, 14*(4), 532-550.

Engeström, Y., Puonti, A., & Seppänen, L. (2003). Spatial and Temporal Expansion of the Object as a Challenge for Reorganizing Work. In D. Nicolini, S. Gherardi & D. Yanow (Eds.), *Knowing in organizations: A practice-based approach* (pp. 151-186). Armonk: M. E. Sharpe.

Eraut, M., Steadman, S., Furner, J., Maillardet, F., Ali, A., Miller, C., et al. (2002). *The LiNEA Project - Learning During the First Three Years of Postgraduate Employment.* Paper presented at the Glasgow SRHE Conferenc, Glasgow.

Eysenbach, G., & Till, J. E. (2001). Ethical issues in qualitative research on internet communities. *British Medical Journal*. Retrieved from http://www.bmj.com/cgi/content/full/323/7321/1103#B15

Feller, J., & Fitzgerald, B. (2000). *A framework analysis of the open source software development paradigm.* Paper presented at the Proceedings of the twenty first international conference on Information systems Brisbane, Queensland, Australia.

Fielding, N. (1999). The Theoretical and Practical Applications of IT in Qualitative Analysis. In H. Millsom (Ed.), *IT in the Social Sciences. A Student's Guide to the Information and Communication Technologies.* (pp. 96-112). Malden, Massachusetts: Blackwell Publishers.

Fischer, G., Lemke, A. C., McCall, R., & Morch, A. I. (1991). Making argumentation serve design. *Hum.-Comput. Interact., 6*(3), 393-419. doi: http://dx.doi.org/10.1207/s15327051hci0603\&4_7

Freeman, L. C. (1979). Centrality in networks: Conceptual clarification. *Social Networks, 1*(3), 215–239.

Fugelli, P. (2004). *Grounding interaction in a CSCL environment*. Oslo: P. Fugelli.

Fuller, A., & Unwin, L. (2003). Fostering Workplace Learning: looking through the lens of apprenticeship. *European Educational Research Journal, 2*(1), 41-55.

Gall, M. D., Borg, W. R., & Gall, J. P. (2003). Case study research *Educational Research: an Introduction* (7th ed., pp. 123-163). Boston, Mass.: Allyn and Bacon.

Garfinkel, H. (1967). *Studies in ethnomethodology*. Englewood Cliffs, N. J.: Prentice-Hall.

Garfinkel, H. (1986). *Ethnomethodological studies of work*. London: Routledge & Kegan Paul.

Garton, L., Haythornthwaite, C., & Wellman, B. (1999). Studying on-line social networks. In S. Jones (Ed.), *Doing Internet research: Critical issues and methods for examining the Net* (pp. 75-105). London: Sage.

Geertz, C. (1973). *The interpretation of cultures*. New York: Basic Books.

George, A. L., & Bennett, A. (2005). *Case studies and theory development in the social sciences*. Cambridge, Mass.: MIT Press.

Glaser, B. G., & Strauss, A. L. (1965). *Awareness of dying*. New York: Aldine.

Glaser, B. G., & Strauss, A. L. (1968). *Time for dying*. Chicago: Aldine.

Glock, H.-J. (2003). *Quine and Davidson on language, thought and reality*. Cambridge: Cambridge University Press.

Grenfell, M. (2006). Critical Virtual Ethnography. In B. L. Mann (Ed.), *Selected styles in web-based educational research* (pp. 413-437). Hersey, PA: Information Science Pub.

Hanneman, R. A., & Riddle, M. (2005). Introduction to social network methods. Retrieved from www.faculty.ucr.edu/~hanneman/nettext/

Heritage, J. (1984). *Garfinkel and ethnomethodology*. Cambridge: Polity Press.

Hernes, T., & Czarniawska-Joerges, B. (2005). *Actor-network theory and organizing*. Malmö: Liber & Copenhagen Business School Press.

Honneth, A. (1995). *The struggle for recognition : the moral grammar of social conflicts*. Cambridge: Polity Press.

Ince, D. (2001). *A Dictionary of the Internet*. New York: Oxford University Press Inc.

Jensen, K., & Lahn, L. C. (2003). *Professional learning in a changing society. Project application.* Institute for Educational Research, University of Oslo.

Jensen, K., & Lahn, L. C. (2004). *The binding role of knowledge. An analysis of nursing students knowledge ties*. Institute for Educational Research, University of Oslo, Norway.

Jensen, K., & Lahn, L. C. (2007). Introduction. *Knowledge, work & society, 4*(3), 9-14.

Jensen, K., Nerland, M., Lahn, L. C., Fugelli, P., Smeby, J.-C., Mathisen, A., et al. (2008). Professional learning in transition. *Final report to the Research Council of Norway, March 2008*. Retrieved from http://www.pfi.uio.no/prolearn/docs/ProLearn_KUL_report_2008.pdf

Johansen, R. (1988). *Groupware - Computer Support Business Teams*. New York: The free press, Macmillan.

Kant, I., Gregor, M. J., & Sullivan, R. J. (1996). *The metaphysics of morals*. Cambridge: Cambridge University Press.

Kaptelinin, V. (2005). The Object of Activity: Making Sense of the Sense-Maker. *Mind, Culture, and Activity, 12* (1), 4-18.

Karl Franz, F. (1999). *Open Source Development with CVS*: Coriolis Group Books.

Knorr Cetina, K. (1997). Sociality with objects: social relations in postsocial knowledge societies. *Theory, culture & society, 14*(4), 1-30.

Knorr Cetina, K. (1999). *Epistemic Cultures. How the Sciences Make Knowledge.* Cambridge: MA: Harvard University Press.

Knorr Cetina, K. (2001a). Objectual practice. In T. Schatzki, K. Knorr Cetina & E. von Savigny (Eds.), *The Practice Turn in Contemporary Theory* (pp. 175-188). London: Routledge.

Knorr Cetina, K. (2001b). Postsocial Relations: Theorizing Sociality in a Postsocial Enviroment. In G. Ritzer & B. Smart (Eds.), *Handbook of social theory* (pp. 520-537). London: Sage.

Knorr Cetina, K. (2004). *Postsocial Knowledge Society. Keynote speech at the conference "Professional Learning in a Changing Society".* Oslo.

Knorr Cetina, K. (Ed.) (2005) Science, Technology, and Society. Oxford: Oxford University Press.

Knorr Cetina, K., & Brugger, U. (2002). Traders' Engagement with Markets: A Postsocial Relationship. *Theory Culture Society, 19*(5-6), 161-185.

Knorr Cetina, K., Savigny, E., & Schatzki, T. R. (2001). *The practice turn in contemporary theory (electronic resource, Oslo University library)*. New York: Routledge.

Kroeber, A. L., Kluckhohn, C., & Untereiner, W. (1952). *Culture a critical review of concepts and definitions*. Cambridge, Mass.: The Museum.

Labrinidis, A., & Roussopoulos, N. (2000). Generating dynamic content at database-backed web servers: cgi-bin vs. mod_perl. *SIGMOD Rec., 29*(1), 26-31. doi: http://doi.acm.org/10.1145/344788.344794

Lahn, L. C. (in press). Professional learning as epistemic trajectories. In S. Ludvigsen, A. Lund, I. Rasmussen & R. Säljö (Eds.), *Learning Across Sites: New tools, Infrastructures and Practices*. Oxford, U.K.: Pergamon Press.

Lakhani, K., & Wolf, R. (2005). Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects. In J. Feller (Ed.), *Perspectives on free and open source software*. Massachusetts: The MIT Press.

Lanzara, G. F., & Morner, M. (2005). Artifacts Rule! How Organising Happens in Open Software Projects. In B. Czarniawska-Joerges & T. Hernes (Eds.), *Actor-network theory and organizing* (pp. 459 - 468). Malmö: Liber & Copenhagen Business School Press.

Lapan, S. D., & DeMarrais, K. B. (2004). *Foundations for research methods of inquiry in education and the social sciences*. Mahwah, N.J.: L. Erlbaum Associates.

Lash, S. (2001). Technological Forms of Life. *Theory Culture Society, 18*(1), 105-120. doi: 10.1177/02632760122051661

Latour, B. (2005). *Reassembling the social : an introduction to actor-network-theory*. Oxford: Oxford University Press.

Lave, J., & Wenger, E. (1991). *Situated learning legitimate peripheral participation*. Cambridge: Cambridge University Press.

Law, J. (1992, 30 Nov. 2003). Notes on the Theory of the Actor-Network. Retrieved Dec. 6, 2004, from http://www.comp.lancs.ac.uk/sociology/papers/law-notes-on-ant.pdf

Law, J., & Singleton, V. (2003). Object Lessons. Retrieved Nov. 22, 2004, from http://www.comp.lancs.ac.uk/sociology/papers/Law-Singleton-Object-Lessons.pdf

Lee, G., & Cole, R. E. (2003). From a Firm-Based to a Community-Based Model of Knowledge Creation: The Case of the Linux Kernel Development. *Organization Science, 14*(6), 633-649.

Leont'ev, A. N. (1978). *Activity, Consciousness, and Personality*. Englewood Cliffs, NJ: Prentice Hall.

Lerner, R. (1999). Writing Modules for mod_perl. Retrieved Nov. 3, 2009, from http://www.linuxjournal.com/article/3351

Linell, P. (2003). Dialogical Tensions: On Rommetveitian Themes of Minds, Meanings, Monologues, and Languages. *Mind, Culture, and Activity: An International Journal, 10*(3), 219-229.

Long, Y. (2006). *Social structure, knowledge sharing, and project performance in open source software development.* University of Nebraska at Lincoln.

Long, Y., & Siau, K. (2007). Social Network Structures in Open Source Software Development Teams. *Journal of Database Management, 18*(2), 25-40.

Lowe, E. J. (2000). *The Oxford Companion to Philosophy*. Oxford: Oxford University Press.

Ludvigsen, S., Rasmussen, I., Krange, I., Moen, A., & Middleton, D. (in press). Intersecting trajectories of participation; temporality and learning In S. Ludvigsen, A. Lund, I. Rasmussen & R. Säljö (Eds.), *Learning across sites: New tools, infrastructures and practices*: Pergamon Press.

Luria, A. R. (1979). *The Making of Mind*. Cambridge: Harvard University Press.

Mann, C., & Stewart, F. (2000). *Internet communication and qualitative research : a handbook for researching online*. London: Sage.

Marshall, G. (1998a). *A Dictionary of Sociology*: Oxford University Press.

Marshall, G. (1998b). A Dictionary of Sociology. Oxford Reference Online.Retrieved 3 March, 2003, from http://www.oxfordreference.com/views/ENTRY.html?subview=Main&entry=t88.000761

Matusov, E. (1996). Intersubjectivity Without Agreement. *Mind, Culture, and Activity, 3*(1), 25-45.

Matusov, E. (2001). Intersubjectivity as a way of informing teaching design for a community of learners classroom. *Teaching and Teacher Education, 17*(4), 383-402

Mead, G. H. (1962). *Mind, Self & Society from the standpoint of a social behaviorist*. Chicago: The University of Chicago Press.

Merleau-Ponty, M., & Smith, C. (2002). *Phenomenology of perception (electronic resource, Oslo University library)*. London New York: Routledge.

Miettinen, R. (2006). The sources of novelty: a cultural and systemic view of distributed creativity. *Creativity and Innovation Management, 15*(2), 173-181.

Miettinen, R., & Virkkunen, J. (2005). Epistemic Objects, Artefacts and Organizational Change. *Organization, 12*(3), 437–456.

Miller, D. C., & Salkind, N. J. (2002). *Handbook of research design & social measurement* (6th ed.). Thousand Oaks, Calif.: Sage.

Moglen, E. (1999). Anarchism Triumphant: Free Software and the Death of Copyright. *First Monday*. Retrieved from http://emoglen.law.columbia.edu/my_pubs/anarchism.html

Nardi, B. (1996). Studying context : a comparison of activity theory, situated action models and distributed cognition. In B. Nardi (Ed.), *Context and Consciousness: Activity Theory and Human-Computer Interaction* (pp. 69-102). Cambridge, Massachusetts: The MIT Press.

Nardi, B. (2005). Objects of Desire: Power and Passion in Collaborative Activity. *Mind, Culture, and Activity, 12* (1), 37-51.

Nerland, M. (2007). Knowledge Cultures and the Shaping of Work-based Learning: The Case of Computer Engineering. *Vocations and Learning: Studies in vocational and professional education, 1*(1), 49-69.

Nespor, J. (1994). Knowledge in motion : space, time, and curriculum in undergraduate physics and management. *Knowledge, identity and school life series*. Retrieved Dec. 20, 2004, from http://filebox.vt.edu/users/nespor/kim.html

News.netcraft.com. (2006). Web Server Survey. Retrieved 03. okt, 2006, from http://news.netcraft.com/archives/web_server_survey.html

Nova, N., Sangin, M., & Dillenbourg, P. (2008). *Reconsidering Clark's Theory in CSCW*. Paper presented at the 8th International Conference on the Design of Cooperative Systems (COOP'08), Carry-le-Rouet. http://infoscience.epfl.ch/record/133143

Packer, M. J., & Goicoechea, J. (2000). Sociocultural and Constructivist Theories of Learning: Ontology, Not Just Epistemology. *Educational Psychologist, 35*(4), 227–241.

Patton, M. Q. (2002). *Qualitative research & evaluation methods*. Thousand Oaks, Calif.: Sage Publications.

Perens, B. (1999). The Open Source Definition. In C. Dibona, S. Ockman & M. Stone (Eds.), *Open Sources: Voices from the Open Source Revolution* (pp. 171–188). Sebastopol, CA: O'Reilly.

Petroski, H. (1992). *The evolution of useful things*. New York: Knopf.

Petroski, H. (2006). *Success through failure : The paradox of design*. Princeton, N.J.: Princeton University Press.

Pfohl, S. J. (1994). *Images of deviance and social control a sociological history* (2nd ed.). New York: McGraw-Hill.

Phillips, D. C. (1992). *The social scientist's bestiary : a guide to fabled threats to, and defenses of, naturalistic social science* (1st ed.). Oxford ; New York: Pergamon Press.

prolepsis. (2009). In Merriam-Webster Online Dictionary. Retrieved from http://www.merriam-webster.com/dictionary/prolepsis

Rheinberger, H.-J. (1997). *Toward a history of epistemic things : synthesizing proteins in the test tube*. Stanford, Calif.: Stanford University Press.

Rheingold, H. (1998). *The virtual community : homesteading on the electronic frontier (electronic resource, Oslo University library)* (2nd ed.). [S.l.]: Howard Rheingold.

Richards, L. (2005). *Handling qualitative data a practical guide*. London: Sage.

Rommetveit, R. (1974). *On message structure : a framework for the study of language and communication*. London: Wiley.

Rommetveit, R. (1978). On negative rationalism in scholarly studies of verbal communication and dynamic residuals in the construction of human intersubjectivity. In M. Brenner, P. Marsh & M. Brenner (Eds.), *The Social contexts of method* (pp. 16-32). London: Croom Helm.

Rommetveit, R. (1979a). On the architecture of intersubjectivity. In R. Rommetveit & R. M. Blakar (Eds.), *Studies of Language, Thought and Verbal Communication* (pp. 147-161). New York: Academic Press.
Rommetveit, R. (1979b). *The role of language in the creation and transmission of social representations*. manuscript. University of Oslo. Oslo.
Rommetveit, R. (1985). Language acquisition as increasing linguistic structuring of experience and symbolic behavior control. In J. V. Wertsch (Ed.), *Culture, communication and cognition: Vygotskian Perspectives*. Cambridge: Cambridge University Press.
Rommetveit, R. (1992). Outlines of a Dialogically Based Social-Cognitive Approach to Human Cognition and Communication. In A. H. Wold (Ed.), *The dialogical alternative* (pp. 19-43). Oslo: Scandinavian University Press.
Rommetveit, R. (2003). On the Role of "a Psychology of the Second Person"in Studies of Meaning, Language, and Mind. *Mind, Culture, and Activity, 10*(3), 205-218. doi: doi:10.1207/s15327884mca1003_3
Rommetveit, R., & Blakar, R. M. (1979). *Studies of language, thought and verbal communication*. London: Academic Press.
Roulston, K. J. (2003). Ethnomethodological and Conversation Analytic Studies. In K. deMarrais & S. D. Lapan (Eds.), *Foundations for Research. Methods of Inquiry in Education and the Social Sciences* (pp. 139-160). Mahwah, New Jersey: Lawrence Erlbaum.
Russell, M. (2006). *Husserl : a guide for the perplexed*. London Continuum ,.
Schatzki, T. R. (2001). Introduction - Practice theory. In K. Knorr-Cetina, E. v. Savigny & T. R. Schatzki (Eds.), *The practice turn in contemporary theory* (pp. 10-23). New York: Routledge.
Scott, J. (2000). *Social network analysis a handbook* (2nd ed.). London: Sage.
Scott, J., & Marshall, G. (2005). "cyber ethnography" A Dictionary of Sociology. Retrieved from http://www.oxfordreference.com/views/ENTRY.html?subview=Main&entry=t88.e496
Scott, J., & Marshall, G. (2005). A Dictionary of Sociology. *Oxford Reference Online*. Retrieved 10. Oct, 2006, from http://www.oxfordreference.com/views/ENTRY.html?subview=Main&entry=t88.e1825
Seidman, B. (1983). Network structure and minimum degree. *Social Networks, 5*, 269–287.
Seidman, B., & Foster, L. (1978). A graph theoretic generalization of the clique concept. *Journal of Mathematical Sociology, 6*, 139–154.
Silverman, D. (2001). *Interpreting qualitative data : methods for analysing talk, text and interaction* (2nd ed.). London: Sage.
Silverman, D. (2010). *Doing Qualitative Research* (3 ed.). London: SAGE Publications.
Singleton, V., & Law, J. (2003). Allegory and Its Others. In D. Nicolini, S. Gherardi & Y. Dvora (Eds.), *Knowing in Organizations : A Practice - Based Approach* (pp. 225-254). New York: M.E. Sharpe.
Soanes, C., & Stevenson, A. (Eds.). (2005). *The Oxford Dictionary of English* (2nd ed.). Oxford: Oxford University Press.
Soanes, C., & Stevenson, A. (Eds.). (2006). *The Concise Oxford English Dictionary*. Oxford: Oxford University Press.
Spainhour, S., & Eckstein, R. (2002). *WebMaster in a Nutshell*: O'Reilly & Associates, Inc.
Stahl, G. (2002, January 7-11). *Contributions to a Theoretical Framework for CSCL.* Paper presented at the Proceedings of: CSCL 2002, Boulder, Colorado, USA.
Stake, R. E. (1995). *The art of case study research*. Thousand Oaks, Calif.: Sage.

Stake, R. E. (2005). Qualitative case studies. In N. K. Denzin & Y. S. Lincoln (Eds.), *The Sage handbook of qualitative research* (pp. 443-466). Thousand Oaks, Calif.: Sage.

Steadman, S. (2005, April 15th 2005). *Early Career Learning at Work (LiNEA) Project - Methodology and Theoretical Frameworks.* Paper presented at the Symposium on Early Career Professional Learning - AERA Conference, Montreal.

Strauss, A. L., & Corbin, J. M. (1998). *Basics of qualitative research techniques and procedures for developing grounded theory* (2nd ed.). Thousand Oaks, Calif.: Sage Publications.

Stutt, A., & Motta, E. (1998). Knowledge modelling: An organic technology for the knowledge age. In M. Eisenstadt & T. Vincent (Eds.), *The knowledge web: Learning and collaboration on the net*. London: Kogan Page.

Säljö, R. (2006). *Læring og kulturelle redskaper : om læreprosesser og den kollektive hukommelsen*. Oslo: Cappelen akademisk forl.

Teddlie, C., & Tashakkori, A. (2003). *Handbook of mixed methods in social & behavioral research*. Thousand Oaks, Calif.: SAGE.

Traum, D. (1999). *Computational Models of Grounding in Collaborative Systems.* Paper presented at the Proceedings of AAAI Fall Symposium on Psychological Models of Communication.

Van Lier, L. (2004). *The ecology and semiotics of language learning : a sociocultural perspective*. Boston: Kluwer Academic.

Von Hippel, E. (2005). *Democratizing Innovation*. Cambridge: The MIT Press.

Vygotsky, L. S. (1962). *Thought and language*. Cambridge: MIT Press.

Vygotsky, L. S. (1978). *Mind in society: The development of higher psychological processes.* Cambridge: Harward University Press.

Wartofsky, M. W. (1979). *Models representation and the scientific understanding*. Dordrecht: D. Reidel Pub. Co.

Wasserman, S., & Faust, K. (1994). *Social network analysis methods and applications*. Cambridge: Cambridge University Press.

Watson, G., & Goulet, J.-G. (1998). What Can Ethnomethodology Say About Power? *Qualitative Inquiry, 4*(1), 96-113. doi: 10.1177/107780049800400106

Weber, S. (2004). *The success of open source*. Cambridge, Mass.: Harvard University Press.

Wertsch, J. V. (1996). The need for action in sosiocultural research. In B. A. Nardi (Ed.), *Context and Consciousness: Activity Theory and Human-Computer Interaction*. Cambridge, Massachusetts: The MIT Press.

Willging, P. (2005). Using Social Network Analysis Techniques to Examine Online Interactions. *US-China Education Review, 2*(9), 46-56.

Wittel, A. (2000). Ethnography on the Move: From Field to Net to Internet. *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research, 1*(1). Retrieved from http://www.qualitative-research.net/fqs-texte/1-00/1-00wittel-e.htm

Wittgenstein, L. (1968). *Philosophical Investigations*. Oxford: Blackwell.

Wittgenstein, L. (1969). *Preliminary studies for the "Philosophical Investigations" generally known as The blue and brown books* (2nd ed.). Oxford: Blackwell.

Wold, A. H. (1992). *The Dialogical alternative towards a theory of language and mind*. Oslo: Scandinavian University Press.

Yin, R. K. (1994). *Case study research : design and methods* (2nd ed.). Thousand Oaks, Calif.: Sage.

Yin, R. K. (1997). The Abridged Version of Case Study Research. Desing and Method. In D. J. Rog & L. Bickman (Eds.), *Handbook of applied social research methods* (pp. 229-260). Thousand Oaks, Cal.: Sage.

Yin, R. K. (2008). *Case study research : design and methods*. London: Sage.

9 APPENDICES

9.1 Appendix: Personal interview guide

Introduction:

The project's main focus is on knowledge development and learning. What enables learning and how do professionals relate to new knowledge? We partly want to know about simple facts related, for example, to what you read, what kind of courses and training you are offered as well as questions related to how you experience today's knowledge and work demands. Prior to this, it is desirable to get to know a little more about your educational background and with what you are currently occupied at work.

Questions:

- How did you get to where you are now — the vast lines? Why did you enter this educational path?
- Can you see a *thread* driving you in choosing your education and profession, or have your preferences changed a great deal along the way?
- How did you experience the transition from education to work life?
- What do you currently do in your work? (Please mention most central tasks)
- Who do you mostly collaborate with? What is it that you collaborate on and how?
- Which tasks are the most interesting to work on? What is so exciting about them?
- Is there something that gives you a real 'rush'? Anything that you find really boring? Can you please tell me more about that?
- Could you please give examples of especially challenging tasks or situations where you feel insufficient? How do you handle such tasks? What resources are applied (e.g., reference books, academic books, advice from colleagues, etc.).
- Do you get to use the knowledge acquired from education in these situations? If not, why is that? Do you need a different kind of knowledge? What have you done to obtain this? Is there anything you would have more or less of during education?
- Where are your educational books located now? Do you ever consult these? Why/why not?
- What are the three most important matters you bring with you from education?

- Are there any occupation-specific challenges related to your work that you frequently return to and wish to know more about?
- The initiative to search new knowledge may come from several sources, for example; you can take the initiative or your employer can demand it. To what extent will you describe yourself as active in this regard? Can you please say a little more about what drives you to seek new knowledge? When do you take the initiative?
- What do you read in relation to your work (content, medium, genre, type of knowledge)? When you read this stuff, do you feel like reading more? Do you apply it in your work? How? Is it something you talk to your colleagues about? Do you read anything occupational outside work? Are you inspired by any other sources?
- Are there any 'knowledge combats' in your field (e.g., defined groups with different perspectives on the field)? If yes, how is this expressed at your workplace / in your professional field? Are there any antagonisms in your field between those who are more theoretically oriented and those who are more practical or skill oriented? Have you decided on such differences in knowledge orientation?
- At a workplace there might also exist different opinions on what it means to produce high quality work. When do you feel that you have produced high-quality work? Can you please give an example of such a situation? What do you do to quality assure your own work?
- Do colleagues comment on how you perform your work? If you get feedback – positive or negative – how do you relate to this (i.e., follow up, ignore or protest)?
- Is there something about your work that you feel a special responsibility for, and that will bother you if it fails? What do you do to cope with this (survival strategies, advice from colleagues, techniques for ignoring the problem, etc.)?
- Are there any qualities or skills you especially wish to develop – what would you like to be better at? What do you currently do in order to develop such qualities/skills?
- We spoke earlier about how you proceed when facing concrete problems or challenges in your work. When it comes to more general developments in the field, how do you keep abreast? How do you prefer to keep updated on these matters?
    - Through practical participation in new tasks
    - Through courses
    - By following journals, vocational literature or other distributed sources (internet)

  - Self-initiated updating or organised by others; which do you prefer?
- Within your field, do you think that the knowledge frontline (the prevailing knowledge and methods) will have changed within e.g., five years? What consequences would this imply for you as a professional practitioner? Do you find this exciting to think about, or does it frighten you?
- During a busy work day it can often be difficult to find time and energy for renewal. Is this a problem for you? What can be done to remedy the situation? Where do you think the responsibility for keeping abreast of new knowledge in the field is situated – with the employer, you or the profession as a whole?
- Are you tired of all this talk about updating, or do you consider it an enjoyable challenge?
- Are you proud of being a computer engineer? What is it that makes you proud? In your experience, has your profession had an increased or decreased status in the last few years? What is this a result of? What can to be done to improve this?
- Is there anything that we have not talked about that you wish to add?

9.2 Appendix: Learning log

Notes on translation

The original learning log forms were written in Norwegian. The following is an English translation of the form and the information provided for participants.

Introductory message to log writer: This learning log is a part of the ProLearn project and focuses on knowledge development and learning. The purpose is to collect information about how people in your vocational group relate to knowledge and learning in their daily work.

The log is to be filled out at the end of four optional work days within the two-week registration period. In this logbook, you will find a form for each of the four days, marked with "Day 1-4". Please consider the professional challenges you have met during the work day and how these were handled. Spend about 10 minutes writing down a few notes/keywords.

Log day form fields' prompts and explanations:

- Field 1: Technical/professional questions you have wondered about today
- Field 2: What triggered the questions/curiosity?
- Field 3: Knowledge sources applied in order to get answers or find out more about it (e.g., books, internet resources, colleagues and the like).
- Field 4: Elaborative comments. If you do have more comments about how you handled today's professional challenges and own learning, please write additional notes here.

Overall comment (last registration day)

When you look back on the two week period of learning log registration, were these typical work weeks for you? If not, in what way was this period atypical (i.e., great workload, little to do, engagement in exception projects, etc.)?

9.3 Appendix: Focus group interview guide

Focus 1: Knowledge profile

We have invited you all here today because you represent a common professional group; namely, computer engineers. Some of the issues that we are interested in knowing more about are: to what degree can your professional group be identified with a particular knowledge profile? Is it reasonable to say that you have your own type of knowledge? Do you know something that others do not know? Which occupational groups can also perform your kind of work? What distinguishes this/these person's/persons' professional practice from yours?

Focus 2: Dynamics of the knowledge field

In our material, we find an interesting gap in how the different groups participating in the ProLearn project view their professional situation in a future perspective. Some foresee great changes; for instance, a computer engineer says that "if you are gone from this profession in two years, then you're out of the game". Other groups do not foresee such substantial changes in their profession. How is this for you guys? What are the changes all about? How do you get to know about them?

Focus 3: Updating strategies

All the participating groups experience their work as rich in learning possibilities. Knowledge developments in the different fields still require different demands for updating, and the updating occurs in different ways. It can be conducted in a systematic or random manner. Some think long-term, whereas others are more concerned about solving problems there and then. Some get an organised training scheme, whereas others have to take the initiative themselves. How would you place yourself in this landscape?

Follow-up question: Discuss the pros and cons of these different strategies. (Do you think there should be more clearly defined updating requirements? More flexibility?)

Updating strategies in relation to knowledge traditions

Updating strategies is also characterised by the fact that the groups have different knowledge traditions. Some are largely based on an oral culture; others have richer access to written sources. Some are predominantly occupied with definite knowledge that is directly applicable

to practical problems; others are provided with a more abstract or ‘superstructured’ knowledge base. Some communities are specifically locally anchored; others may have an international character. How would you situate your professional group in this picture?

Problematise consequences of knowledge traditions for updating strategies: Risking local nearsightedness, too little context adaptation, too much focus on rules and lack of creativity?

Focus 5: Motivation / drive to further learning

There also appear to be differences in the material when it comes to what motivates people to learn more and how such an urge is projected onto the demands and environment of which one is a part. What gives you an urge to learn more? What types of knowledge trigger your curiosity?

Some report that they are satisfied with the current ‘learning pressure’, but nevertheless do not think they will manage it in the long term: the ‘learning pressure’ may be interesting now, but it is not something they want carry on with in a long-term professional life. What do you think about this?

The motivation related to variation in work tasks and the use of own competencies: In which situations do you experience variations that you find challenging and fun; something that creates the urge to learn more? On the other hand, what will suppress the urge?

9.4 Appendix: The Open-source Definition 1.9

The Open Source Definition

Introduction

Open source doesn't just mean access to the source code. The distribution terms of open-source software must comply with the following criteria:

1. Free Redistribution

The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.

2. Source Code

The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost preferably, downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.

3. Derived Works

The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.

4. Integrity of The Author's Source Code

The license may restrict source-code from being distributed in modified form *only* if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time. The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.

5. No Discrimination Against Persons or Groups

The license must not discriminate against any person or group of persons.

6. No Discrimination Against Fields of Endeavor

The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research.

7. Distribution of License

The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.

8. License Must Not Be Specific to a Product

The rights attached to the program must not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.

9. License Must Not Restrict Other Software

The license must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.

10. License Must Be Technology-Neutral

No provision of the license may be predicated on any individual technology or style of interface.

The Open Source Definition

9.5 Appendix: Freeman's degree centrality measures

---

		1 OutDegree	2 InDegree	3 NrmOutDeg	4 NrmInDeg
12	dev-12	62.000	40.000	28.972	18.692
29	dev-29	29.000	19.000	13.551	8.879
1	dev-1	25.000	21.000	11.682	9.813
81	dev-81	24.000	23.000	11.215	10.748
23	dev-23	22.000	12.000	10.280	5.607
16	dev-16	18.000	12.000	8.411	5.607
3	dev-3	17.000	16.000	7.944	7.477
14	dev-14	16.000	11.000	7.477	5.140
28	dev-28	13.000	14.000	6.075	6.542
52	dev-52	11.000	4.000	5.140	1.869
40	dev-40	11.000	12.000	5.140	5.607
144	dev-144	9.000	11.000	4.206	5.140
22	dev-22	9.000	5.000	4.206	2.336
39	dev-39	8.000	8.000	3.738	3.738
49	dev-49	8.000	7.000	3.738	3.271
20	dev-20	7.000	6.000	3.271	2.804
43	dev-43	7.000	10.000	3.271	4.673
65	dev-65	7.000	6.000	3.271	2.804
9	dev-9	7.000	12.000	3.271	5.607
153	dev-153	7.000	8.000	3.271	3.738
80	dev-80	7.000	5.000	3.271	2.336
125	dev-125	7.000	6.000	3.271	2.804
31	dev-31	6.000	5.000	2.804	2.336
64	dev-64	6.000	6.000	2.804	2.804
61	dev-61	6.000	2.000	2.804	0.935
46	dev-46	5.000	4.000	2.336	1.869
78	dev-78	4.000	6.000	1.869	2.804
116	dev-116	4.000	0.000	1.869	0.000
177	dev-177	4.000	4.000	1.869	1.869
85	dev-85	4.000	6.000	1.869	2.804
89	dev-89	4.000	3.000	1.869	1.402
53	dev-53	4.000	4.000	1.869	1.869
41	dev-41	4.000	2.000	1.869	0.935
161	dev-161	4.000	3.000	1.869	1.402
63	dev-63	4.000	4.000	1.869	1.869
94	dev-94	4.000	4.000	1.869	1.869
21	dev-21	4.000	4.000	1.869	1.869
186	dev-186	4.000	4.000	1.869	1.869
59	dev-59	3.000	3.000	1.402	1.402
38	dev-38	3.000	5.000	1.402	2.336
166	dev-166	3.000	3.000	1.402	1.402
115	dev-115	3.000	4.000	1.402	1.869
110	dev-110	3.000	4.000	1.402	1.869
150	dev-150	3.000	1.000	1.402	0.467

104	dev-104	3.000	3.000	1.402	1.402
19	dev-19	3.000	3.000	1.402	1.402
99	dev-99	3.000	2.000	1.402	0.935
145	dev-145	3.000	1.000	1.402	0.467
58	dev-58	3.000	2.000	1.402	0.935
182	dev-182	3.000	7.000	1.402	3.271
96	dev-96	3.000	3.000	1.402	1.402
90	dev-90	3.000	3.000	1.402	1.402
66	dev-66	3.000	2.000	1.402	0.935
91	dev-91	3.000	3.000	1.402	1.402
56	dev-56	3.000	2.000	1.402	0.935
148	dev-148	3.000	0.000	1.402	0.000
26	dev-26	3.000	0.000	1.402	0.000
45	dev-45	2.000	2.000	0.935	0.935
15	dev-15	2.000	2.000	0.935	0.935
88	dev-88	2.000	6.000	0.935	2.804
33	dev-33	2.000	2.000	0.935	0.935
131	dev-131	2.000	2.000	0.935	0.935
50	dev-50	2.000	0.000	0.935	0.000
57	dev-57	2.000	2.000	0.935	0.935
113	dev-113	2.000	1.000	0.935	0.467
55	dev-55	2.000	3.000	0.935	1.402
93	dev-93	2.000	2.000	0.935	0.935
30	dev-30	2.000	2.000	0.935	0.935
69	dev-69	2.000	1.000	0.935	0.467
111	dev-111	2.000	0.000	0.935	0.000
97	dev-97	2.000	0.000	0.935	0.000
141	dev-141	2.000	6.000	0.935	2.804
138	dev-138	2.000	2.000	0.935	0.935
27	dev-27	2.000	3.000	0.935	1.402
62	dev-62	2.000	2.000	0.935	0.935
76	dev-76	2.000	2.000	0.935	0.935
118	dev-118	2.000	3.000	0.935	1.402
139	dev-139	2.000	1.000	0.935	0.467
163	dev-163	2.000	6.000	0.935	2.804
172	dev-172	2.000	2.000	0.935	0.935
133	dev-133	2.000	2.000	0.935	0.935
175	dev-175	2.000	0.000	0.935	0.000
136	dev-136	2.000	3.000	0.935	1.402
171	dev-171	2.000	2.000	0.935	0.935
51	dev-51	2.000	2.000	0.935	0.935
102	dev-102	2.000	3.000	0.935	1.402
197	dev-197	2.000	2.000	0.935	0.935
146	dev-146	2.000	2.000	0.935	0.935
5	dev-5	1.000	2.000	0.467	0.935
184	dev-184	1.000	2.000	0.467	0.935
207	dev-207	1.000	0.000	0.467	0.000
74	dev-74	1.000	1.000	0.467	0.467
87	dev-87	1.000	1.000	0.467	0.467
188	dev-188	1.000	0.000	0.467	0.000
114	dev-114	1.000	1.000	0.467	0.467
17	dev-17	1.000	1.000	0.467	0.467
178	dev-178	1.000	2.000	0.467	0.935
18	dev-18	1.000	0.000	0.467	0.000
67	dev-67	1.000	2.000	0.467	0.935
34	dev-34	1.000	2.000	0.467	0.935

42	dev-42	1.000	0.000	0.467	0.000
36	dev-36	1.000	0.000	0.467	0.000
103	dev-103	1.000	1.000	0.467	0.467
132	dev-132	1.000	0.000	0.467	0.000
126	dev-126	1.000	0.000	0.467	0.000
47	dev-47	1.000	1.000	0.467	0.467
100	dev-100	1.000	0.000	0.467	0.000
187	dev-187	1.000	1.000	0.467	0.467
162	dev-162	1.000	0.000	0.467	0.000
137	dev-137	1.000	1.000	0.467	0.467
98	dev-98	1.000	1.000	0.467	0.467
106	dev-106	1.000	1.000	0.467	0.467
60	dev-60	1.000	0.000	0.467	0.000
6	dev-6	1.000	0.000	0.467	0.000
8	dev-8	1.000	1.000	0.467	0.467
117	dev-117	1.000	1.000	0.467	0.467
169	dev-169	1.000	5.000	0.467	2.336
32	dev-32	1.000	1.000	0.467	0.467
119	dev-119	1.000	2.000	0.467	0.935
2	dev-2	1.000	1.000	0.467	0.467
121	dev-121	1.000	2.000	0.467	0.935
122	dev-122	1.000	0.000	0.467	0.000
149	dev-149	1.000	1.000	0.467	0.467
151	dev-151	1.000	3.000	0.467	1.402
204	dev-204	1.000	1.000	0.467	0.467
152	dev-152	1.000	1.000	0.467	0.467
214	dev-214	1.000	1.000	0.467	0.467
128	dev-128	1.000	0.000	0.467	0.000
75	dev-75	1.000	3.000	0.467	1.402
130	dev-130	1.000	3.000	0.467	1.402
105	dev-105	1.000	1.000	0.467	0.467
158	dev-158	1.000	1.000	0.467	0.467
92	dev-92	1.000	5.000	0.467	2.336
147	dev-147	1.000	2.000	0.467	0.935
201	dev-201	1.000	0.000	0.467	0.000
190	dev-190	1.000	1.000	0.467	0.467
4	dev-4	1.000	2.000	0.467	0.935
203	dev-203	1.000	1.000	0.467	0.467
183	dev-183	1.000	0.000	0.467	0.000
173	dev-173	1.000	0.000	0.467	0.000
194	dev-194	1.000	1.000	0.467	0.467
195	dev-195	1.000	1.000	0.467	0.467
209	dev-209	1.000	2.000	0.467	0.935
24	dev-24	1.000	1.000	0.467	0.467
174	dev-174	1.000	0.000	0.467	0.000
11	dev-11	1.000	1.000	0.467	0.467
212	dev-212	1.000	0.000	0.467	0.000
206	dev-206	1.000	1.000	0.467	0.467
68	dev-68	1.000	0.000	0.467	0.000
198	dev-198	1.000	1.000	0.467	0.467
211	dev-211	1.000	1.000	0.467	0.467
124	dev-124	0.000	2.000	0.000	0.935
108	dev-108	0.000	1.000	0.000	0.467
48	dev-48	0.000	3.000	0.000	1.402
7	dev-7	0.000	1.000	0.000	0.467
10	dev-10	0.000	0.000	0.000	0.000

86	dev-86	0.000	0.000	0.000	0.000
73	dev-73	0.000	2.000	0.000	0.935
127	dev-127	0.000	1.000	0.000	0.467
112	dev-112	0.000	4.000	0.000	1.869
155	dev-155	0.000	0.000	0.000	0.000
109	dev-109	0.000	2.000	0.000	0.935
84	dev-84	0.000	1.000	0.000	0.467
164	dev-164	0.000	0.000	0.000	0.000
165	dev-165	0.000	2.000	0.000	0.935
160	dev-160	0.000	1.000	0.000	0.467
82	dev-82	0.000	0.000	0.000	0.000
168	dev-168	0.000	1.000	0.000	0.467
143	dev-143	0.000	1.000	0.000	0.467
170	dev-170	0.000	5.000	0.000	2.336
25	dev-25	0.000	2.000	0.000	0.935
140	dev-140	0.000	1.000	0.000	0.467
167	dev-167	0.000	0.000	0.000	0.000
54	dev-54	0.000	1.000	0.000	0.467
123	dev-123	0.000	0.000	0.000	0.000
176	dev-176	0.000	2.000	0.000	0.935
72	dev-72	0.000	0.000	0.000	0.000
71	dev-71	0.000	0.000	0.000	0.000
120	dev-120	0.000	2.000	0.000	0.935
101	dev-101	0.000	1.000	0.000	0.467
181	dev-181	0.000	2.000	0.000	0.935
129	dev-129	0.000	0.000	0.000	0.000
157	dev-157	0.000	1.000	0.000	0.467
77	dev-77	0.000	2.000	0.000	0.935
179	dev-179	0.000	2.000	0.000	0.935
79	dev-79	0.000	0.000	0.000	0.000
134	dev-134	0.000	4.000	0.000	1.869
135	dev-135	0.000	0.000	0.000	0.000
189	dev-189	0.000	1.000	0.000	0.467
83	dev-83	0.000	0.000	0.000	0.000
191	dev-191	0.000	2.000	0.000	0.935
192	dev-192	0.000	0.000	0.000	0.000
193	dev-193	0.000	0.000	0.000	0.000
35	dev-35	0.000	1.000	0.000	0.467
142	dev-142	0.000	1.000	0.000	0.467
196	dev-196	0.000	0.000	0.000	0.000
37	dev-37	0.000	0.000	0.000	0.000
13	dev-13	0.000	1.000	0.000	0.467
199	dev-199	0.000	1.000	0.000	0.467
200	dev-200	0.000	3.000	0.000	1.402
95	dev-95	0.000	1.000	0.000	0.467
202	dev-202	0.000	1.000	0.000	0.467
70	dev-70	0.000	1.000	0.000	0.467
44	dev-44	0.000	0.000	0.000	0.000
205	dev-205	0.000	1.000	0.000	0.467
180	dev-180	0.000	0.000	0.000	0.000
154	dev-154	0.000	0.000	0.000	0.000
208	dev-208	0.000	2.000	0.000	0.935
156	dev-156	0.000	0.000	0.000	0.000
210	dev-210	0.000	1.000	0.000	0.467
185	dev-185	0.000	2.000	0.000	0.935
159	dev-159	0.000	0.000	0.000	0.000

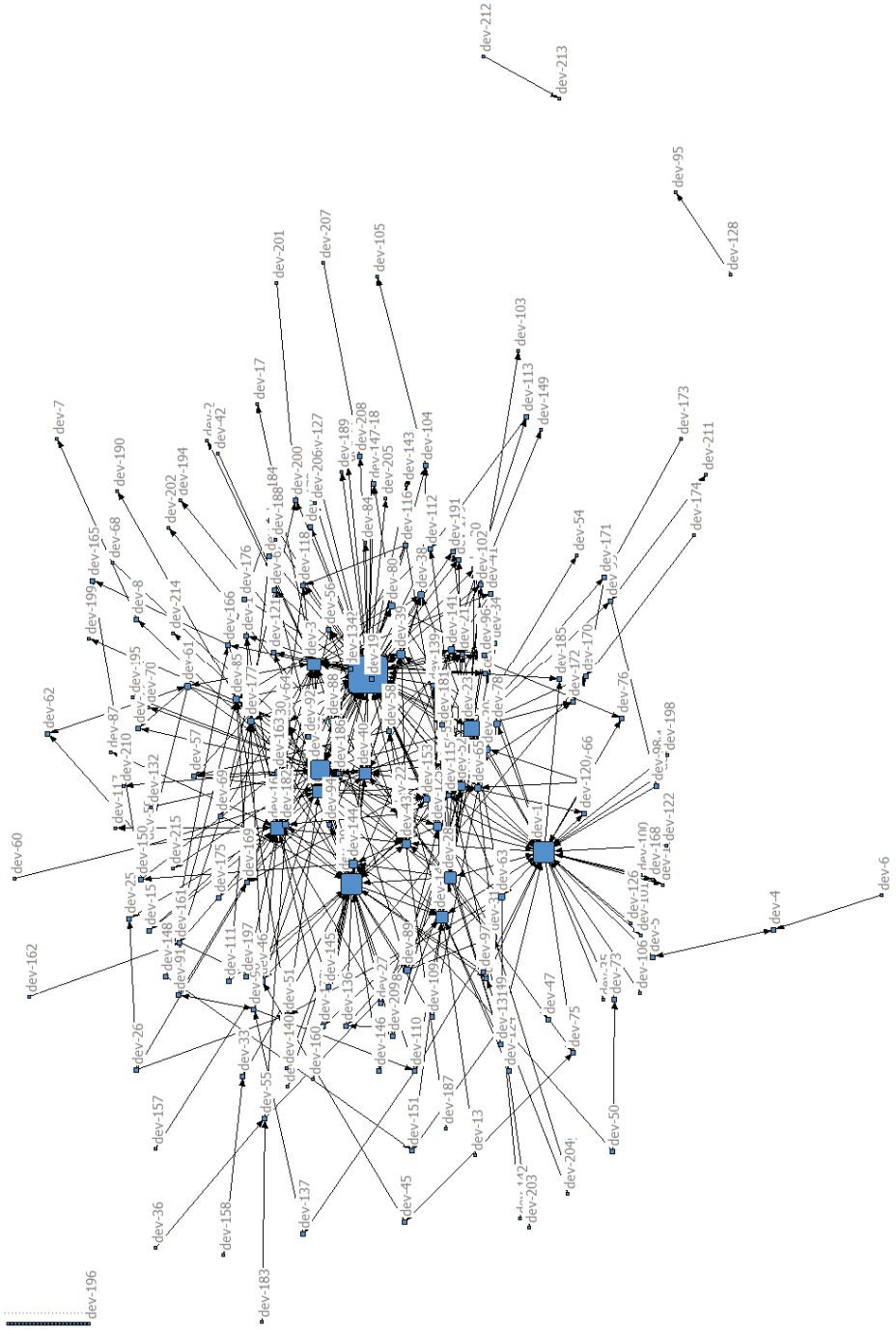
213	dev-213	0.000	1.000	0.000	0.467
107	dev-107	0.000	0.000	0.000	0.000
215	dev-215	0.000	1.000	0.000	0.467

DESCRIPTIVE STATISTICS

		1 OutDegree	2 InDegree	3 NrmOutDeg	4 NrmInDeg
1	Mean	2.716	2.716	1.269	1.269
2	Std Dev	5.859	4.335	2.738	2.026
3	Sum	584.000	584.000	272.897	272.897
4	Variance	34.324	18.789	7.495	4.103
5	SSQ	8966.000	5626.000	1957.813	1228.492
6	MCSSQ	7379.693	4039.693	1611.427	882.106
7	Euc Norm	94.689	75.007	44.247	35.050
8	Minimum	0.000	0.000	0.000	0.000
9	Maximum	62.000	40.000	28.972	18.692

Network Centralisation (OutDegree) = 27.832%
Network Centralisation (InDegree) = 17.504%

9.6 Appendix: Overall network structure



9.7 Appendix: Simple core-periphery model

```
Type of data:                 Positive
Fitness measure:              CORR
Density of core-to-periphery ties:
Number of iterations:         50
Population size:              30

Starting fitness: 0.334
Final fitness: 0.384

Core/Periphery Class Memberships:

    1:  dev-1 dev-3 dev-9 dev-12 dev-14 dev-16 dev-20 dev-22 dev-23
dev-28 dev-29 dev-39 dev-40 dev-43 dev-52 dev-64 dev-65 dev-78 dev-
80 dev-81 dev-125 dev-144 dev-153

    2:  dev-2 dev-4 dev-5 dev-6 dev-7 dev-8 dev-10 dev-11 dev-13
dev-15 dev-17 dev-18 dev-19 dev-21 dev-24 dev-25 dev-26 dev-27 dev-
30 dev-31 dev-32 dev-33 dev-34 dev-35 dev-36 dev-37 dev-38 dev-41
dev-42 dev-44 dev-45 dev-46 dev-47 dev-48 dev-49 dev-50 dev-51 dev-
53 dev-54 dev-55 dev-56 dev-57 dev-58 dev-59 dev-60 dev-61 dev-62
dev-63 dev-66 dev-67 dev-68 dev-69 dev-70 dev-71 dev-72 dev-73 dev-
74 dev-75 dev-76 dev-77 dev-79 dev-82 dev-83 dev-84 dev-85 dev-86
dev-87 dev-88 dev-89 dev-90 dev-91 dev-92 dev-93 dev-94 dev-95 dev-
96 dev-97 dev-98 dev-99 dev-100 dev-101 dev-102 dev-103 dev-104 dev-
105 dev-106 dev-107 dev-108 dev-109 dev-110 dev-111 dev-112 dev-113
dev-114 dev-115 dev-116 dev-117 dev-118 dev-119 dev-120 dev-121 dev-
122 dev-123 dev-124 dev-126 dev-127 dev-128 dev-129 dev-130 dev-131
dev-132 dev-133 dev-134 dev-135 dev-136 dev-137 dev-138 dev-139 dev-
140 dev-141 dev-142 dev-143 dev-145 dev-146 dev-147 dev-148 dev-149
dev-150 dev-151 dev-152 dev-154 dev-155 dev-156 dev-157 dev-158 dev-
159 dev-160 dev-161 dev-162 dev-163 dev-164 dev-165 dev-166 dev-167
dev-168 dev-169 dev-170 dev-171 dev-172 dev-173 dev-174 dev-175 dev-
176 dev-177 dev-178 dev-179 dev-180 dev-181 dev-182 dev-183 dev-184
dev-185 dev-186 dev-187 dev-188 dev-189 dev-190 dev-191 dev-192 dev-
193 dev-194 dev-195 dev-196 dev-197 dev-198 dev-199 dev-200 dev-201
dev-202 dev-203 dev-204 dev-205 dev-206 dev-207 dev-208 dev-209 dev-
210 dev-211 dev-212 dev-213 dev-214 dev-215



Density matrix

           1     2
       ----- -----
    1  0.281 0.043
    2  0.030 0.003
```

9.8 Appendix: Information letter to mod_perl contributors

E-mail by Pål Fugelli
Date: 2009-07-07 09:58
To: modperl
Subject: Information about doctoral research project

Dear mod_perl contributors,

This is to inform you briefly about my research purpose in relation to the mod_perl project. I am a PhD student in educational science at University of Oslo, Norway. The research is focused on interaction and knowledge sharing in open-source projects. More specifically, it is about how shared understandings evolve around open-source codes.

Due to the dynamic distributed peer review and high quality output of the mod_perl project, it is of special interest for gaining deeper insight into the 'knowledge machineries' associated with open-source software development.

The plan is to read through the mailing list archive available at http://mail-archives.apache.org/mod_mbox/perl-modperl/ and study the content of postings from two contrasting sub-groups within the project that contribute frequently and less frequently.

The research is conducted from July 15th through to November 29th of this year. Results will be published and shared in a monograph with the working title "Intersubjectivity around expert objects: Inquiries into the knowledge practices of software development".

Keep up the good work.

Best regards,
Pål Fugelli
PhD Researcher, University of Oslo