FREEZING LANGUAGE

Conceptualisation Processes across ICT-Supported Organisations

Stijn Hoppenbrouwers

SIKS Dissertation Series No. 2003-14

The research reported in this thesis has been carried out under the auspices of SIKS, the Dutch Research School for Information and Knowledge Systems. It was supported by ID Research B.V., the *Nijmeegs Instituut voor Informatica en Informatiekunde* (NIII), and the *Informatiekunde* department of the University of Tilburg.

Cover photo: Jan Scholten

ISBN 90-9017318-8

FREEZING LANGUAGE

CONCEPTUALISATION PROCESSES ACROSS ICT-SUPPORTED ORGANISATIONS

een wetenschappelijke proeve op het gebied van de
Natuurwetenschappen, Wiskunde en Informatica

Proefschrift

ter verkrijging van de graad van doctor
aan de Katholieke Universiteit Nijmegen
op gezag van de Rector Magnificus prof. dr. C.W.P.M. Blom
volgens besluit van het College van Decanen
in het openbaar te verdedigen op woensdag 10 december 2003
des namiddags om 3.30 uur precies

door

Stijn Johannes Baptist Apollonia Hoppenbrouwers

geboren op 19 juli 1970
te Nijmegen

Promotors:
prof. dr. H.A. Proper
prof. dr. M.P. Papazoglou (University of Tilburg)
Copromotor:
dr. H. Weigand (University of Tilburg)

Manuscript committee:
prof. dr. R.P. van de Riet (Free University of Amsterdam)
prof. dr. V.E. van Reijswoud (Uganda Martyrs University, Uganda)
prof. dr. J.R. Taylor (University of Montreal, Canada)

For Roos Wijnants and Frans Hoppenbrouwers

What we construct through our concepts is not reality.

Thich Nhat Hanh, "Zen Keys"

I cannot speak except from my own experience and with my own personal language, always running the risk that this language does not reach beyond my lips.

Translated from Ton Lathouwers, "Meer dan een mens kan doen"

Preface

In the summer of 1995, I had moved back to my home town of Valkenswaard after finishing university. Having received both a Dutch MA in English Language and Literature (Utrecht) and a Welsh MA in Linguistics (Bangor), I felt somewhat disillusioned about the field of research I had thus far specialised in (generative linguistics), in particular its lack of genuine applicability. Inspired by the work of my brother Jeroen, I strongly felt that there must lie opportunities for an open-minded linguist in the vast domain of information systems analysis and design. I read a book about NIAM information analysis and then went to harass Hans Weigand at the Infolab of the University of Tilburg, informing him of my intentions to do a PhD there. Hans briefly inquired why I had come to him instead of to the Arts faculty, and then pointed out a desk where I could start writing my proposal.

Three years later, I was still thinking about the proposal. In the mean time, I had worked full time on two research and development projects at the Infolab: the design and implementation of a CASE-tool based on grammatical analysis (the Grammalizer project), and a large multi-lingual lexicon for use in an information filtering system (the TREVI project). By then, I had become quite familiar with the practical and theoretical problems posed by the task of capturing bits of natural language in order to put them to use in various kinds of information systems. One central question kept turning up in my mind: how can an optimal balance be struck between, on the one hand, the intuitive, free use of natural language (as encountered in good old speaking and writing), and on the other the use of its rigid, systemised, engineered counterpart (as built into information systems). Apart from issues concerning the representation of language and meaning, I felt that some key answers might lie in the *processes* that lead to the construction (and adaptation) of such representations.

In the fall of 1998, I was asked by Gerard Wijers and Erik Proper to join ID Research, Gouda. They allowed me to work on a PhD thesis half time, working the other half as a 'junior research consultant'. This kick-started my actual PhD project, which was to be supervised by Hans Weigand at the Infolab. Meanwhile, working in Gouda gained me valuable experience and insight in advanced ICT[1] architecture and alignment projects. Becoming part of ID Research's wonderful band of clever and industrious characters was an experience that seriously influenced both my intellectual and my personal life, mostly in a very positive way indeed. I thank all my former ID Research colleagues for having made life easy for me in many ways and very hard in some; industry can be an education. Apart from Gerard and Erik, I would like to explicitly thank Hans Bosma, Thijs Ott de Vries, Sven van de Riet, Andries Stam, and Denis Verhoef, for their support and constructive interest in my

[1] Information and Communication Technology

academic work.

In this period, work on my thesis took place mostly at the Infolab of the University of Tilburg. Many thanks to everyone there. Special thanks to Hans Weigand, who was not only my chief supervisor, but who has been a main intellectual compass for me in matters concerning language, philosophy, and ICT for over seven years. His conscientiousness and kind rationality have been an example to me. Explicit thanks also to Willem-Jan van den Heuvel, Jeroen Hoppenbrouwers, Alice Kloosterhuis, Frans Laurijssen, Kees Leune, Aldo de Moor, Cristina Moreira, and Mike Papazoglou.

The road to finding the right focus on the PhD project was a long and winding one. In the end, it was the architecture approach to information system development that brought me home, with Erik Proper as my main guide. Our collaboration was greatly intensified when first me (2000) and then Erik (2001) moved to the University of Nijmegen. It was there that the thesis really started to take shape: I was to provide a well-founded, architecture-oriented framework of analysis for conceptualisation processes in order to shed light on what I had metaphorically named the activity of "freezing language". From then on, I could start a case study, work towards a concrete analytical method, and further develop the underlying theory.

Industry had been an education; landing at the University of Nijmegen equally so. I suddenly found myself in a fairly traditional department of Computing Science (indeed with capital letters). Apart from continuing to work on my thesis, I spent a considerable amount of time doing ground work for the newly started *Informatiekunde*[2] curriculum, which put my whole PhD effort in a broader and increasingly fascinating perspective. In addition, it gave me the opportunity to teach –even about my own research. This I found inspiring and helpful.

The case study took place at Gak Nederland (now UWV) between March and December 2001. It was a true catalyst for my emerging ideas about language and ICT. I would like to thank Albert Souman and Louise Hellwig of Gak 'Gelderland Zuid' (Arnhem, Nijmegen), and Sam Korteweg of Gak 'DZ-AG' (Amsterdam), for kindly allowing me to be nosy and draw some resources from their departments without giving much back. Special thanks to Jaap Zweers of Gelderland Zuid, who helped me enormously with the obtaining of a detailed overview of the complex Gak situation, and with acquiring some crucial example data. Of course, I also thank my interviewees, who have to remain nameless here, and other people at Gak who kindly helped me do my stuff, which I am sure must have seemed rather obscure to them. Finally, I thank Mario Seekles of PinkRoccade/ASZ, for sharing his views on information and system architecture at Gak.

Once I landed in Nijmegen, the IRIS[3] group headed by Theo van der Weide quickly became my professional home. I thank IRIS and the *Nijmeegs Instituut voor Informatica en Informatiekunde* for allowing me to spend time on writing a PhD

[2] *Informatiekunde* (a hard to translate Dutch word) aspires to combine Computing and Computer Science with a system-oriented approach to the environments that ICT is intended to function in, in order to achieve optimal alignment in the broadest sense between individuals, organisations, information, and computer technology; see (Proper, 2003). We are tempted to translate it, awkwardly, as "Information Systems Science".

[3] Information Retrieval and Information Systems

that not only was not initially ‘theirs’, but also (and infinitely more suspiciously) contained *hardly any mathematics to speak of.* My gratitude extends to all people at IRIS, and others at the institute. Apart from Theo, who kept an eye on my general progress and provided sound strategic advice and support when the going was particularly tough, I would like to explicitly mention Sander Bosman, Franc Grootjen, Bas van Gils, Janos Sarbo, Gert Veldhuijzen van Zanten, and Hanno Wupper, for exchanging thoughts and providing comments and support concerning my thesis. In addition, working together with Angenita Heijmans, Arnoud Vermeij, and many other students has been both enjoyable and enlightening. Very special thanks go out to Erik Proper, who did not only put in more than his share coaching me in my daily work, but also has been a great source of professional inspiration and has been invaluable to me in terms of both intellectual and moral support.

On a more official note, I would of course like to thank my promotors Erik Proper and Mike Papazoglou, my co-promotor Hans Weigand, the manuscript committee consisting of Reind van de Riet, Victor van Reijswoud, and James Taylor, and the members of the promotion committee.

Thanks also to Jan Scholten for kindly providing the cover photo, and to Jeroen Hoppenbrouwers for helping me out with the last LaTeX hiccups.

Finally I would of course like to thank all my friends and family for their support. And in the same period that the University of Nijmegen became my new professional home, in my personal life I have been building one with Saskia Schiltmans, who was *the* reason I moved to Nijmegen. We got married in August 2001; our son Koen was born in March 2002, followed by our daughter Eleanora in September 2003. Saskia’s love and support have been absolutely crucial to me, especially in that notorious final year of the writing process.

And now: Onwards.

Stijn Hoppenbrouwers
Nijmegen, October 2003

Contents

Chapter 1

Introduction

In this introductory chapter, we briefly present the subject that underlies this thesis (the "Frozen Language Issue"). We provide a summary of our analysis of this issue, and a possible solution of some of the main problems it entails. This leads to the formulation of our research questions and the presentation of our approach to answering these questions. The problem analysis is indeed only summarised: the detailed analysis of the Frozen Language Issue is too extensive to be included in the introduction, and constitutes Chapter 2.

1.1 Area of Research: Language Functionality of Information Systems

We are concerned with a particular aspect of the functionality of computerised information systems: the language aspect. Many information systems make use of pre-specified, built-in words and phrases to enable people to communicate with each other. Typical examples of such language items are names of database fields, pre-listed words or symbols that may be used to fill in database fields (menu choices), terms used for categorisation, and pre-specified requests and signals. Predetermined and fixed language thus is a crucial part of the functionality of most information systems; users often have to make do with previously specified sets of language items made available through a computerised system. Selection and specification of such sets (a form of language engineering) is an important part of the design of the functionality of information systems. We call such specification 'freezing language', and its result 'frozen language'.

Our work over more than a decade, concerning linguistics as well as language-related aspects of information system development, has made us aware of limitations in language functionality that seem to result directly from building fixed language into machines. Crucially, such language by nature remains unchanged unless the information system itself is changed. If a language item does not work well for some communication situation that should be supported by the system (for example, because a desired database field is lacking), there is little that can be done about it on an immediate basis. In addition, there often is a lack of knowledge about what some available language items mean, both at the sending and at the receiving end of the utterances produced through the system. In other words, both the language and the knowledge of the users about it may be flawed, damaging language functionality and ultimately the functionality of the information system.

So we wonder what exactly happens if language items (typically words) are built into machines (typically, information systems) used for human-to-human communication. In particular, we wonder what may go wrong because of this and how such negative effects may be countered, though along the way we also need to consider the positive aspects and the reasons why language is frozen in the first place. We will first thoroughly analyse the phenomenon and the problems involved. The crucial insight this brings us is, in a nutshell, that the nature of language and language use conflicts with the creation and use of frozen language. The necessary use of *static representations* of language in information systems engineering often hinders or disables some crucial *dynamic* properties of language.

This initial analysis will lead us to the exploration of a particular solution to the problem that centers around various ways of *communicating about language* ('linguistic meta-communication'). We propose that to 'better freeze language' (i.e. create more functional static representations of language), we need to look beyond the representations as such and focus on the dynamic and diverse *processes* that lead to the creation of language representations, and on the *environments* such representations are to be used in. In addition, we suggest that to compensate for the negative effects of freezing language, it is not sufficient to 'better freeze language'. Processes for freezing language need to be *complemented* by other processes for communicating about language. Generally spoken, the approach presented in this thesis should help understand the basic language-and-meaning games that are played among the various stakeholders involved in information system development and use. We hope this eventually leads to those games being played more skillfully, resulting in better language functionality across the board, and ultimately in better information systems.

1.2 Aim, Problem Statement, Suggested Solution, and Research Questions

The aim of this thesis is to *provide deeper insight in* but also *generate more awareness of* the frozen language issue, which has so far been insufficiently understood and the complexity and implications of which seem to have been underestimated. Given the nature of the frozen language issue, it transpires that improvement of the 'freezing language' aspect of information systems should not be looked for exclusively in instances of information systems, but also (and perhaps predominantly) in the *processes* that constitute the development, evolution, use, and management of information systems. We therefore aim in particular to improve insight into and awareness of matters of freezing language as involved in *processes of information system development and use.*

Assuming that the frozen language issue is indeed relevant to the development of 'better' information systems, understanding and awareness of the frozen language issue is of course part of the larger effort to improve the art and science of building good information systems. Eventually, therefore, we aspire to *help build better information systems by making it possible to avoid or solve problems related to the* *frozen language issue*. However, we consider this too pretentious an aim to explicitly commit to with respect to this particular thesis. Most of all, within the scope of our PhD project it would have been next to impossible to validate our results in terms of having achieved "better information systems". Therefore, providing better insight and awareness will have to do for now.

To achieve our main goal, we will have to show that the phenomenon of freezing language exists and how it can be made visible. We will have to explain its deeper motivations and underlying mechanisms, centring around *processes of freezing language* and the *environments* they take place in. Minimally, this should enable us to better see and understand the limitations and possibilities of creating and using frozen language in relation to information systems. However, this is likely to also enable us to suggest some ways in which related problems may be countered.

We formulate the main problem as follows.

> If representations of linguistic concepts are built into information systems and are then used as linguistic means for human-human communication, the linguistic and communicative functionality depending on those representations may suffer. This is the case because the larger, dynamic context of language use and language freezing is not taken into account. In particular, the freezing of language often entails *absence or inadequacy of meta-communication processes vital to the effective functioning of language in human-human communication*.

We also suggest a direction for the solution of the problem:

> If the absence and inadequacy of various processes of linguistic meta-communication underlies the problem, an obvious solution is to *put such processes in place, and/or improve them*. However, as mentioned, achieving this in practice is not the immediate goal of this thesis. We will settle for better understanding of the processes in question. Importantly, we focus on *processes resulting in language representations* instead of on language representations as such –contrary to most other approaches to the language functionality problem.

A central activity in the freezing of language is the creation of *concept descriptions*. We call such activity *explicit conceptualisation*[1]. Thus, we place processes of explicit conceptualisation central in our analysis, but emphasise their embedding in *conceptualisation environments* which are the stage for various other processes of linguistic meta-communication as well as for language use. Importantly, the conceptualisation processes should fit into and *support* the language environments they take place in.

Given the basic research problem (the absence or inadequacy of linguistic meta-communication processes in information system development and use), the chief, immediate goal of this thesis (increasing insight and awareness concerning the research problem), our longer term goal (supporting the development of better information systems), and the suggested solution to the problem (enabling adequate conceptualisation processes in view of the language environments they take place in) we formulate two basic research questions:

[1] As opposed to *implicit* conceptualisation, which is the 'internal' socio-cognitive activity resulting in the emerging of concepts in human minds; see Chapter 3.

Q1 Which theoretical framework enables us to describe and analyse processes and environments in which language is being frozen?

Q2 What is an operational form of this theoretical framework, by which we might use it for concrete analysis and evaluation of processes for freezing language in an information systems context, in order to improve such processes?

These two questions are of a somewhat different nature, and require a different approach in order to be satisfactorily answered. Question one suggests the formation of a theoretical framework that can, up to a point, *describe* and *explain* the phenomena we focus on. Question two suggests the development of an operational method based on the theoretical framework. We will now discuss the two approaches separately.

Theoretical Underpinnings of Freezing Language

Question 1 can be decomposed:

Q1.1 What are the processes involved in freezing language, and how can we describe them?

Q1.2 What are the environments in which the freezing of language may take place, and how can we describe them?

However, we should not try to see processes and their environments as strictly separate, and hence must also ask a third question:

Q1.3 How do processes and environments for freezing language relate to and influence each other?

We need to ask some additional questions in order to appropriately ground our theoretical framework in existing theory and apply it to the domain of information systems design and management. For this we require both a set of focussed theoretical concepts and an overview of the approaches, methods, tools, and techniques that are involved in the freezing of language in information system development. Therefore we ask two more questions:

Q1.4 What are the linguistic mechanisms and factors underlying the description of, need for, and course of the processes of freezing language?

Q1.5 How are processes and environments for freezing language related to information system development?

An Analytical Method to Operationalise the Theory

The second part of our effort is not purely theoretical. It concerns an approach aiming to help face actual challenges concerning freezing language in an information systems context. However, given the scientific nature of our effort, this approach should of course be based on the framework and observations covered by Q1. The two sub-questions underlying Q2 are formulated as design goals rather than actual questions:

Q2.1 Develop a conceptual framework for description and analysis of situation-specific aspects of frozen language in the ICT[2] domain.

Q2.2 On the basis of the conceptual framework, develop an analytical method enabling the analysis and evaluation of concrete processes and environments of freezing language in real ICT situations.

After answering all questions posed, we should be able to understand, describe, and explain matters involving the frozen language issue as it occurs in information systems development and use. This should eventually help avoid or solve observed problems related to this issue. It would thus indeed fulfil the 'insight' aim. As for the 'awareness' aim: if researchers and professionals are willing (and able) to read and digest this book, we are confident that they will indeed be more aware of a number of important and previously unknown language-related mechanisms and factors at play. In the end, we can only hope all this actually contributes to the improvement of information systems –but we believe that indeed it will.

1.3 Our Approach to Answering the Research Questions

To answer the research questions, we follow a five step approach:

1. Adopt or develop concepts necessary to understand and analyse the mechanisms underlying the frozen language issue, in particular explicit conceptualisation processes and the environments they take place in (Chapters 2, 3, 4, and 5).

2. Study approaches to and practices of explicit conceptualisation as documented in the literature of various relevant fields (Appendices A and B, reflected mostly in Chapter 5).

3. Develop a comprehensive theoretical framework for the description and analysis of processes of explicit conceptualisation, on the basis of various assembled theoretical concepts and the findings of our study of conceptualisation practices (Chapter 6).

4. Create an operational method for analysis of conceptualisation processes and their environments on the bases of the theoretical framework (Chapter 6).

5. Achieve a 'proof of concept' of the method by applying it to a real situation: an existing organisation (Chapter 7).

[2] ICT stands for Information and Communication Technology, which is a bit of a Dutchism. In Dutch, ICT is roughly synonymous to IT but the C is now commonly added. In this thesis, the emphasis lies on the *C* in ICT: Communication.

1.3.1 Theoretical Background

In developing the ideas presented in this thesis, we build on existing theory that helps us deal with explicit conceptualisation, viewed as an exponent of meta-communication, in the context of restricted language use through an information system. We select basic theoretical concepts from various theories relevant to the issues at hand, but also add some refinements and new concepts to suit our particular theoretical goals. In fact, quite a few of the concepts introduced in the thesis are newly coined, but our ideas have of course been inspired by various different theories, fields, and disciplines.
A small number of clearly identifiable theories form the basis of the conceptual framework in this thesis. Other theoretical fields have been a more general, implicit inspiration. As an example of the latter: this thesis contains no references to generative or 'Chomskian' linguistics, because no actual concepts have been taken from this distinguished academic field. Nevertheless, generative linguistics has been an undeniable influence and we are intellectually indebted to it.

Linguistics

Our linguistic theoretical ideas are inspired by the work of one of the founding fathers of modern linguistics, Ferdinand de Saussure (de Saussure, 1916), and (as mentioned) by ideas from generative linguistics as championed by Noam Chomsky and his countless followers. However, we are much more explicitly indebted to a linguistic movement that stands apart from mainstream generative linguistics: functionalist linguistics, in particular Functional Grammar. Relevant names that should be mentioned here are Dik (Dik, 1989), Weigand (Weigand, 1990), and Harder (Harder, 1996). A functionalist approach to definition is gratefully borrowed from Viskil (Viskil, 1994). Discussion of general approaches to Semantics and Pragmatics are mainly based on texts by Cruse (Cruse, 2000) and Clark (Clark, 1992), though they in turn draw from the linguistic tradition in general, including cognitive linguistics and psycholinguistics. Finally, though it serves mainly as a source on terminology-related practices rather than linguistic theory, Wright and Budin's *Handbook of Terminology Management* (Wright and Budin, 1997; Wright and Budin, 2001) deserves to be explicitly mentioned here. Especially the second volume has been a tremendous help in exploring the interdisciplinary no-man's-land in between conceptualisation and information system development.

Computer Science

This thesis is strongly inspired by, even builds on, general database theory, especially on the Universe of Discourse principle and the view that natural language is important as a basis for information analysis. Some relevant names here are Nijssen (Nijssen, 1993), Halpin (Halpin, 1995), van de Riet, van der Weide, and Weigand (Weigand, 1990) (among many others). A more immediate inspiration in this vein is the LIKE project[3] (van de Riet and Meersman, 1991), and in particular some of the PhD theses that it brought forth: (van Reijswoud, 1996), (Burg, 1997), (Hop-

[3]Linguistic Instruments in Knowledge Engineering

penbrouwers, 1997), (Verharen, 1997), (de Moor, 1999). We would like to think that this thesis can be seen as a belated, extraneous contribution to the LIKE project.

Information and Communication Theory

Distantly related to the database and information analysis tradition, but specifically orienting on communication theory, the Language Action Perspective as initiated by Winograd and Flores (Winograd and Flores, 1986) plays a crucial role. Names of chief contributors here include those of Weigand, Dietz, Goldkuhl, and Schoop (among others). The work of Van Reijswoud (van Reijswoud, 1996) has been particularly influential. The Language Action Perspective at large owes much to work by Austin (Austin, 1962) and Searle (Searle, 1969), and Habermas (Habermas, 1984). Winograd and Flores at times lean on ideas by Heidegger. Finally, we were also strongly inspired by James Taylor's work on communication and organisation theory (Taylor, 1993).

1.3.2 Study of Documented Conceptualisation Approaches and Practices

Our study of conceptualisation practices and approaches covers a number of fields: conceptual analysis and design; computer supported cooperative work; ethnography; knowledge management and knowledge engineering; information retrieval and library systems; lexicography, translation, and terminology management. Though the study as such has constituted a fair part of our effort to understand conceptualisation and its current place in information systems development, the two resulting overviews are presented as appendices, since they are only of secondary importance to the actual thesis. However, some important conclusions drawn from the study, as well as a sizable list of 'conceptualisation techniques', are included in the main text.

1.3.3 Comprehensive Theoretical Framework

The analytical framework is composed of various theoretical components. The main motivation behind the selection of the individual components and the way they are linked hinges on the principle that language, meta-communication, conceptualisation, and information system development are interrelated phenomena that serve to fulfil each other's requirements. For example, linguistic meta-communication supports language functionality by enabling adaptation of language and the sharing of linguistic knowledge; explicit conceptualisation is an activity that fulfils requirements posed in view of some meta-communication processes. The framework should make it possible to view the freezing of language in terms of the various requirements posed and possibilities offered within some particular situation of language creation and use.

1.3.4 Operational Analytical Method: an Architecture Viewpoint

The proposed method should enable us to seek out and describe the relevant mechanisms and factors with respect to the frozen language issue in organisations, and evaluate the situations encountered. It need not, however, be an ‘industrial’ method; it will have to be operational only in the sense that it can be readily used to chart a real domain in terms of conceptualisation processes and their environments. Yet it will still be an ‘academical’ method in that we will not try to prove its worth in the *actual* running, managing, and changing of organisations. Even though the method has to be applicable in to real, concrete cases and generate useful insights concerning them, it is as of yet intended as a ‘tool for thinking’ rather than a ‘tool for management’.

The method should not only enable us to chart organisations, it should also help guide balanced choices related to the interests and priorities of the various different stakeholders in some particular domain. We therefore shape our final model as an *architecture viewpoint* as defined in IEEE standard 1471 (Sherlund et al., 2000). This makes our model compatible with architectural approaches to information systems development, making it possible to integrate our viewpoint with others within the context of general information system development and management. Our views on information systems architecture and architecture description are mostly indebted to ongoing work by Erik Proper; in this field, some influence was also felt of ideas of Jan Dietz. The IEEE standard 1471 (Sherlund et al., 2000) was a beacon concerning architecture description.

We will now briefly discuss what is meant by ‘Architecture’ in ICT. This is not the place to discuss this field at length, nor to question its value in solving general ICT problems. In other words, in principle we assume ‘architecting’ to be indeed a valuable activity, and will attempt to formulate our own, conceptualisation-oriented analytical framework in such a way that it can be ‘plugged into’ architecture thinking and may become part of it.

In ICT terminology, a system’s ‘architecture’ refers to the “fundamental organisation of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution” (Sherlund et al., 2000, p.3). ‘Architecting’ thus is “the activity of defining, documenting, maintaining, improving, and certifying proper implementation of an architecture” (*ibid.*).

The definitions above are quoted from Recommended Practice 1471-2000 of the Institute of Electrical and Electronics Engineers (IEEE). This standard specifically addresses “the activities of creation, analysis, and sustainment of architectures of software-intensive systems” (*ibid.*, p.i), and thus is essentially technology-oriented, but it has been embraced by a wider audience (including information system and business process designers) as a standard for what systems architecture is and what it entails in terms of working practice, modelling, and documentation.

According to the ‘IEEE 1471’, architecture descriptions are used for the following:

- Expression of the system and its potential evolution
- Communication among the system stakeholders

- Evaluation among the system stakeholders
- Planning, managing, and executing the activities of system development
- Expression of the persistent characteristics and supporting principles of a system to guide acceptable change
- Verification of a system's implementation's compliance with an architecture description
- Recording contributions to the body of knowledge of software-intensive systems architecture

General stakeholders in system development and evolution, i.e. intended users of architecture descriptions, are listed as follows (Sherlund et al., 2000, p.3):

- Those that use, own, and acquire the system (users, operators, and acquirers, or *clients*)
- Those that develop, describe, and document architectures (architects)
- Those that develop, deliver, and maintain the system (architects, designers, programmers, maintainers, testers, domain engineers, quality assurance staff, configuration management staff, suppliers, and project managers or *developers*)
- Those who oversee and evaluate systems and their development (chief information officiers, auditors, and independent assessors)

Figure 1.1 is a diagram defining twelve core concepts as presented in IEEE 1471 (p.5)[4].
Recently, architecture has –not without some justification– been branded a 'hype'. Indeed it has sometimes been sold to client organisations as being yet another definite answer to their every ICT need. Acknowledging the limitations of the architecture approach in its current shape, we would nevertheless like to express our strong conviction that to take into account (i.e. describe and evaluate) a number of different perspectives and concerns while creating information systems, thus seeking an optimal balance between all means and ends involved (alignment), and thereby ultimately working towards more functional and more economical solutions to real

[4]We also include part of a footnote as originally included alongside the diagram: "[The figure] provides an informative summary of the key concepts introduced by this recommended practice and their inter-relationships. [...] In the figure, boxes represent classes of things. Lines connecting boxes represent associations between things. An association has two roles (one in each direction). A role can optimally be named with a label. The role from A to B is closest to B, and vice versa. For example, the roles between SYSTEM and ENVIRONMENT can be read: A SYSTEM inhabits an ENVIRONMENT, and an ENVIRONMENT influences a SYSTEM. In the figure, roles are one-to-one unless otherwise noted. A role can have a multiplicity, e.g. a role marked with "1..*" is used to denote *many*, as in one-to-many or many-to-many association. A diamond (at the end of an association line) denotes a *part-of* relationship. For example, VIEWS are a part of an ARCHITECTURAL DESCRIPTION. This notion is from the *Unified Modeling Language Specification* (Object Management Group, 1997)."; (Sherlund et al., 2000, note on pp.5-6).

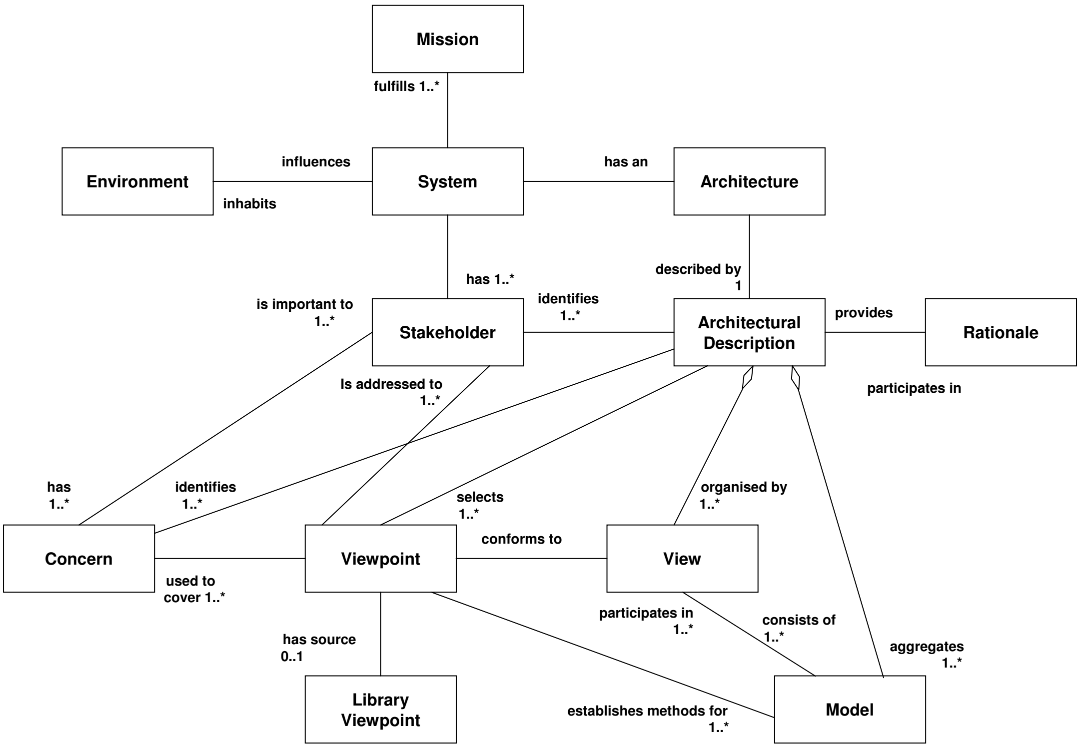

Figure 1.1: Conceptual Model of Architecture Description

problems, is a practice that is here to stay. Whatever buzz-words or trends are temporarily associated with it, large-scale development and introduction of ICT in organisations is a very complex game, and to understand and control it advanced approaches and tools will be needed. One may call it 'architecture', or 'high-level, pluriform, interrelated (sub)-system description', or whatever, but the bottom line is that we will not be able to do without it given the ever increasing scale, complexity, and pervasiveness of the application of ICT in our society.

Why the Architecture Approach is Relevant to the Frozen Language Issue

Why choose an architecture approach in context of the freezing of language? Architecture is the main field where many different aspects of information system development and management meet. Though some aspects are more standardly incorporated in architecture thinking than others (prominent ones are the *process aspect*, the *information aspect*, the *technological aspect*, and the *strategic aspect*; see van den Heuvel and Proper, 2002), the number of aspects or 'sub-aspects' that may in principle be taken aboard is endless, and the 'frozen languages aspect' is one of them. In addition, the architecture approach enables us to look beyond system design as a mere result: to include considerations of the development *process* (system development architecture). This is especially relevant to the process-oriented aspect derived from the 'frozen language aspect': the conceptualisation aspect.

Freezing language for use in information systems, or as we can now more precisely call it, 'explicit conceptualisation in ICT supported environments', is after all closely related to software development in relation to operational processes in organisations. For concrete solutions involving the frozen language issue to emerge, integration and alignment of language aspects, technological aspects, and operational process aspects (among others) in view of information system development and use is direly needed. We deliberately construct our language-oriented framework with accepted architecture and system development practices in mind. We thus attempt to extend the reach of architecture by adding to it an advanced language-oriented perspective.

1.4 Structure of this Thesis

After this introductory chapter, which sketches the area of research, the problem we focus on, the 'solution direction' chosen, and our approach, we continue with a detailed analysis of the Frozen Language Issue. First, we introduce our view on language and how it relates to information systems. This view is boiled down to a formalised conceptual framework. Next, we explain why language is frozen and why the functionality of frozen language may be sub-optimal. We then consider the role the Frozen Language Issue plays in current ICT practice and how this role relates to the general approach we suggest: creating and supporting processes of meta-communication and conceptualisation, and aligning these processes with the language environments in which they take place. This rounds off Chapter 2.

In Chapter 3 we present an overview of theory that underlies our view on language and its relation to communication. After some basic considerations of what language is and what it is for (which reflects our functionalist approach to language), we discuss the limitations of the use of representations to capture the meaning of language. This leads us to a particular view on language representation, namely as a means for linguistic meta-communication. We present some existing communication theory and explain how meta-communication fits in, paying particular attention to the activity of definition.

In the next chapter, we introduce some theoretical considerations concerning language and meta-communication in organisations, and how this relates to the development and use of information and communication technology. We extend our conceptual framework, adding to it some concepts needed to deal with language specification and use in context of ICT-supported organisations. We also present a typology of language use intended to facilitate the identification of various different language needs, and a typology of meta-communication processes that may occur in relation to ICT. We provide examples of the types of meta-communication distinguished, and of how they can be used to compensate for the sub-optimal functionality of frozen language. This concludes Chapter 4.

In Chapter 5 we focus on a particular type of linguistic meta-communication: explicit conceptualisation. On the basis of an elaborate study of conceptualisation practices as documented in the literature of various fields (as reported in Appendix A), we consider the current state of conceptualisation practice, and what is unsatisfactory about it. In order to enable better alignment of conceptualisation processes with the situations in which they occur, we develop some theoretical concepts for the description and analysis of conceptualisation processes. We present in considerable detail 5 generic phases of conceptualisation processes that we distinguish, covering all aspects of explicit conceptualisation. These phases form the foundation of our descriptive/analytical framework for conceptualisation processes, which also includes a typology of sorts of input and output of each phase, quality attributes for the conceptualisation process, and techniques that may be used to implement particular conceptualisation phases (29 categories of conceptualisation technique are described in Appendix B). This concludes the theoretical part of the thesis.

We continue with the presentation of an architectural viewpoint for conceptualisation (Chapter 6): a coherent analytical framework based on the theory presented in chapters 2 to 5, supported by dedicated diagrams and question templates, and a general 'way of working'. First we briefly discuss what an architecture viewpoint should consist of, and summarise the concepts to be included in the framework. We then consider various areas of concern to which these concepts belong, and how these areas are related. This leads to the presentation of the general structure of our method, which consists of four analytical stages and one evaluative stage. These stages, and in particular the analytical instruments belonging to them, are then presented.

The analytical method discussed in Chapter 6 is put to the test in Chapter 7. We describe an elaborate case study of a real organisation. After presenting the research set-up and reporting our general experiences, we describe the information system that is central to our study (a general description of the case organisation can be found in Appendix C). We then present the results of the case study, structured along the lines of our method. We conclude with an evaluation of how well the method worked, including a detailed evaluation of a considerable number of individual analytical concepts and instruments.

In the concluding Chapter 8, we summarise our results and discuss how well they answer the questions we promised to address in this thesis. We also discuss various directions for further research that we envisage, concerning further validation and development of our theory and method, but also application of our ideas to conceptualisation support and to various fields of research outside the scope of the thesis.

Chapter 2

The Frozen Language Issue

In this chapter, we explore and define the Frozen Language Issue. We first introduce some basic concepts and theoretical notions needed to describe our focus on (frozen) language and its relation to information systems. We then discuss the frozen language phenomenon in some depth, considering its negative consequences but also some basic reasons why language is (indeed, may need to be) frozen. We also provide some illustrations, and present an overview of various ways in which people are currently confronted with, and cope with, frozen language. We finish with some conclusions concerning the current "frozen language situation" in the general field of ICT, and indicate what we hope to contribute to improve this situation.

2.1 Language in an ICT Context

In order to clarify our analysis of the frozen language issue, we start with a discussion of the way we use some basic notions. The central terms in question are *information system* and *language*. By explaining what we mean by them, and which relation between them exists, we hope to clarify our focus and interest.

With *information system* we refer to a computerised system designed to offer support to people performing information-related tasks in some organisation; for example, a database system. We emphatically do *not* mean the whole socio-technical system in which the database may be embedded[1]. We also keep apart the logical abstraction of an information system and its actual physical implementation as a computational machine.

The point of view on information systems taken in this thesis is strongly influenced by their relation to language and communication. We view language primarily as a means that supports communication. We are specifically interested in information systems as an enabling factor in human-human communication, in which language plays a key role. Implemented information systems are thus seen as *medium machines*; they are machines that are used as a means for human-human communication.

[1]This is a matter of definition, not a fundamental point of view. We are in fact very much in favour of viewing information systems in intimate relation to their users and even their developers, but for our current discussion we need a clear separation of the human way of 'doing language' on the one hand, and of language representations as built into engineered systems on the other.

2.1.1 Our View on Language and its Relation to Information Systems

Let us begin by stating that language as we view it concerns spoken or written symbols: elementary symbols (words) and complex symbols (phrases or sentences).[2] Such symbols are closely related to *concepts*. A popular interpretation of the notion "concept" is that concepts are abstract notions of meaning. For example, it may be assumed that the word-symbol "cat" in English and the word-symbol "kat" in Dutch roughly refer to the same concept. Confusingly, however, in the literature the term "concept" is both used to denote isolated meanings (as in the cat example) and to denote *combinations* of form and meaning. This appears to be the case because in most situations it is highly impractical to refer to some meaning without using the form commonly representing it. Acknowledging the theoretical elegance of the former definition of "concept", we plead guilty of using the latter. We thus refer to concepts by means of some form, and even go as far as defining a concept as a *combination of a form and a meaning*[3]. The relation between a form and a meaning is a matter of *interpretation* by the language user. Importantly, the meaning carried by a language form that is actually *used* in some representation of language (for example, a sentence being written) is combined with contextual information to render its full interpretation. However, linguistic concepts can also be seen 'out of context', in which case they serve the purpose of *triggering* interpretations. In other words, the concepts of a language are *decontextualised*. For more on this, see Chapter 3.

The languages most commonly associated with information systems are *programming languages* and the like, rather than 'natural language'. However, our research primarily concerns 'natural' use of language, as an instrument for human-to-human communication. So we do talk about information systems, but focus on natural language communication through these systems (for a similar view, see Andersen, 1990). We very specifically do *not* focus on *language designed for mechanical use*, as seen, for example, in programming languages. By 'mechanically used language' we mean language that is used in a mechanical process of symbol manipulation. Computational systems for a large part depend on such symbol manipulation, and it is fundamental to our approach that we keep apart language use in human-to-human communication and language use in mechanical processes. As an intermediary form of language use, *mechanically processed natural language* should also be mentioned: language use in which symbols originating in natural language are subject to mechanical processes of symbol manipulation. This type mostly belongs to the field of Computational Linguistics, and it too lies outside our immediate focus.

In linguistic terms, this thesis mainly concerns typical elementary language symbols (lexis), and matters related to word forms and word meaning. This is because lexis is the aspect of language most relevant to the process of freezing language: it is frequently subject to explicit description and consideration by non-linguists –unlike, for example, rules of grammar. Also, semantics (the study of the meaning

[2]We discard non-verbal communication, though we acknowledge the role it plays in face-to-face human communication. Contemporary information systems including non-verbal forms of language are simply outside our current scope.

[3]In fact, in our framework the notion of *concept* closely resembles the notion of *sign* as defined by de Saussure (for more on this, see Section 3.2.1).

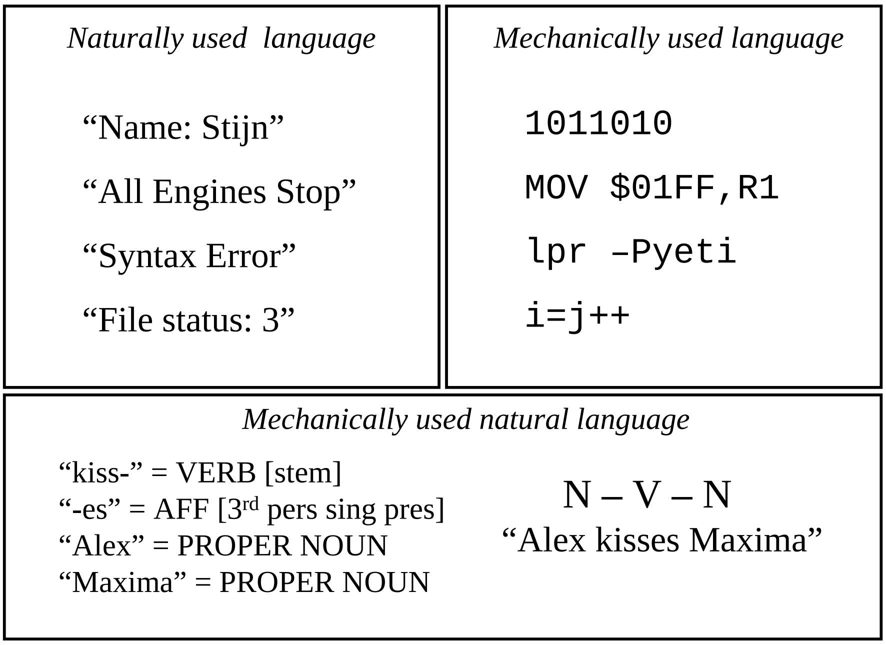

Figure 2.1: Representations of three types of language use

of symbols) is more prominently involved than syntax and morphology (the study of grammatically combining elementary symbols to form complex symbols), and an interest in lexical meaning tends to prevail at the expense of matters of grammatical meaning (relational semantics). We are also concerned with word meaning *in context*, and therefore we need to combine semantic insights with insights from linguistic pragmatics (the study of language use in context). A discussion of selected linguistic issues can be found in Chapter 3.

Within the realm of language used for human-to-human communication, we distinguish two main types of language. We will refer to these as *open language* and *closed language*. The distinction between them is based on types of *language use*. Open language communication takes place through some medium system (computerised or not) that does not in principle restrict the meaningful symbols allowed[4]. So in a manner of speaking, open language does not restrict the 'conceptual bandwidth' available to the communicating parties.

However, we are chiefly interested in medium systems employed for *closed language use*. *Closed language communication*, then, takes place by means of an explicitly and deliberately restricted set of meaningful symbols. Typically and importantly, such limitation is *imposed* by the communicative situation, so to communicate people have to make do with the symbols available. This usually means that such language

[4]The letters of the alphabet, for example, are not considered to be typical meaningful symbols, since they are normally used to compose meaningful symbols *with*: words, or morphemes. In this vein, we define "meaningful symbol" by referring to the old structuralist linguistic notion of *sememe*: the smallest possible semantic unit (Kaldewaij, 1986).

restriction goes beyond mere agreement about the set of concepts used: it is carried though in the medium system. Indeed, we are chiefly concerned with *closed language as imposed through the deployment of an information system.*

Within context of Information and Communication Technology (ICT), typical media for open language communication are audio and video channels (e.g. voicemail, video conferencing, audio and video recordings) and freely written language media (e.g. documents produced with a generic editing tool (typewriter, word processor), and e-mails). Such 'media' (audio, video, writing) are abstract notions which may be instantiated in actual medium systems. The minimal medium system, perhaps, is the air present between two speaking individuals. More typically, it may be a telephone system or e-mail facility. Within our particular focus, it is usually an implemented information system.

Consider the following example. In a relational database system, the label of a data field (e.g. a field named 'Gender') is normally fixed, at least from the user's perspective. Therefore, field labels are part of a closed language, and using them boils down to closed language communication. The symbols used to express the *content* of a specific data field may be either closed (e.g. a choice has to be made between either 'male' or 'female') or open (any string may be entered). More often than not, open and closed language communication occur side by side, and though our main interest is in closed language use, open language cannot be ignored.

Some comparative examples of the main types of language use discussed are given in Figure 2.1.

2.1.2 Our Conceptual Framework for Language and its Relation to Information Systems

Having informally presented our basic focus, we now present a less informal conceptual framework. We use formalism strictly for reasons of clarity, not for proving essential properties.

We more precisely describe language, and the relationship between language and medium machines, as follows. Language helps people to *communicate meanings*: to cause meanings to occur in each other's minds (for more on this, see Chapter 3). Assume that communication, at least for the part relevant to us in the current discussion, takes place through use of a language $L \in \mathcal{LA}$, where $\mathcal{LA}$ is the set of languages intended for use in human-human communication. L is the complete language as *actually used* in a particular community or organisation, based on a natural language such as English. Sub-languages of L may exist, for example including linguistic items typical to a particular department.

Let $\mathcal{CO}$ be the set of all decontextualised *concepts* (both complex and elementary) that may be communicated by means of a language. We presume that for each $L \in \mathcal{LA}$ there is a set $C_L \subseteq \mathcal{CO}$ which includes the linguistic concepts belonging to language L.

A linguistic concept is the combination of a decontextualised linguistic meaning (involving types rather than instances) and a linguistic form (for example, the form "cat" and its meaning description). Consequently, let $\mathcal{FO}$ be the set of concept forms and $\mathcal{MG}$ the set of decontextualised concept meanings. We could therefore informally say that $\mathcal{CO}$ is a subset of the cartesian product of $\mathcal{FO}$ and $\mathcal{MG}$. It is a subset because not all combinations of form and meaning will be recognised as a concept. Note that forms and meanings can be both complex and elementary; at our level of abstraction, we just do not make explicit any construction rules (syntax) there might be. Formally, we model the conceptual form-meaning dichotomy by presuming that we have the following two functions:

$$\mathsf{Form} : \mathcal{CO} \rightarrow \mathcal{FO}$$

$$\mathsf{Meaning} : \mathcal{CO} \rightarrow \mathcal{MG}$$

With these functions and given a set of concepts C_L for a language L, we can define a set of associated forms and meanings as:

$$\begin{aligned} F_L &\equiv \{\mathsf{Form}(c) | c \in C_L\} \\ G_L &\equiv \{\mathsf{Meaning}(c) | c \in C_L\} \end{aligned}$$

For a given language L and its associated set of concepts C_L, we can therefore distinguish a set of forms $F_L \subseteq \mathcal{FO}$ and a set of meanings $G_L \subseteq \mathcal{MG}$. Note again that we discard the identification of compositional rules for the creation of complex concepts (grammar). Hence, we do not on a formal level distinguish between elementary and complex concepts within C_L.

Traditionally, the linguistic meaning underlying linguistic form is mostly represented by entities and relationships constituting context free predicates that may be placed in context and thereby become propositions (Gamut, 1991; Dik, 1989). "Entities", "relationships", and "predicates" are means for representing concepts. However, it is widely acknowledged that propositions are only a component of a larger linguistic category: the *speech act* (Austin, 1962; Searle, 1969). A speech act represents an attempt to *get something done through language*[5]. For example, the proposition "John kissed Mary" can be wrapped up as a question ("did John kiss Mary?"), or as a statement ("John did kiss Mary!"). In the first case, the speaker requires information about the truth of the proposition, in the second she claims the proposition to be true and so creates information. Speech acts thus consist of a proposition plus a so-called *illocution*: a conceptual device explicitly meant to represent what it is that a speaker wants to get done by means of the proposition. A limited but common classification of illocutions is the following (Searle, 1979; Searle, 1969):

- Assertives commit the speaker to something being the case (e.g., stating);
- Directives try to get the hearer to do something expressed in a proposition (e.g., questioning and commanding);
- Commissives commit the speaker to some future course of action (e.g., promising);
- Declarations bring about a (new) state of affairs by merely declaring it (e.g., declaring);
- Expressives express the speakers attitudes/feelings about a state of affairs (e.g., apologising).

[5]For more on speech acts, see Chapter 3.

We consider speech acts to be the highest component level of linguistic composition. We assume that $\mathcal{CO}$ covers not only concepts needed to compose predicates, but includes the entire conceptual apparatus needed to perform speech acts[6]. Formally, we presume the set $\mathcal{SA}$ to contain all speech acts. Let $\mathsf{Concept} : \mathcal{SA} \rightarrow \mathcal{CO}$ be a function providing the *decontextualised* concept conveyed by a speech act. This concept is almost inherently complex, in other words, composed of other concepts. In addition to this decontextualised concept, the context of the speech act must be provided. Let us presume that the context of a speech act is minimally captured by giving a time t and place p of the occurrence of a speech act in combination with its speaker s and hearer h[7]. The context of a speech act thus consists of some tuple $\langle t, p, s, h \rangle$. Formally, we presume the set $\mathcal{CN}$ to contain all possible contexts for speech acts, where the function $\mathsf{Context} : \mathcal{SA} \rightarrow \mathcal{CN}$ is assumed to provide the context for a specific speech act.

Consider the following example. The command "Stop!" being given by a certain speaker addressing a certain audience at a certain time and place is a speech act; the exact same command uttered by the same speaker ten seconds later is a *different* speech act, that may be interpreted differently. Note that we do not formally identify all separate aspects of a context as the precise description of particular contexts lies outside our scope. For a discussion of various aspects of context description see Section 3.2.8.

For a given language L we presume $A_L \subseteq \mathcal{SA}$ to be the set of speech acts that may be produced by means of that language. Next, we define set P_L of concepts in L representing speech acts:

$$P_L \equiv \{\mathsf{Concept}(a) | a \in A_L\}$$

We require each of the concepts used in the set of speech acts (belonging to a given language) to be part of the set of concepts of that specific language:

Axiom 1 $P_L \subseteq C_L$

We can now define the set of *utterances* U_L, that corresponds to the set of forms of concepts (taken from F_L) that can be associated with complete speech acts in language L:

$$U_L \equiv \{\mathsf{Form}(c) | c \in P_L\}$$

As a direct consequence of this definition and axiom 1, we have:

Corollary 1 $U_L \subseteq F_L$

[6] Note that we ignore discourse as a linguistic compositional level. We do this strictly for reasons of focus.

[7] This is a somewhat simplistic analysis of context, but it will do nicely for our current purposes.

In sum, a speech act $a \in A_L$ is expressed through a specific utterance $u \in U_L$. We now move on to combine our view on language with our view on information systems as medium systems.

Language alone is not enough to achieve communication. Besides language, a *medium* is needed, obvious examples being audio, video, writing, etc. More generally, we define a medium as a physical means to achieve communication. The ultimate, "mediumless" communication situation would require a full-scale 'mind melt' between agents. We assume any other situation to require some sort of physical system, either natural or engineered, to provide input to one or more of the senses.

We view a *medium system* as the combination of a medium and a language (potentially, a closed language). If a medium system is used for communication by means of a language, it is inherently used to perform speech acts. Formally, we presume the set $\mathcal{MS}$ to be the set of all medium systems. Furthermore, let $\mathcal{ME}$ be the set of all media. Next we presume the following functions to provide us with the language built into a medium system and the media used respectively:

$$\mathsf{Language} : \mathcal{MS} \rightarrow \mathcal{LA}$$

$$\mathsf{Medium} : \mathcal{MS} \rightarrow \mathcal{ME}$$

Thus, if S is a medium system, it is used for communication by means of language $\mathsf{Language}(S)$ via medium $\mathsf{Medium}(S)$. The notion of a medium system S, combining a language $\mathsf{Language}(S)$ and a medium $\mathsf{Medium}(S)$ into an actual means of communication, provides a more concrete instantiation of the more abstract notion of a combined medium and language.

We observe that people continuously influence the medium systems they use. As a part of this, people may make agreements about restricting the language they will use even if such restrictions are not reflected in the internals of some medium machine; for example as in the case communication by means of a strict and limited set of agreed commands. In many cases, however, people influence the medium system they use by introducing technology: purposely engineered physical phenomena. Let us call such a purposely engineered medium system a *medium machine*. In this respect, electronic computational devices are merely exponents of a rather advanced technology that may be used to realise medium systems.

We also observe that specification in one form or another is a crucial part of the process of 'purposely engineering a medium machine'. From this follows that ideally, when engineering a medium machine, both the medium aspect and the language aspect of a medium machine should be specified. In engineering it is common practise to distinguish between *requirements* of the artefact to be realised and the resulting artefact as such. With respect to the realisation of a medium machine we therefore presume there is some set of requirements R which reflects some ideal medium system S_R. After realising a medium machine conform to these requirements, we obtain a medium machine M. Such *conformity* entails the following matchings:

$$\begin{aligned}\mathsf{Language}(S_R) &= \mathsf{Language}(M)\\ \mathsf{Medium}(S_R) &= \mathsf{Medium}(M)\end{aligned}$$

In current information system development, both sets of requirements are recognised, but the view on language specification is usually one that does not sufficiently take into account the nature of language. This is mostly due to the fact that people involved in the specification process are biased by technical (medium oriented) interests and viewpoints. We will return to this point in Section 2.3.2.

In our perspective, a medium machine is strictly implemented in terms of (linguistic) *form*: the form part of the specified concepts, including the utterances that can be produced through the machine. The meaning part of a language can optionally be *described* somewhere in the system (for example, in an informative 'help window') but can never be *implemented* as part of the actual medium machine being *used* for communication. Any meaning description of a language form involves other language forms. In other words, all meaning representation is pure form (this point will be more elaborately clarified in Section 3.2.6).

Finally, note again that a speech act is created at a particular moment by a particular speaker, and that in principle it ends there; it is an *event*. However, it is possible that the medium allows for the related language representation to be *persistent*: to be available for some time (perhaps indefinitely) after production. This allows for the creation of *texts* (see Section 4.3.1). Texts are important to our account, but should not be confused with a closed specification of a language.

The above definitions mainly serve to single out $\mathsf{Language}(S_R)$ and distinguish its specification from the specification of $\mathsf{Language}(M)$. In this thesis we are primarily interested in the processes leading to specifications of $\mathsf{Language}(S_R)$. Within this focus, closed language primarily involves the restrictive specification – in a representational framework of choice – of the (required) conceptual 'bandwidth' of a medium system (as may be implemented as some medium machine M). The conceptual bandwidth of a medium system S is defined as $C_S \equiv C_{\mathsf{Language(S)}}$.

In sum: we are concerned with concept specification, in the context of closed language use in human-to-human communication implemented through a medium machine which actively restricts the utterances available in communication, the typical medium machine being an information system. We metaphorically refer to closed language communication as the *use of 'frozen language'*, and to the process of specifying the set of concepts of the language to be built into a medium machine as *'freezing language'*.

2.2 The Problem of Freezing Language in an ICT Context

Open language communication amounts to typical natural language use and even if we use ICT-related media, it should cause no trouble apart from any trouble that already comes natural with natural language use. And if we use closed language, either because we choose to or are technically forced to, then if we draw up our specifications with some care and with an open eye to the communicational functionality of the system, all is well. But is it? Let us look more closely at the differences between open language use as opposed to closed language use in case we use an information system as a medium system.

Concept	Explanation	Formal Definition
$\mathcal{CO}$	all decontextualised linguistic concepts	–
$C_L \subseteq \mathcal{CO}$	all linguistic concepts in language L	–
$\mathcal{MG}$	all linguistic concept meanings	–
$\mathcal{FO}$	all linguistic concept forms	–
$\mathsf{Form} : \mathcal{CO} \to \mathcal{FO}$	a linguistic concept has a form	–
$\mathsf{Meaning} : \mathcal{CO} \to \mathcal{MG}$	a linguistic concept has a meaning	–
F_L	all concept forms in language L	$\{\mathsf{Form}(c) \mid c \in C_L\}$
G_L	all concept meanings in language L	$\{\mathsf{Meaning}(c) \mid c \in C_L\}$
$\mathcal{SA}$	all possible speech acts	–
$\mathcal{CN}$	all possible contexts for speech acts	–
$\mathsf{Concept} : \mathcal{SA} \to \mathcal{CO}$	a speech act has a decontextualised concept that it conveys	–
$\mathsf{Context} : \mathcal{SA} \to \mathcal{CN}$	a speech act has a context	–
$A_L \subseteq \mathcal{SA}$	all speech acts that may be produced through language L	–
P_L	all concepts that may represent a complete speech act in language L	$\{\mathsf{Concept}(a) \mid a \in A_L\}$
U_L	all forms of concepts (*utterances*) that may express complete speech acts in L	$\{\mathsf{Form}(c) \mid c \in P_L\}$
$\mathcal{MS}$	all medium systems	–
$\mathcal{ME}$	all media	–
$\mathsf{Language} : \mathcal{MS} \to \mathcal{LA}$	all medium systems have an associated language	–
$\mathsf{Medium} : \mathcal{MS} \to \mathcal{ME}$	all medium systems have an associated medium	–
C_S	conceptual bandwidth of a medium system	$C_{\mathsf{Language(S)}}$

Table 2.1: Overview of conceptual framework

2.2.1 Language Diversity and Adaptability

A basic claim we lay down is that indeed there a significant difference between free-flowing natural language as we encounter and use it in 'open' spoken or written communication, and the use of bits of language glued tightly into the mechanism of an information system. The difference lies for a large part in the natural capacity of human beings to adapt their language use to occurring situations as they go. This we call the *adaptability* of language. Typically, natural language is adapted to the usage situation *intuitively*. Situational adaptability lies not only with the language producing party, but also with what concepts the receiving (interpreting) party believes the producing party might be using. Now if a *closed* language medium is used in communication (and to keep the discussion transparent, let us assume the medium is the *only* means of communication), the adaptability of the language as built into the medium system determines the adaptability of the available language system at large.

In some cases, the intuitive ability of people to use and interpret language flexibly does not suffice, and a misunderstanding or communicational breakdown may arise because the *language used cannot get the job done*. In such a case, explicit conversation may be engaged concerning the language used, for example: "eh... what do you mean with X?", followed by some exchange aiming at the *repair* of conversation: clarifying language use. Such communication-about-communication we call *linguistic meta-communication at speech act level*, as it focuses on the meaning in context (i.e. both semantic and pragmatic) of a particular speech act. In natural language, meta-communication[8] is an integrated part of the language capacity and 'built into' language use. Importantly, *speech act-level meta-communication can compensate for lack of situational adaptability.*

In addition, it is sometimes useful for the communicating parties to communicate about the general language system that is to be used in conversations between them. This also involves meta-communication, but since it concerns *agreement about language* instead of the clarification of particular utterances produced *by means* of some language system, we call it *linguistic meta-communication at language level.* Such meta-communication concerns rules and agreements (conventions) about language to be used, as may or may not be made explicit in some form of documentation (for example, a dictionary).

Let us make clearer what exactly we mean with linguistic meta-communication[9]. In speaking and writing, people take into account the immediate context of language use as well as what they believe their audience knows, and how it will interpret what is said. Interpreting language that we hear or read may seem simpler than producing it, but in fact it can be argued that it is at least as difficult: from all the possible meanings of a sentence, including details of word meaning, the interpreting party

[8]Alternative meanings of the term are sometimes used. One is "communication not relayed by words but by other signs accompanying the verbal message" (for example, facial expressions). We prefer to call this *para-communication.* Another, related sense of the term is "that part of the message that tells us how to interpret the rest of the message". This sense comes closer to our own, but as it focuses on the message as such it still fundamentally differs from it. We strictly refer to *separately engaged, distinguishable acts of communication about communication.*

[9]We will present a more in-depth theoretical discussion of meta-communication in section 3.4, relating it in particular to theory by Habermas

must select the meaning that makes most sense. In order to do this, she needs both *linguistic knowledge*, such as grammatical rules and conventional word meaning, and *contextual knowledge*, ranging from the position and state of particular objects referred to in conversation to highly implicit emotional and cultural matters.

Meta-communication as focused on here (both at language level and speech act level) thus strictly concerns conversations about *language and language use*, not about more general contextual knowledge or information. In other words, we refer to *linguistic* meta-communication, involving linguistic knowledge, and only marginally with meta-communication about knowledge of the world. However, it has to be mentioned here that in many cases separation of linguistic knowledge (especially semantic and pragmatic knowledge) and 'general' knowledge is no trivial matter. For more on this see Section 3.2.8.

In addition to the notions introduced so far (adaptability, meta-communication), the *links* between these concepts are also important in our comparison of 'open' and 'closed' language use. In natural (open) language, there is an inherent link between speech act level and language level meta-communication, since discussion about language as it is used in a particular situation may instantly lead to a changed (intuitive) opinion of what the shared language system should look like. Also, if meta-communication triggers a change in opinion concerning what constitutes the appropriate language system, this will intuitively lead to actual adaptation of the language system: the people involved will intuitively learn from their meta-communication. Consequently, in 'free' or 'open' natural language use, *adaptation of the language system to a particular situation will only rarely need to involve language level meta-communication, and actual adaptation will take place 'automatically'.*

We now have sketched five properties of natural language that are potentially (in fact, frequently) missing when it comes to closed language built into information systems:

1. adaptability
2. speech act level meta-communication
3. language level meta-communication
4. coupling of speech act level and language level meta-communication
5. coupling of language level meta-communication and actual adaptation

As a result, communication may be hampered in two main ways (resulting from the unavailability of some property, or of several properties):

Expressive power may fall short. The linguistic means available may not be optimal in view of what needs to be said, what it needs to be said for, and who it is said to. *Missing properties 1, 3, 4, 5, or combinations of these*

Miscommunications may remain unresolved at speech act level. If a miscommunication occurs, and repair through speech act level meta-communication is unavailable, the communicational trouble cannot be solved quickly and easily. *Missing property 2*

In addition, another, related problem may occur:

Contextual interpretation may fail because of insufficient information
A listener or reader needs contextual information to optimally interpret the speech act. Lack of such information may hamper successful interpretation.

The latter problem does not concern the difference between open or closed language as such, but is a side effect of the potential abstraction of time, place, and speaker/hearer inherent in the use of many types of medium (for example, letters, telegraph messages, forms, databases). Nevertheless, it is quite relevant to our account, because it is a likely cause of the kind of trouble reflected by the "falling short of expressive power".

Let us elaborate on the above by providing some examples. Since the factors discussed are strongly interrelated, we discuss them in an integrated fashion. Please note that the examples serve to demonstrate the sorts of problem that may occur and various ICT contexts that may be involved, not to illustrate the urgency of the frozen language issue (for that, see Section 2.2.4).

Example 1. It is quite possible that given the limited language available, the 'speaker' cannot find the exact right words to express herself, and on the other hand, it is possible that the linguistic and contextual knowledge needed for in-depth interpretation is insufficiently available on the receiving side, causing some interpretation trouble. The resulting loss of accuracy and/or richness of meaning is not necessarily a problem in every case, but caution is needed, especially if communication is complex, concerns an intricate context, and requires precise interpretation.

To give a concrete example, if a typical web search engine (for example, Google) is used to recover a document from a library, and this search engine attempts to match a query representation against an index consisting of keywords, the person formulating the query is should "correctly" use some terms to formulate the index. This may present the querying party with a conceptual puzzle: find the right terms (for example, select not "Spanish Flue" but "Influenza A/H1N1"), and also establish that they actually mean to the indexer what they mean to the query formulator (which is what is meant by "correct use").

This example is related to well known issues information retrieval, as described in, for example, (Cleverdon, 1984), (Blair and Maron, 1985), and (Schatz et al., 1996). In terms of 'missing properties' it relates to both lack of expressive power and lack of contextual knowledge.

Example 2. This example also concerns lack of expressive power. If in a workflow support system, action code "4711" stands for "client status is in the process of being reviewed", but there are in fact two relevant kinds of reviewing process possible (quick batch processing and slow "exception handling"), then expressiveness is insufficient (there is only one code where two are required), but in an inflexible medium system the concepts available cannot be extended even if all communicating parties agree this would be better.

Example 3. Speech act level meta-communication is not generally supported by information systems:

If two colleagues work with a central database, and one enters a "code 4711" in the database field "client status", then if the other person does not know what this means, there is need for extra communication, about what code 4711 means. This cannot take place through the system: it requires some other way of communicating. Now perhaps one can engage in such a conversation about language by picking up the phone or sending an e-mail, but this only goes to show that an entirely different *open* communication channel has to be deployed in order to cope. This is fine, but what if such an open-communication backup is not available? Worse, what if the 'speaking party' is unknown (forms, especially electronic ones, may go through many anonymous –virtual– hands) or has left the organisation five years earlier? Databases may store utterances for a considerable time, adding to the problem. And what if there is no identifiable 'speaking' person as such involved[10], and there is only this chunk of data (re)produced by a machine? Interpretation either becomes impossible altogether or, perhaps more dangerously, the human instinct for making sense *no matter what* produces some meaning anyway, which may be so different from the one intended that damage is done. (In this example, lack of speech act level meta-communication is combined with lack of contextual knowledge).

Example 4. A typical contemporary information system does not usually allow users to adapt language to their situation at run-time: the concepts are linked up to the design and are not flexible. For example, depending on the type of user involved in a query process, one might like to make available either *"Spanish Flue"* (layman) or *Influenza A/H1N1* (medical expert). Most current information systems are unable to linguistically respond to (or even recognise) different types of users. (This example concerns lack of expressive power related to insufficient adaptability, and also lack of contextual information).

Example 5. If the possibility for language level meta-communication is absent, then even if communication can be repaired temporarily by meta-communication, next time the same concepts are used, the same trouble may occur; as the use context of the medium system changes, the trouble will only increase. And even if necessary changes can be implemented in principle, the *effort* of constantly specifying and updating the medium system may be just too much (combination of the lack of language level meta-communication and the coupling thereof with adaptive action).

We trust that by now it is sufficiently clear what we mean with frozen language in context of information systems, and what the basic problems are that it may cause.

2.2.2 Why Sub-Optimal Conceptual Functionality May be Condoned

Exponents of the problems presented thus far are commonplace if one looks for them in ICT-supported organisations, if not in all systematically organised environments. Speculatively, there may be various explanations why few people seem to care much about them.

First, people are prepared to put up with an awful lot when it comes to working within organised, ICT-supported environments, and consider this normal. Frustration about communications is, unfortunately, part of working life.

Second, thanks to the robustness of human language/thinking capacity, people are

[10]With respect to 'depersonalised' communication, it is interesting to consider the properties of its counterpart: 'face-to-face' communication. See (Agerfalk, 2003; Clark, 1996).

very good at actually coping with difficulties involving expression and interpretation, so many potential problems do not actually occur as such (no thanks to the information system). This is mostly due to the vast knowledge of their working situation that people can acquire, through extensive training or (more commonly) through plain working experience.
Third, information systems are not (yet) much used in situations where the communicational context is complex enough for closed language use to cause trouble at a scale that priority-wise puts it up there with the Millennium bug.
In addition, it could be argued (as some have) that meta-linguistics in human-human communication lies too far outside the common focus of the people who busy themselves thinking about ICT: their perspective is primarily technical, leaving little or no room for in-depth discussion or even awareness of intricate matters involving language and meaning. Even if all stakeholders involved in a domain actively validate the frozen domain language proposed, they will be unaware of the finer linguistic points of the matter. Clearly, we advocate a more linguistic perspective on domain analysis, but on the other hand we cannot but acknowledge ICT developers are not, and never will be, linguists. Therefore, we look for improvements that somehow harnas the *natural* capacity of people to talk about and improve the language they use in their daily activities and communication.

2.2.3 Why People Freeze Language

Having discussed some fundamental aspects of perceived trouble with freezing language, let us now view the same issue from an opposite, more positive perspective. After all, freezing language is not necessarily something imposed on us by powers beyond our control: people do it, and they do it more or less intentionally. So why would we want to introduce closed language communication in the first place?

Conceptual Certainty and Efficiency in Communication

First of all, let us assume that people generally prefer to do as little as possible, or perhaps more accurately, they want to do certain things and as much as possible keep to them without doing other, supporting things that do not belong to the primary activity of choice. With respect to secondary activities, they strive to be *efficient*, i.e. put in as little effort as possible. This holds for communication and languaging just as it does for other activities. While talking in itself may be a goal if it is considered enjoyable or sociable, it may in some situations be considered a bit of a burden, especially if it takes on a purely instrumental role. *Communicational effort* could be measured, for example, in the number of words required, or in the time spent to utter them.
The most direct way to minimise communication is to simply cut it away. This means that all agents involved should optimally know the situation and directly respond to what they *observe*, without any coordinative or communicative action being needed. Another time-honoured way to minimise communication, and the one most relevant to us, is to use a limited set of words or other symbols. This reduces communication to the exchange of a few optimally relevant concepts that everyone involved knows, and trough which operations can be coordinated with minimal effort.

Also, if people engage in some activity, they are typically keen on being certain that what they try to do will in fact work. In other words, they strive for *certainty.* It could be argued that cutting away communication also cuts communicational uncertainty. While this certainly makes some sense, it can alternatively be argued that communication enables back-up coordination and the sharing of situational information, and thus increasing *operational* certainty. And to complete the picture, *meta*-communication can help improve *communicational* certainty. In other words, if it is certainty we want, cutting away communication is not necessarily the best way to go. Thus we are faced with the likely need for a trade-off between efficiency and certainty in communication, especially when seen as part of a larger set of activities. We therefore believe that *communicational certainty can best be seen as essentially distinct from, and orthogonal to, communicational efficiency.*

Given that some communication does occur, the use of a restricted vocabulary is also a standard approach to achieve certainty, but now the emphasis lies on the concepts (typically, vocabulary) being *well known and well understood by everyone involved.* If the concepts involved are not transferred implicitly (through experience) but *explicitly*, this is normally achieved by listing all concepts to be used in some situation and describing what they should mean, perhaps after first reaching agreement on this. Normally, the fewer words are listed, the easier we can make sure that their meaning is in fact sufficiently clear for all involved. Here, efficiency in *meta-communication* inspires the use of a limited set of words.

Note that the trade-off between efficiency and certainty does not *necessarily* imply that an increase in one means a decrease in the other. However, it will often be the case that the use of a relatively inefficient ('long') utterance will be relatively certain in terms of its ultimate interpretation, and that a short and efficient utterance requires a very knowledgeable interpreting party for it to be communicationally certain.

In sum, the combined tendencies to strive for *efficient* and *certain* communication encourage, and in some cases *require* the use of closed sets of well-understood concepts. Now let us try and be more precise about what we mean with efficiency and certainty.

Tentative Definition of Conceptual Efficiency and Certainty

Efficiency of communication might be expressed (or hypothetically even measured) in relation to some benchmark, for example the number of explicit sememes (elementary expressions of concepts) minimally needed to effectively but *completely* express a certain message (speech act), in context. The benchmark might, for example, be set by a layman, i.e. someone with no more than 'common knowledge' of the situation at hand and the language used to communicate about it. Ruling out zero communication, optimal efficiency for any speech act then is achieved if 1 such sememe is needed. Actual efficiency thus depends on the complexity of a particular message in 'common language'.

$$\text{efficiency} \triangleq 1 - \frac{\#\ \text{actually used sememes}}{\#\ \text{sememes needed in common language}}$$

For example, the message "Captain orders engineer to immediately stop engine" (let

us say this utterance consists of 9 sememes: *captain-order-s-engineer-to-immediate-ly-stop-engine*) might be conveyed by one "STOP" command (1 sememe). The efficiency of C_l for this particular speech act would then be 1-(1/9) = 0,889, which is reasonably good; efficiency equal to the benchmark would be 0, very high efficiency approaches 1.
Certainty[11] might be expressed in terms of the number of interpretations of some message (in context) as intended by the speaker, related to the number of interpretations ultimately arrived at by the hearer. 100% certainty should be represented by 1. Differences in interpretation are counted as such only if they are relevant in the particular context.

$$\text{certainty} \triangleq \frac{\#\ \text{intended interpretations}}{\#\ \text{actual interpretations}}$$

For example, a (rather dim) deck hand may be uncertain about what a "STOP" command might mean:

- "Captain orders: immediately stop the ship"
- "Captain orders: immediately stop the engine"
- "Captain orders: immediately stop what you are doing"

The certainty score in such a case would simply be 1/3=0,333. In case the number of interpretations goes up much more, quantification becomes rather useless, but the point is that in such cases, certainty approaches 0.
Please note that we *do not suggest actual use of our simple and tentative definitions in calculations.* We include them strictly in an attempt to carefully express what we mean with the notions in question.
Possibly, efficiency counts should include the number of concepts needed in meta-communication at speech act level. Thus, it may well be the case that by engaging in meta-communication, two communicating parties can successfully raise certainty to 1 while in the mean time seriously damaging efficiency. Whether or not such a 'linguistic investment' would be acceptable depends completely on the situation in which this occurs.
Earlier in this chapter, we have referred to the notion of "expressive power" (Section 2.2). Note that expressive power might in fact be expressed as a combination of the notions "efficiency" and "certainty". So in our example, the expressive power of the "STOP" command in a context involving our clumsy deck hand as a listener as would be '0,889 and 0,333'. Maximal expressive power is thus represented as '1 and 1'. However, note that the tentative apparatus introduced above strictly refers to *single speech acts*; to extend the definition to the expressiveness of *a* language, we would have to take it as a norm over the expressiveness of a representative number of speech acts (see Section 2.1.2). For example: "90% of speech acts score 1 and 1".

[11] Some readers may be reminded here of the ideas of Shannon on certainty (Shannon and Weaver, 1949). Indeed one might see our tentative definition as an attempt to import his notion of certainty into the domain of interpretation –yet without the elaborate mathematical underpinnings; our definition is a mere suggestion, Shannon conceived a well-founded scientific theory.

A standard example of a situation that is both highly communication-efficient and communication-certain is the stereotypical interaction between a surgeon and his team, who focus on the medical procedures and keep communication to a minimum[12]. This can only be achieved in a situation where everyone knows pretty much what can be expected to happen on the basis of both their knowledge of similar situations , and what they see (no 'clumsy deck hands'). Only minor, well-understood communications are then required: brief expressions like "scalpel" and "OK, close it up" will do in regular circumstances.

The principles of efficiency and certainty presented so far can be observed to operate across all work-supporting communication that takes place in a well-understood, sufficiently predictable environment. Note, however, that if *social* factors come into play, or if communication concerns complex and unpredictable situations, other principles apply, usually demanding open instead of closed communication. In such cases, the use of the above definitions of efficiency and certainty becomes trivial: people's natural feeling for language use will take over, and to evaluate open communication in terms of efficiency and certainty is thus irrelevant to our purpose.

Technology-related Motivation for Freezing Language

There is a third reason why a limited, well-specified set of concepts may be enforced in a communicational situation: it may be required to do so for *technology related* reasons. There are two related arguments concerning the use of conceptual representations as parts of engineered machines:

1. machines need to be specified and put together from known parts
2. since it takes time and effort to specify and implement these parts, it is required that their number is limited

We suggest that in mechanical devices designed to aid communication, the words built into them can be seen as a peculiar type of 'parts'. For example, consider a ship's telegraph (Figure 2.2). The words on its dial (e.g. "STOP" or "FULL AHEAD") are a crucial part of the machine's workings largely determine its functionality.

In the twentieth century, a novel factor was introduced to the design of symbol-wielding machinery: automated symbol processing. Machines started to take over some of the actions previously performed by humans, and this involved automation of symbol-based communication at several levels. Sometimes communication was the primary process being automated, but in many cases it came along as an auxiliary mechanism. In numerous situations, the computer-based media did not make use of open communication, and therefore restricted sets of concepts had to be introduced on a large scale.

As mentioned before, an important case in point is data structure. Databases and other 'data wielding systems' require explicit specification of their conceptual structure. Such a structure should include a limited number of conceptual (linguistic) elements, and for practical reasons, the specification must be limited. The use of

[12] Even if they are, at the same time, chatting away about non-medical subjects, as we are reliably informed is quite common. Scenes from the classic TV series 'M.A.S.H' also spring to mind here.

Figure 2.2: A ship's telegraph

limited sets of concepts thus follows from the very nature of computational machinery.

The three main arguments for freezing language presented above (efficiency-related, certainty-related, and technology-related) are heavily intertwined in many cases of frozen language use, especially when ICT is involved, and it is often quite impossible to decide which of the three is the leading incentive to use frozen language. We will refer to the combined arguments as 'the practical factor'. This factor is illustrated by the following example.

If a medium functions through a limited set of states, and if such states are somehow used as or linked with symbols, this will by nature limit the number of symbols that can be transmitted. The ship's telegraph, for example, can show a limited number of symbolic states, involving actual depicted words which come from a known language (English). It has two interlinked pointing devices that single out one concept at the time, thus communicating between the bridge and the ship's engineers what the current state of the ship's engine should be.

At least for some part, the success of the ships' telegraph appears to be related to the simplicity of its conceptual load. Whereas implementation of a similar machine wielding a hundred or so concepts seems possible, the communicational situation in which a ship's telegraph is deployed seems both to allow and to demand a more limited set of concepts in order to be useful. Whether these demands concern matters of conceptual efficiency, conceptual certainty, techno-conceptual simplicity, or a combination of two or three of these factors is impossible to say.

As a further illustration, note that a ship's telegraph, given its context of use, could not have functioned as well had it shown the twenty-six letters of our alphabet, since this would require people to signal and take down messages letter-by-letter, which is cumbersome, slow, and may not be sufficiently persistent and robust. One could not at any time and under quite unhelpful circumstances (say, in a hurricane) glance at the telegraph and instantly infer the current (intended) state of the engine[13]. The ship's telegraph is designed for a specific, conceptually stable context that requires a simple and robust signaling device.

It may appear strange to compare advanced ICT-based communication (e.g. electronic 'business to business' communication, federated and distributed database schemes, etc.) with a ship's telegraph, but as far as the selection, definition, and use of restricted languages is concerned, all these examples share certain basic properties. *They are subject to mechanisms and factors that involve and require the freezing of language.*

Note once more that our focus excludes the fascinating world of symbol *processing*, i.e. those mechanical manipulations in which one or more symbols are replaced by one or more others according to a certain algorithm. We refrain from trying to explain what happens to the meaning of symbols as they are processed, and restrict our account to 'input' and 'output' symbols as such. However, our framework may eventually also help shed some light on matters related to the semantics and pragmatics of mechanical symbol processing (see Section 8.2).

In view of the above, let us recapitulate what we mean with "freezing language": *the meta-communication process in which a set of selected concepts, belonging to some language, is specified in order to be built into some medium machine, thereby limiting the concepts (elementary or complex) that this machine can convey.* The *definition* of the meanings of the concepts is optional.

Importantly, there is a difference between the freezing of language and the *use* of frozen language: freezing language is a meta-communication process that stands apart from the use of its *product*: frozen language. Frozen language is to be used in a regular, 'operational' communication situation. In addition, freezing language is *not* the same as fixing language *utterances*: this results in *texts*.

The use of frozen language does not exclusively play a role in the context of information and communication systems. As we have seen it also occurs in a ship's telegraph. Indeed, the phenomenon of the freezing of language is not restricted to ICT-supported domains, but we do apply this extra restriction to the focus of this thesis. Why we do this will be made clear in the next section.

2.2.4 The Importance of the Frozen Language Issue

Having sketched the basic drive behind the creation and use of frozen language, and how this generally takes place, the question presents itself how urgent the subject of 'freezing language' is in view of general ICT development and practice. Our communication-oriented approach may strike the reader as quite academical, and as presented in this thesis, it doubtlessly is. However, we have reason to expect that

[13]Note that typically, a ship's telegraph is complemented by a "speaking pipe", which enables open communication but is much less robust as a medium.

as a more practical problem it will acquire an increasing urgency and visibility as the 'digitisation' of work and society progresses.
Information and communication systems owe their usefulness for a large part to their functionality as a medium, apart from their (perhaps more prominent) computational/symbol manipulation functionality. The selection and description of closed languages evidently plays some role in the specification of engineered support mechanisms for efficient and certain communication (Wright, 2001c). Yet how crucial is the issue, and how big an impediment will related problems eventually be to the global ICT effort?
We believe it will become a very serious impediment indeed, especially if we hope for ICT to become more 'hidden' and user friendly. There is a clear move towards the development of "information appliances" (Norman, 1998): embedded ICT that moves away from PC-like devices and into small, dedicated tools[14]. Such devices will have to fit their conceptual context of use closely, or else be judged user unfriendly or downright dysfunctional.
Perhaps more importantly, major forces like industry, public authority, and science are unfolding and executing many exciting plans that, if realised, will inevitably entail large-scale freezing of language. Virtual enterprises, e-commerce, web-services, agent-based systems, inter-organisational workflow, distributed and federated information systems: their development and maintenance will necessarily and increasingly involve the freezing of language. The availability of global public networking (the Internet) has of course been a crucial impulse for these activities. In this vein, a quite relevant though rare statement comes from the field of 'terminology management':

> "[...] Information does not just flow through database management systems like water in a pipe. It is carried by discrete, identifiable, retrievable vehicles called *data elements*. These data elements are rendered identifiable in that they have *names*, and the naming of data elements in information systems is an essentially terminological act. In fact, the harmonisation of data element names and data structures is one of the most critical challenges facing information specialists today in their effort to create global systems that can "talk to each other" across hardware and software barriers. Consequently, the link between terminology management, information management, and quality management is one of intimate interdependence" (From: Wright and Budin, 1997, *Handbook of Terminology Management*, vol. I, pp.2-3).

The challenge of freezing language further extends to related concept-oriented fields such as information retrieval, knowledge engineering, and language technology, though these fields have their own specific priorities and challenges.
As more, and more different, application domains are interconnected, conceptual differences between them will increasingly be reflected in the (in)compatibility of concepts used to communicate in and between those domains, but also in the *substantial increase in the number of specification tasks*. Designing optimally functional

[14]It is traditional to exemplify the "information appliance" through reference to that common household device named the Toaster, a tradition that we can but honour.

information and communication systems may not be impossible in principle, but many efforts will bog down because of *limited specification capacity*: not enough analysts, informants, programmers, time, money, and good will to keep up with the ever increasing stream of specification and re-specification jobs. Conceptual specification (as seen apart from 'technical specification') will represent an important part of the specification tasks at hand.

In the field of ICT architecture and design, the idea has emerged that perhaps the typical model of centralised specification of systems, and their incremental adjustment (releases), may not be the only option. An alternative view that is being explored concerns *evolutionary systems* (Proper, 1994). In this view, the need for change no longer is a necessary but cumbersome property of information systems management: the ability to change should be a natural and built-in aspect, and be part of the very nature of information systems. Language specification will, however, still be part of the deal.

Much research inspired by the evolutionary systems concept involves automated approaches to system specification, using artificial intelligence techniques. We believe this approach to be promising in the long run, but it will be a rather long wait until ICT is fully self-specifying. And if not automatically derived, specifications will still have to be delivered by humans: by analysts, designers, or users.

Standardisation (Strehlow, 1997a; ISO/IEC, 1995) will also take away some of the pain. However, initial experiments and projects depending heavily on the use of frozen language (for example, EDI-related efforts[15]) have run into problems involving rigidness of the standardised communication media created (Ekering, 2000, pp.89-95), and more recent initiatives (Berners-Lee et al., 2001; Kotok, 2002) introduce increasingly advanced solutions that, however, seem not so much to try and avoid the freezing of language, but to find better ways of freezing language and of using and combining frozen languages.

But even if sufficient resources are available to specify one frozen language communication system, there is always the changing context, arguably being propelled towards the diverse rather than the uniform. A substantial part of specification will always lag behind; some standards will always fall short of their mark. And therefore, specify we will.

Freezing language is not an isolated phenomenon: it is of course tightly connected with a general effort to standardise, streamline, and make more efficient and certain many processes in human society. We are confronted with a fundamental tension between the complex, unpredictable, *ad hoc* aspects of human activity and our desire (and need) to analyse, organise, simplify, systemise, and mechanise it. If ICT is involved (with emphasis on the C), the freezing of language seems to be inevitably part of this. What is more, ICT now is the *crucial* enabler of the structuring of information, and also of the improvement of efficiency and certainty of action. Perhaps the biggest challenge then is to find in ICT a balance (maybe 'compromise' is a better term) between rigidness and flexibility, openness and standardisation, the needs of the individual and the needs of the community. We believe the issue of (the specification of) closed sets of symbols in ICT to be important in this light, quite important enough to justify serious research on the particular subject.

[15] Electronic Data Interchange

2.3 Current Situation: How People Freeze Language in ICT

Quite clearly, the challenges related to the freezing of language are far from new, and neither are attempts to cope with them. The question is how far the current understanding of the problems goes, and how well guided the solutions are. Let us recapitulate the basic problems:

1. Expressive power may fall short
2. Miscommunications may remain unresolved at speech act level

Revisiting our earlier analysis, two basic problem solving strategies can be used for problem 1: avoid the use of frozen language altogether, or freeze language carefully and perhaps cover for its ensuing weaker points through meta-communication (but see problem 2).
Avoiding frozen language use can be achieved by enabling open communication, or avoid communication altogether. The latter may be achieved through automation ('removing' the task from human-executed operations), or by making sure people know their job so well that no communication is needed. We disregard the abandoning of the task at hand, as it is not a 'solution'. Approaches that circumvent problematic *closed language* communication are important and even recommendable. Even so, there will be cases where the freezing of language is both necessary and problematic. In such cases, there are three directions to consider:

- First, we have to ensure that the set of concepts made available is sufficiently expressive, and that sufficient knowledge of it exists within the domain of use (directly tackling problem 1).
- If this fails, we can enable effective *clarifying* (meta-)communication of some sort (if not, we have problem 2).
- Alternatively, we can attempt to *adapt* the system as conceptual needs emerge –which also requires meta-communication, yet of a different and more *intervening* type. This confronts us with the adaptability-related factors underlying problem 1 (Section 2.2).

We will return to these combined solution directions in Chapter 4.

2.3.1 Practices and Support of Freezing Language

We will now discuss how our ideas about language as a functional factor in information systems fit with current practice in information and communication system design and management, and refer to some general approaches to the specific support of the freezing of language and of working with resulting concepts. We further discard avoidance strategies. Reference will be made to various fields in research and industry that typically deal with some listed aspect, illustrating the interdisciplinary nature of our enterprise. More elaborate discussion of various disciplines that feature conceptualisation processes, including more references, can be found in Appendix A.

Conceptual Modelling Reflecting Language

As computers started to invade the administrative domain, concepts related to the intended communicational functionality of systems needed to be analysed and represented. The semi-independent discipline of *information analysis* evolved, mostly in the form of the introduction of representational languages, instantiations of which could be mapped onto implementation-oriented structures. This required formal or at least semi-formal notation. To this day, many information analysis methods are inspired by first order logic. Often, languages for conceptual modelling include diagrammatic representations (for example, Entity-Relationship diagrams, ORM, the UML). Also, mappings of models to natural language expressions and exploration of resemblances between natural language and modelling languages (Hoppenbrouwers et al., 1997) have always been a part of conceptual modelling. A large number of representational languages for information analysis have been specifically designed for use in database design and engineering. This branch is particularly interesting within our focus, since databases are among the primary examples of closed language communication devices in ICT.

CASE Tools

Along with the many different languages for conceptual modelling that have been developed came the need for tools that helped use them. The most typical exponent of this is a type of dedicated software called a CASE[16]-tool, which in its most advanced form can be used to generate executable code. However, many CASE-tools fall short of this mark, and merely support the creation, (re)presentation, and storage of expressions in some representational format, and possibly some mapping (translation) from one such format to another. In addition, some CASE tools support the verification of correct use of a particular formalism, or even some advanced reasoning about represented structures.

Combining and Translating Conceptual Frameworks

Another relevant development concerns attempts to cope with the conceptual problems that arise when, for example, two or more database systems *with different data structures* are combined (for example, one might want to access them as if they were one database). In areas like the field of distributed and federated databases, much interest has been raised by such conceptual matching problems (Papazoglou and Schlageter, 1998). The problems encountered are mostly perceived as technical, but they are also related to matters of multiple domain information analysis, and therefore with freezing language. One particularly popular solution to the problem of concept matching concerns the use of *ontologies* (defining 'ontology' for the moment as "a specification of a conceptualisation"; Gruber, 1995). The use of ontologies is not limited to the field of database engineering; it has roots in mathematics, philosophy, and artificial intelligence.

If discrepancies between conceptual structures need to be 'solved', this might be done by careful construction of a 'one-to-one mapping': a literal translation. If realistically possible, this is a reasonably good solution except for the substantial effort

[16] Computer Aided System Engineering

it takes plus the fact that such an effort is needed for every new link between two different database schemas that occurs. Though 1:1 mapping is usually strived for, it may be that only partial translation is achieved, and this may or may not suffice. Perhaps the most important point here is that once two different conceptual frameworks meet, they can no longer be discussed 'in terms of themselves': there arises a need for a common expression of meaning through some sort of meta-language, or at least a meta-conceptual framework, for example an ontology.

Ontologies are tools that may help us to cope with the combining of different conceptual systems (Berners-Lee et al., 2001). Such systems do not only include different sets of symbols, they are also likely to have differing *semantics*. Matters of semantics have always played some role in information system design. Formal notations have often been used to describe system-internal (operational) semantics, and this continues to be the case. However, in many cases *cognitive semantics* need to be considered as well: what words (which are but arbitrary labels to the system) mean to *humans*. This calls for a description of concepts in terms of human activity and interaction. In this vein, various approaches have been suggested in fields like Database Engineering, Knowledge Engineering, and Computer Supported Cooperative Work (CSCW). However, they rarely go beyond the observation that *user participation* in various forms is vital to the design process, and that language built into information systems should as much as possible approach the 'natural' language of the user. What the wider linguistic implications of such approaches are is, to our knowledge, hardly subject to discussion.

Dedicated Concept Repositories

Ontologies, dictionaries, lexicons: we will refrain from discussing the differences between them here. It suffices to say they all serve to represent words and (usually) their meaning, sometimes by listing them and relations between them, sometimes by specifying properties of concepts by means of other concepts. Intricate formalisms or specialised semantic representations may be used, but also simple descriptions in a natural language like English. An established phenomenon in database and system design is the data dictionary. It is generally a dictionary describing the meaning of the labels that are included in the data structure (apart from items like name, type, values, source, and authorisation for access for each data element; see Wright, 2001c). However, descriptions of database concepts and their meaning may also be found in manuals or other system-related documentation.

It will come as no surprise that any artefact explicitly listing and defining concepts for the specific purpose of human-oriented specification and clarification lies in the centre of our focus. However, in our perspective, such representations are mere tools. We need to keep in mind that what is of even more immediate concern to us are the *processes* through which such tools are formed, used, and manipulated, and the reasons and goals behind such processes.

Thesauri and Information Retrieval

Another relevant use of a particular kind of conceptual representation artefact, the 'thesaurus', is widespread in the field of *library science* (sometimes, confusingly, called *information science*). Thesauri as viewed in this field are, roughly, networks of categories and sub-categories (and so on) representing *topics* covered by some collection of documents. Thesauri were around long before ICT appeared in the libraries, but they have now become an important object of study in the field of *digital libraries*. Not only is a thesaurus a reflection of what its compiler considered to be a 'useful' system of categories, it also raises the question whether such 'useful' categorisation is the same for every user and every situation. The conceptual problems concerning both the construction of a thesaurus and its matching with or translation to another thesaurus are therefore tightly related to our focus.

Similar but perhaps greater challenges exist in the field of information retrieval at large: any searching of a document through the matching of some query representation with a document characterisation from a collection of many such characterisations. If the collection is not neatly controlled, indexed, and managed but is a more or less chaotic, giant collection of databases (we are of course referring to the World Wide Web), the conceptual problems encountered are vast. Without considerable computational effort and/or the availability of source-specific information, there is no telling what some term means to whom, and in which context it originated. Again the use of ontologies to solve this problem has been suggested and explored, with varying success. Let us merely observe here that no real solution is in sight so far. Among the chief problems faced, there is again that of the *specification* of ontologies: who will do it, and how? An answer that many researchers work on is *automation* of specification.

Advanced Concept-based Computation

This brings us to the final, and in the view of many researchers ultimate solution to conceptual problems (including the problems related to freezing language): automate. Have intelligent, learning machines sort it all out. Indeed, this could be the answer. However, before it becomes a real answer, science will have to solve an enormously complex problem that has occupied thousands and thousands or researchers and developers for decades without any real breakthroughs (though not without some progress).

How to deal with the complexities, ambiguities, flexibility, and fuzziness of 'human' meaning in computation and ICT is still largely an open question. Our focus involves these matters only indirectly, since we are primarily concerned with the selection and definition of closed sets of concepts, not with what happens if such concepts are manipulated by a (computational) system that, for example, changes symbols according to some program. Nevertheless, in the long run the fields of human-human communication and computational linguistics share a number of problems. Specification of knowledge, linguistic or other, is important to both of them, as is meta-communication.

To name just a few fields involving computational linguistics: automatic indexing, information extraction, automatic translation, automatic summarisation. We will refrain from further discussion of these general fields, but some related approaches and techniques are discussed in appendices A and B.

2.3.2 Some General Observations Concerning the Current Situation

As we have seen, the need for careful specification of concepts for use in closed language medium machines goes back decades and has led to the development and use of many intricate representational languages and tools in ICT. In addition, in various fields of research (e.g. advanced database and information system design, CSCW) the necessity of introducing user-oriented and even differentiated vocabularies has been acknowledged. This has in general led to more emphasis being put on the structuring and validation of the language to be built into medium systems. In some cases, such efforts have doubtlessly improved the usability of the systems in question. In other cases, the proposed design methods strike many people in both industry and science as labour intensive, expensive, or cumbersome; consequently, their actual application is still limited.

What has emerged, however, is an ever growing number of languages and meta-languages, modelling techniques, CASE tools, repositories, translation software, and so on. Most of these techniques and tools are oriented towards software engineering (i.e. are more technology-oriented than user-oriented), but some are indeed user-oriented or in any case encourage user participation. Though we emphasise again that such approaches and devices may have great merit, we also observe that what seems to be lacking is sufficient insight into a highly relevant aspect: the dynamics and priorities of the 'language game' that is played when language is created, specified, and built into medium systems.

From a language point of view, differentiation between various user domains and, more importantly, *change* of the conceptual situation in those domains is rule rather than exception; change is a constant factor. In addition, for concepts to be functional, everyone confronted with them will need to *know enough about them*. To effectively deploy the many means for conceptualisation support available, in the many differentiated and changing language environments that occur in ICT practice, we need to better understand the dynamic mechanisms involved in those situations and then find the means to select and implement the right approach for the right situation. However, the language-and-meaning games played are complex and require a theoretical framework of their own in order to be understood, charted, and explained. *We observe a knowledge gap in this respect, and hope to contribute to filling it.* For a detailed discussion of how we intend to do this, we refer to Chapter 1.

2.4 Summary

We view language as a set of concepts, each concept consisting of a form and a meaning. Concepts can be complex or not. Their meaning in use depends chiefly on interpretation in context. Our focus is on concepts used for human-to-human communication, not on representations used for computational symbol manipulation. We are chiefly concerned with matters of lexis (words), and mostly discard syntax (the structure of phrases and sentences). Also, our main interest is in lexical meaning rather than in issues of word form (though form can never be ignored). We distinguish between open an closed language. A closed language has a specified, limited number of concepts, and thus has a limited 'conceptual bandwidth'.

We are concerned with closed communication (i.e. communication performed by means of a closed language) performed through an information system, in which that system actively restricts the language used (making available only a limited set of concepts). Such a system is a closed language medium system. Medium systems can be implemented by means of medium machines. The language that may be used through a medium system is its medium system language. It can be used to perform speech acts, using the utterances that may be composed by means of the medium system language. If the medium system language is restricted, so are the utterances that may be produced through the medium system, and consequently also the speech acts that the user may perform by means of it. Concepts and utterances may exist outside context, but speech acts can only exist in context.

Five properties of natural language use may be hampered in case frozen language is used in communication. Language may not be adapted; there may be no possibility to discuss the meaning of speech acts; there may be no possibility to discuss the meaning of de-contextualised concepts and utterances. Also, agreement on speech act meaning may not automatically lead to related considerations concerning concepts and utterances, and reached agreement on (changes in) language is often not coupled to an actual mechanisms to change language specifications. These properties are intimately related to linguistic meta-communication: communication about language.

Restricted availability of the above properties may lead to sub-optimal language functionality in information system use. Such flawed functionality may be condoned because people are very good at coping with hampered conceptual functionality, and are used to it occurring in information system use. Closed language specification (or the freezing of language) and closed language use (or the use of frozen language) can be driven by three interwoven factors: efficiency of communication, certainty of communication, and the need for using finite state medium machines. These three factors together we call the 'practical factor'.

The frozen language issue is expected to gain importance as information systems based on frozen language grow in number and complexity, and are used more widely. Their functionality and user-friendliness for a large part depends on their language functionality. In addition, information systems (including medium system languages) need to be increasingly flexible, allowing them to remain aligned to their changing contexts of use. Large-scale specification and adaptation of language seems increasingly unavoidable, and the pressure this puts on medium system language flexibility and functionality is growing.

Quite a few ways of dealing with frozen language in ICT exist. However, most approaches consider the static specification of language unproblematic in principle; they focus on the quality of medium system language specifications (the result of a limited form of meta-communication) rather than on the more general problems related to flawed meta-communication and the frozen language issue. We consider it useful to look both at the ability to better freeze language and at ways of countering its various side-effects. Many useful meta-languages and specification aids may be available, but to position these optimally in view of general meta-communication activities we need more insight in meta-communication processes (especially processes of explicit conceptualisation) and their relation to the situations in which they occur. This we will try to achieve.

Chapter 3

Language and Communication

In view of further development of our chosen 'direction of solution' for the Frozen Language Problem (Section 1.2), and to provide a wider background for basic theoretical notions previously introduced (Section 2.1.2), a further theoretical underpinning of our view on language and (meta-)communication will now be provided. Chapter 3 thus serves the purpose of answering research question Q1.4 (see Section 1.2):

Q1.4 What are the linguistic mechanisms and factors underlying the description of, need for, and course of the processes of freezing language?

So the chapter concerns a general theoretical background of the theory we want to develop. We discuss our basic view on language, especially on what it is in relation to what it is for. Matters of language structure are hardly discussed at all. Instead, we focus on general and fundamental ideas about meaning, information, and communication, and touch philosophy at times. We present a patchwork-like overview of theories concerning language and communication that for a large part have inspired and support the ideas put forward in this thesis. Our presentation of linguistic and communication theory can only be a brief one, and one that is somewhat biased by the direction we have already chosen to take. It is by no means a comprehensive listing of all matters related to our research, and it does not offer an elaborate evaluation of possible theoretical options or a full account of our own theoretical decisions and preferences. Least of all does it pretend to contribute to fundamental linguistic theory. We merely hope it provides a coherent description and explanation of those insights concerning language, communication, and information systems that are fundamental to our discussion and the direction it takes.

Yet on the other hand, the theoretical presentation necessarily does include and reflect our own ideas concerning meta-communication and conceptualisation processes, and we would like to think that some of these do involve some innovative views on fundamental matters. The patches of theory may have been borrowed from various fields and theories, but the patchwork still is our own. This also means, of course, that we are responsible for any incorrect representation of borrowed theory. Also, many of the more general insights have been around for many decades and cannot be easily contributed to a particular author.

3.1 What Language is For

Human beings are aware of their own and each other's existence; they are "conscious" beings. They depend on each other not only for reproduction: they have discovered that in most cases things look better for them when they help each other out, and *cooperate*. What is more, somewhere along their long and winding evolutionary path they have acquired the trick of not only observing each other in an attempt to find out individual and common purposes, but to make sounds and gestures, and more in general *do things* with the primary purpose of signalling to others intentions, situations, orders, etc., in order to *coordinate* their behaviour, and even their thinking. We call such signalling *communication*. The average Homo Sapiens has in her repertoire all sorts of patterns of behaviour and knowledge dedicated to achieving communication.

But not all communication involves language. Put generally, language concerns a particularly advanced set of communicational tricks that involves the use of (often arbitrary) *symbols*[1]. In addition, it typically includes the *meaningful combination of elementary symbols into more complex symbols*. The most common way of talking about such combination is to say that *words* are combined into *phrases* and *sentences*, though this does no justice at all to the full complexity and variation of language structure. Importantly, in complex symbolic structures, the combined meanings of the individual words as well as the order in which they occur influence the meaning of the whole: the meaning of a sentence is not just all individual word meanings "added up". For a good overview of these basics, see (Cruse, 2000).

We might call the general, innate capacity (and drive) of human beings to acquire and apply linguistic skills and knowledge the *language instinct* (Pinker, 1994). Viewed from an individual perspective, both cooperation and language can be seen in the light of helping out the individual human; looking at the human race as one distributed, continuous organism, the general human capacity to cooperate and communicate and to hand over this trick to every new generation (genetically or otherwise), has been a great help in sustaining the species[2]. Somewhere in between individual benefit and the benefit of humanity, language is vital for the forming and functioning of *communities*.

Humans also invented many other things (and tricks) to help them do things, called *tools* (and *techniques*). Highly developed use of tools and language have both been suggested to be crucial attributes by people pointing out the considerable difference between humans and '(other) animals'. In this sense, language and tools (and *language use* and *techniques*) have much in common, and our general approach to language and communication hinges around the metaphorical view of language *as a tool*[3]. In this view, language takes its place among the pointed stick, controlled fire lighting, plumbing, and the bunjee jump[4]. Undoubtedly, language is a very special,

[1]This is in itself a matter of debate. We are of the opinion that indeed, it is possible to communicate through, for example, gestures that are not conventional and previously determined symbols (instead they may be strictly iconic), and that these are not 'language'.

[2]Even if both cooperation and language are regularly used by members of the species to seriously damage other members

[3]We use the term 'tool' in a sense including not just objects, but also more abstract entities alternatively denoted by terms like 'technique', 'procedure', or even 'method.'

[4]Respectively allowing us to defend ourselves, keep warm, not get ill, and be scared senseless.

unique tool, but a tool nonetheless. Language, then, enables communication, and thereby facilitates coordination and cooperation.
Arguably, language is also used for purposes not so directly related to cooperation or even communication. Language can alternatively be seen as a means of describing and even understanding the world: of devising a 'conceptual chart' of the phenomena that we as humans observe and experience. This view emancipates language from its role as an instrument for communication. However, we assume that its function as an instrument for reflection and understanding coexists with its communicative and cooperative (and perhaps other) functions, and that in fact these functions often blend, and depend on each other.
For example, language may be used to express emotions without immediate coordinative purpose, or for pure socialising (Clark, 1992). Indeed, a curse or surprised outcry is perhaps not primarily aimed at coordination, or even communication. However, it seems reasonable to suggest that such means of individual expression at least originated in communicative functionality. As for socialising, this too does have a coordinative function, though admittedly it is less obviously coordinative than, say, a direct request for action. Socialising coordinates social structures: feelings of group coherence, shared interest and purpose, and so on. Also, given that humans are social beings, the seeking of understanding of the world necessarily involves co-understanding at some point (also see Section 3.3), and this provides a link between the reflective and the communicative function of language.
So we acknowledge that seeing language strictly as a tool for communication and cooperation represents a somewhat onesided view. However, it is the one most relevant to our perspective, and we will allow it to dictate the scope and direction of our discussion.

3.2 Language and Meaning

We will now present our view on language, in particular on how it relates to 'meaning'. Our discussion is directed towards some theoretical standpoints that underly and support the ideas on freezing language introduced in Chapter 1.
First we discuss the fundamental notion of *sign* and the intuitive and 'internalised' nature of language. Next we present our central, functionalist and action oriented perspective on language. We continue with a philosophically flavoured discussion of the relation between language and reality, and our stance in this respect. This is followed by a discussion of how concepts emerge (conceptualisation), and how meaning relates to the human mind. We then consider how the description of meaning fits into our theoretical framework as presented so far, and suggest a new perspective on meaning description. After this crucial section, we continue with a discussion of some additional aspects of language and meaning relevant to our effort: context, linguistic knowledge, and the robustness of language.

3.2.1 Language and Signs

In the first decennium of the 20th century, Ferdinand de Saussure (de Saussure, 1916) explored the key idea that language consists of *signs*, and that a sign (French:

signe) consists of a form (*signifiant*) and some concept this form stands for (*signifié*), thereby raising one of the pillars of contemporary linguistics (very roughly put, the fundamental linguistic dichotomy of form and meaning). Another fundamental distinction made by de Saussure was between *langue* (the sign system of which knowledge is shared by all members of a language community) and *parôle* (language use), where *parôle* was somewhat negatively described as everything that is unsystematic and not part of *langue* (Kaldewaij, 1986). De Saussure thus emphasised both the systematic and the social (shared) nature of language.

We mainly follow de Saussure in our view on signs, putting the idea central that a sign consists of a form part and a meaning part. We do this not because de Saussure's work is the predominant source when it come to the study of signs; in fact there are various more recent, and far more advanced theories concerning the subject[5]. However, for our purposes we prefer de Saussure's definition of sign exactly because of its simplicity.

The sign as the basic, fundamental entity in language is central in the field of *semiotics*. A number of approaches to semiotics exist, and the field rises above the exclusive study of language to include the study of signs and their meaning at its most general, covering any perceivable or mental entity that 'means something to someone' (for example, see Galinski and Picht, 1997). As such, semiotics is very much a philosophical field. In this thesis, we are *not concerned with semiotic studies in the more technical sense*, yet our general perspective on language embraces the basic semiotic one: language is a toolkit that allows us to use (complex) signs in order to communicate within a community. The semiotic perspective envelopes the linguistic perspective and contributes greatly to the understanding of meaning in language.

So the definition of sign as maintained in this thesis is a simple and relatively limited one: signs are *combinations of a language form and a meaning*. Our notion of concept is directly derived from that of sign; in fact, it is *identical* to it (also see Section 2.1.2). Elsewhere, the *meaning of a sign as such* is referred to as being a concept (i.e. a concept then is a chunk of 'pure meaning', so to speak; for example in J. Hoppenbrouwers, 1997), but we refer to a sign's meaning as simply its 'meaning', or sometimes its 'semantics'. We realise our choice of terminology is slightly eccentric in view of most of the literature, but our agnostic view on 'cognitive' concepts convicts us to a stance in which language *forms* are exceptionally important in any description of meaning (see Sections 2.1.1, 3.2.6, and 3.2.4 for more on this). Also, since the item combining language form and language meaning is central to our effort, and in principle covers both elementary and complex linguistic entities, we find it preferable to talk about *concepts* instead of either *signs* or *word meanings*.

Finally, whereas the Saussurean notion of sign ('signe' in French) refers to an item belonging to the shared language system (for example, a national language like Welsh),

[5]We should mention here the work of C.S. Peirce (Peirce, 1931), who developed a highly complex and sophisticated theory of signs. Attempts are made to create implementable theoretical/computational frameworks of linguistics based on (Peircian) semiotics (for example, see Sarbo et al., 2001). We also refer to work by Eco (Eco, 1999). Relevant contributions concerning semiotics and ICT are Andersen's computational semiotics (Andersen, 1990, based on fundamental work of Hjelmslev), and also (Stamper, 1996) and (Liu, 1993), who suggest a semiotics-based view on communal norms.

we distinguish between *concepts* (which are mental entities in the mind of the individual)[6] and *linguistic conventions* (see next section), which are abstract representations of 'the knowledge of concepts which is shared among people who's language includes those concepts'. In this respect, therefore, our definition of 'sign/concept' differs from that of De Saussure.

3.2.2 Intuitive Skills and Internalised Conventions

The toolkit of language is based on two things: the innate and universal capacity of all humans to 'do linguistic things' (see Pinker 1994), and the socio-cultural interaction between human beings that shapes, develops, and passes on the conventions of language. Such conventions are subject to change, and have developed differently in different communities. This explains why there are different languages, many of which are so different as to be incompatible in use. In fact, it can be argued that when it comes to details of form and meaning, no two humans share the exact same language. In this thesis, it is mostly the conventions we are interested in, or more correctly put, the *limitations* of those conventions: the small and often treacherous differences in linguistic form and meaning that tend to be just outside the reach of clear, intuitive conventions.

Language conventions (most prominently: the word forms that are part of some language and the meaning associated with those forms by members of the language community) mostly emerge and are passed on *intuitively*: the human capacity for learning language is mainly based on mental processes that take care of themselves without rational or even conscious intervention. People seldom pass on words by passing on written dictionaries. Yet *seldom* does not mean *never*[7], and in fact we are interested precisely in the relatively rare acts of making explicit the internalised conventions about words, and the even rarer deliberate acts of creating linguistic conventions from scratch.

3.2.3 Language and Meaning as Action

In the study of linguistics, relatively little attention has been payed to the *functional perspective* on language: thinking about language primarily in view of what it does and is for, and relate this to other aspects (mainly structure). Mainstream linguistics (most prominently, *generative linguistics*, also known as Chomskian linguistics) has mostly attempted to identify universal properties of language, generally focusing on structure of language forms as an isolated topic ("autonomy of syntax"). There are, however, a few schools of functional linguistics, most prominently *Functional Grammar* (Dik, 1989) and *Functional-Systemic Grammar* (Halliday, 1976), that aim at describing structural properties of language (both form and meaning) in relation to their function in communication, putting this function central.

[6]Note that their status as 'mental entities' does not contradict our earlier definition of concept being 'a combinations of a form and a meaning'.

[7]For example, a notable exception is the acquisition of what is called a 'second language', i.e. any language that is not learned early in childhood (first language acquisition) but after, say, the age of twelve, normally by means of specialised education (Groot, 2000).

It should be made clear that functional and generative linguistics do not differ so much in their view on what language ultimately is, but in what methodological perspective they take and what aspects of language they are interested in. Given our research goal, which concerns language as a communicative tool, we embrace the functionalist approach to linguistics.

Though few linguists would object to metaphorically seeing language as a tool for communication, it is a point of view not usually carried through as far as we do in this thesis (not even in functional linguistics). This is partly because we link up the use of language with the use of machines, which makes the metaphor stronger. However, we do maintain that theoretical aspects of language lend themselves to a strong functionalist view. In (Weigand and Hoppenbrouwers, 1998a), the authors argue in favour of an action-oriented approach to the lexicon within the framework of functional grammar (in line with Searle 1969). Put briefly, they view words as linguistic tools that can be used to do things *with*, but also are objects one can do things *to*. For example, inflecting a word can be seen as an *action on a word*, while using a word in a predication is an *action with a word*. Note that both such actions take place on an intuitive linguistic level and in the individual, though they largely follow socially constructed conventions. In this thesis, we extend the action view on language to *conversations about words*, which are complex, essentially social sequences of actions on and with words, which take place at a meta-linguistic level. We will return to the action view on language further on in this section.

What is it, then, people *do* with words, if we assume this defines what they are? The answer is: many things, on many levels. In (Austin, 1962), titled "How to Do things with Words", the idea is first explored that ultimately, language is used to achieve something. An utterance may even be the very act bringing a change about ("I hereby declare you husband and wife"), but the more general point is that people say things to do things: inform someone, get someone to do something, etc.

If we look at how words are used to do something, then this "something" (say, a request) for a large part depends on interpretation within the context within which those words are used: a particular situation, involving a particular speaker and listener. This point of view adds something to the more common linguistic approach (including many functionalist efforts) that prefers to look at the meaning of words and sentences out of their immediate context of use. Note, by the way, that logical approaches to the description of propositional meaning *do* take into account matters of context, at least up to a certain point: a predicate structure becomes a proposition only when it is placed in some domain, in which it then is either true or not true. We will further discuss context in Section 3.2.8.

Searle further developed Austin's ideas (Searle, 1969), and distinguished the following aspects (or 'types') of speech act[8]: *illocutionary acts*, *propositional acts*, and *utterance acts*. Illocutionary acts are the main functional 'envelope' (actions like putting forward requests, orders, assertions, etc.; see Section 3.3 for more). Inside the envelope there are contextualised propositions (called the "propositional content") and the actions creating these are *propositional acts*. Finally, Searle distinguishes *utterance acts*, the acts of composing a grammatical structure and the sounds of an utterance (Cruse, 2000, 331-2).

[8]In fact, he also distinguished a fourth aspect: the Perlocutionary Act, which we discard here for reasons of relevance.

Austin and Searle's speech-act view does not contradict the more standard, predication-oriented view of language, nor proposition-oriented semantics, but envelops both in a 'wider' view of language as action. The creation and use of propositions is still crucial in the production of sentences (note however that from a purist action view on language, sentences do not 'predicate'; humans do).

In (Weigand and Hoppenbrouwers, 1998a) it is pointed out that productive language acts should be complemented with receptive/interpretive acts on the side of the listener: it is quite possible that an 'incomplete' utterance is put forward that needs some creative work on behalf of the receiver to be intelligible. In addition, it is suggested that "many more [acts] can be identified" (*ibid.* p.284): the main speech acts can be seen as being complemented by many other 'sub-acts', such as predicating, referring, assigning topic, invoking some association in the listener's head (for example, invoking the general concept of *cat*), etc.

It is quite reasonable to see such an 'action view' on language as a variation on the structuralist view on language, as it relates linguistic actions to the deployment, creation, or transformation of the various structural elements distinguished in structuralist theory. This is indeed one of the main points in (Weigand and Hoppenbrouwers, 1998a), which relates linguistic actions to structures belonging to the Functional Grammar framework. However, the action view on language also, and importantly, serves to emphasise a perspective on languaging as a *process* (Hoppenbrouwers, 1994), and as *something that is done in order to achieve something.* In a similar vein, (Clark, 1992) distinguishes the *product tradition* and the *action tradition*: "The product tradition starts with language structure and investigates how it is manifest in speaking and listening. The action tradition starts directly with what people do with language and investigates how that works" (*ibid.* p.xiii). (Van Reijswoud, 1996, pp.21-36) compares three perspectives on language relevant to our present discussion: the *representational*, *functional*, and *action* perspectives. Though his work has been a main inspiration to us, his point of view differs from ours concerning the basic view on meaning. On p.31, van Reijswoud suggests that "the study of the representational character of language signs concentrates on form and meaning, and excludes the function and the effects of language". He not only states that the standard representational view leaves something out (which we agree with) but also suggests that meaning and function are distinct, maintaining the typically *representationalist* distinction between 'propositional content' and 'illocutionary load'.

In our stronger version of the action perspective on language, both proposition and illocution are essentially grounded in action and experience (which is "remembered action"), and both are part of "meaning" (see Section 3.2.4). Rather than distinguishing "meaning" and "action" in language, we assume that meaning can (and in some cases might better) be discussed in terms of action and experience. Thus, even a functionalist theory like Functional Grammar, though it includes action-oriented concepts like "illocution" and attempts to explain syntactic structure in view of its underlying semantics and pragmatics, is still based on the traditional, proposition-based representational view on *meaning*. It should be very clear that we do *not* believe this to be 'wrong' within the goals of the Functional Grammar framework, nor in many other contexts (including our own), but we do prefer to entertain a different view on the *fundamentals* of meaning (for a similar stance, see Harder 1996).

We believe that at the very least the action view on meaning can 'peacefully coexist' with the structuralist view on meaning, but also, and importantly, *points towards another way of dealing with meaning in practical situations.* To avoid the suggestion that we intend to start a polemic on the nature of semantics, we will where possible refrain from the use of the term "semantics" (with its distinct representational flavour) in our further theoretical discussion, giving preference to the broader term "meaning".

We have characterised our own approach as both functionalist and action-oriented. Indeed, we look at language from the perspective of what language is used for (functionality), and believe that on a deeper level this can be best explained in terms of action and experience (strong action perspective). To van Reijswoud, the action perspective merely represents the speech act (illocution-oriented) part of linguistic theory. In our view, it can be taken as a fundamental perspective on language and meaning in its broadest sense.

3.2.4 Language, Meaning, and Reality

We have so far focused on the relation between language and meaning, and put forward an action-oriented perspective on both language and meaning. We carry on with a discussion of a rather philosophical issue: the relation between language and reality, on which any view on the meaning of language inherently depends. We do this mainly at the hand of (Winograd and Flores, 1986), who have pioneered explorations in the interdisciplinary field covering linguistics, information systems, and computation, and who's work has been a main inspiration to our current effort. As is discussed in (Winograd and Flores, 1986), the prevailing direction in Western philosophy[9], as based on the tradition going back to Plato and Aristotle, tends to divide the world in "two separate domains of phenomena: the *objective* world of physical reality, and the *subjective* mental world of an individual's thoughts and feelings" (*ibid.*, p.30). Language is seen as a system of (complex) symbols that relates objects in the objective world to (presumably linguistic) symbols in the subjective world.

Commonly, the meaning of a linguistic item is represented through the devices of set theory and first order logic, e.g. quantifiers, arguments, predicates, sets, etc. The most basic form of meaning is then expressed by assessing the truth of a logical proposition: *true* or *not true* (in some given domain or context). The meaning of more elementary linguistic items is usually expressed in terms of sets denoting elements with certain properties (Gamut, 1991). In essence, linguistic meaning is thus described by translating linguistic representations to logical representations. This renders a "semantic" description, which is considered to be more exact and unambiguous than a natural language description; in fact, the language of logic was created as a tool for being more precise and unambiguous in the expression of meaning than is possible through natural language. Because the above view on meaning boils down to translation from one representation to another, it belongs to the *representational* perspective on language and meaning. We have already

[9]Please forgive us the disrespect for philosophical precision and depth in the short account that follows. We are merely trying to make a rough point, without claiming to even break the philosophical surface.

discussed some aspects of the representational perspective on language in Section 3.2.3.

Though representational approaches to meaning are quite effective in theoretical modelling, calculation, and simulation (and are in fact instrumental in most branches of scientific endeavour), they explain little about the essential nature of meaning. Unless we believe that our thoughts (or even reality) consist of sets and propositions (and many have indeed entertained similar beliefs, or refused to think beyond them[10]), a strictly representational view of meaning unescapably keeps alive an insolvable polemic that moves between two extreme philosophical views. Quoting (Winograd and Flores, 1986) on Heidegger: “Some [schools] argue that we cannot coherently talk about the mental domain, but must understand all behaviour in terms of the physical world, which includes the physical structure of our bodies. Others espouse solipsism, denying that we can establish the existence of an objective world at all, since our own mental world is the only thing of which we have immediate knowledge[11]” (*ibid.*, p.31).

Heidegger (among others[12]) confronted the paradoxical duality of the subjective-objective world view, and this led him to “reject both the simple objective stance (the objective physical world is the primary reality) and the simple subjective stance (my thoughts and feelings are the primary reality), arguing instead that it is impossible for one to exist without the other” (*ibid.*, p.31). “The thoughts of an individual perceiving the world and acting in it are thus as much part of that world as the world is part of the thoughts. [...] Separation of subject and object denies the more fundamental unity of *being-in-the-world (Dasein).* By drawing a distinction that I (the subject) am perceiving something else (the object), I have stepped back from the primacy of experience and understanding that operates without reflection. The interpreted and the interpreter do not exist independently: existence is interpretation, and interpretation is existence. Prejudice is not a condition in which the subject is led to interpret the world falsely, but is the necessary condition of having a background for interpretation (hence Being)” (*ibid.*, pp.31-2).

From this we conclude (and this is no longer Heidegger speaking) that the ‘objective’ approach to meaning in language is essentially flawed. Representations of meaning can ultimately only describe subjective and abstracted *approximations* of meaning, in one form or another, and may do so quite adequately *given a limited goal.* This stance relieves us of the burden of having to discuss ‘ultimate’ or ‘fundamental’ meaning. Let us therefore cut a very long story short and assume that we should no further busy ourselves with attempts to break the concept/reality deadlock, but focus instead on selecting those representations of meaning that best serve *particular communicative goals in particular communicative situations.*

However, such an approach still focuses on representations. Though the discussion of which representation of meaning best serves a particular communicative purpose is a

[10] We do not deny that some of the important tricks the human mind can perform concern ‘representations’. We merely believe that logic-like structures in particular are not *fundamental* to the meaning of *language.* For more on this, see (Harder, 1996, pp.107-).

[11] At this point, falling trees in distant forests are often called to mind

[12] For example, the rejection of the fundamental subjective-objective distinction has been commonplace for several millennia in various influential branches of Eastern philosophy and is central in the teachings of Siddhattha Gotama.

crucial one, it fails to cover the observation that meaning is not only subjective, but given the highly complex and dynamic nature of subjective thoughts it also changes over time, even for a particular individual. So even if some individual provides a meaning description, she can in principle only provide a *momentary* echo of meaning. Note that this does not contradict the observation that basic word meaning may be relatively constant (for example, the word 'cat' has not significantly changed its basic meaning for centuries). It merely emphasises that such constancy *should not be taken for granted*: that *change is a constant in the relation between language and meaning.*

Consequently, we cannot but consider *the subjective, individual mind to be pivotal in the relation between language and reality*, and essentially depend fully on *context-dependent descriptions of meaning provided by individuals* in our approach to the meaning of language in relation to 'reality'. This does not imply that we consider language and meaning to be an individual matter (to the contrary, language has important social aspects); it just means that we distrust in principle any description of 'general' or 'true' meaning. We will always, ultimately, have to ask: who's meaning description is this, and in which context was it described?

3.2.5 Emerging Concepts: Conceptualisation

Now let us consider how concepts come into existence. According to (Winograd and Flores, 1986), concepts emerge from a specific and limited *need* to 'conceptualise'. They are the result of a creative *conceptualisation* aimed at the understanding of 'problematic' or at least 'relevant' aspects of the world as perceived by a human. The above notion of concepts being "relevant" is strongly linked to the notion of "breakdown" (Winograd and Flores, 1986, p.69): the occurrence of a situation in which something hitherto unnoticed requires specific attention, typically because of some problem. Note that this is primarily related to the reflective function of language: the process of making sense of the world.

Here we must carefully distinguish two interpretations of the term 'concept' (as previously mentioned at the beginning of Section 2.1.1): concept as 'pure meaning' and concept as 'a combination of language form and associated meaning'. Winograd and Flores refer to the first interpretation of the term (also see Section 3.2.1), and thus talk about 'idea formation' or 'thought formation' rather than 'word formation'. Our interpretation of 'concept' being the second one, we define conceptualisation differently, as 'word formation' or to be more precise, 'sign formation' (in the Saussurean sense). Therefore, whereas initial 'cognitive concept formation' of the Winograd-and-Flores kind indeed occurs as the result of a 'being-in-the-world breakdown' (let us call it an 'experiential breakdown', primarily related to the reflective function of language), we are more interested in what happens if such a breakdown also results in the need to *communicate about* it (obviously, primarily related to the communicative function). If such communication is impossible, we are confronted with a *communicative breakdown.*

However, a communicative breakdown may have many causes, insufficient language being only one of them. We therefore further distinguish a particular type of communicative breakdown, a *language breakdown*: a problematic lack of language to express (communicate) something. It is this type of breakdown that creates the

need for *'linguistic conceptualisation'*.
Concepts, then, are the result of a creative conceptualisation ('concept making') aimed at the 'languification' of relevant aspects of the world as perceived by a group of humans. Things that need not be communicated about are not 'named': there is no word for them, nor need there be. As the concepts that are relevant within a language community change, the language changes with them. Assuming that all concepts are the result of conceptualisation, there are three possibilities in a situation in which concepts are needed for communication:

1. Use an existing sign

2. Adapt an existing sign

3. Create a new sign

Conceptualisation as we see it only occurs in case of options 2 and 3; the "use of existing signs" in option 1 applies to regular language use, not to 'language creation' or 'language adaptation'.
Even if existing linguistic *forms* are used, it is quite possible that the meanings associated with them are sufficiently different to speak of 'adapting existing concepts' (option 2). Therefore, it is a fine line between options 1 and 2. Arguably, distinction between the two can only relevantly be made if explicit concept descriptions (both form and meaning) are involved, since it is reasonable to assume that these can be compared.
We will not at this point address the question which of the three options is preferable in some situation. We cover both option 2 and 3 by the general term of 'conceptualisation': it takes place wherever concepts are *selected* or *introduced* at *meta-level*.
In addition to conceptualisation taking place, to linguistically communicate information both speaker and hearer must know the linguistic signs used: they must both realise that the linguistic forms are linked to particular meanings. Yet these 'meanings' ultimately do go back to the 'experiential' level as discussed by Winograd and Flores. Two different types of knowledge come into play here: *experiential knowledge*, which may be similar for individuals but cannot as such be communicated, and *linguistic knowledge*, which is present in individuals but *must* be shared to be of any value. For example, it is notoriously difficult to describe in detail something like the taste of a strawberry. However, it is quite possible to *name* 'strawberry taste' and use it successfully in communication with someone else who has tasted a strawberry –and who as a result may even *remember* it. However, not the taste experience as such has been communicated, just the label for it. For more on this, see Section 3.2.9.
So for linguistic concepts to be truly meaningful, they have to be associated with 'deeper', cognitive concepts. After all, linguistic knowledge merely provides a vehicle enabling communication of a 'skeleton meaning' that relates one individual's thoughts and experience to that of another individual. Once more quoting (Winograd and Flores, 1986): "Heidegger insists that it is meaningless to talk about the existence of objects and their properties in the absence of concernful activity, with its potential for breaking down. What really *is* is not defined by an objective omniscient observer, nor is it defined by an individual [...] but rather by a space of

potential for human concern and action" (*ibid.*, p.37). Language then is a tool that helps humans coordinate their movements through that space, but it is at the same time a product of breakdowns (at various levels) that are part of those very movements.

3.2.6 Meaning and the Mind

As we have argued, language does not have to capture the intricate depths of thought and experience to be useful. As long as it invokes *enough* common *associations* in two communicating parties, it has its use; the people involved will actively (though mostly intuitively) 'fill in' many details in their individual interpretation, but this need not hinder communication; in fact, it allows for communication of 'skeletal meaning'. In this respect, it is important that we emphasise how many things people do *not* communicate (in a literal, representational sense) through linguistic acts, and that this arguably is why language works in the first place considering some of the gruelling conditions in which it is regularly applied (see Section 3.2.10).
Let us now at last discuss in more detail what linguistic forms *do* when they "mean" something. Consider the popular metaphor of language as a *vehicle that allows people to adjust each other's 'mental picture of the world'.* Despite our rejection of the representationalist view as a fundamental explanation of meaning, we see it as unescapable that in the mind/brain of each individual human there take place some processes that we may refer to as "mental", and that at a particular point in time, somewhere in the mind/brain some sort of neurobiological data is stored that allows the individual to consistently imagine the world. Using the term "picture" strictly metaphorically, this picture of the world is, among other things, used for interpretation of signs (including linguistic signs). The picture is extremely complex, somehow including not only thoughts, beliefs, memories, or predictions about "facts in the physical reality", but also harbouring reflections of emotions, commitments, social structures, possibilities and scenarios, linguistics tools as used differently by numerous different people in different situations, and so on and so forth, down to constructions like "I think that you think that I think..." (for more on this, see Section 3.2.9, and Clark, 1992, pp.68-).
Our language-as-action view seems to be compatible with cognitive models of linguistics (see Johnson-Laird, 1983; Jackendoff, 1983; Levelt, 1989; Aitchison, 1994), but in this thesis we are careful not to concern ourselves with what exactly happens in the black box of the mind. We assume there to be an interpretation, but how it looks cognitively is beyond our current scope. Our access to the meaning of language is strictly through language: description of meaning by human actors.
Communication is often seen as the carrying over of *information.* In our account, the signal (data) carrying the information triggers an interpretation *event*: it causes *a change in someone's mental picture of the world.* To say that the data by itself is information is therefore, strictly speaking, incorrect: information in its purest sense is an action, "an information" (deverbal noun[13]).
In (Harder, 1996, pp.112-4), as part of a lengthy and careful argumentation that in spirit is not unlike our comparatively brief presentation, Peter Harder puts it

[13] A deverbal noun is a nominal derivation of a verb, e.g. "protection" from "to protect"; "clarification" from "to clarify".

like this: "each semantic element has a job to do in producing the representational content that the utterance is supposed to convey. [...] The word "procedural" thus transfers the focus to what goes on before we have reached the complete representation that constitutes our interpretation of an utterance – meaning as process input rather than meaning as the content of the envelope. [...] What you have arrived at when you have properly understood a linguistic utterance is not its linguistic meaning, but something much richer and more valuable to you: the way in which the utterance changes your own actual, situational understanding. Linguistic meanings are just a step on the way." In the words of Simon Dik (Dik, 1989), messages are like Deltas: they specify a kind of operation that the hearer must apply to the shared knowledge. At the informational level, the context is the total of background knowledge relevant to the message that the communicative agents share. The less context they share, the more explicit the message has to be.

For an information to occur, a human mind is needed (including all the knowledge needed to perform the interpretation trick), and human minds are necessarily part of individual humans. We therefore see information as both *interpretational* and *situational* by nature, and unique in principle for a particular person, with a particular "picture of the world", and occurring in a particular situation within a particular context.

We have, until this point, repeatedly used the standard image of a speaker and a hearer. This image, while being useful, requires some further discussion concerning the social aspects of communication. Information may be interpretable by an individual, but for it to feature in *communication*, a relation between two communicating agents has to be established. This is important, since most "information" that is called by that name is part of a communication process and is carried by language, which involves a system of conventions. Even if someone is speaking to herself but is secretly overheard by someone, two-way communication takes place, though it takes place *unintended* by the speaker.

Language, then, derives its communicative power from evident and deliberate cross-community similarity in the interpretations triggered by certain linguistic forms. Up to a point, this enables groups of individuals to systematically *invoke sufficiently similar interpretations in each other* and ultimately to use this trick to coordinate their actions. Such invocation thus covers the result of what language does when it 'means'. As explained, we refrain from further discussion of the mechanisms and structure of the cognitive processes underlying and constituting it.

Sufficient similarity refers to a degree of similarity in the invocation of interpretations that is high enough to *fulfil the communicative demands of the particular situation*. In other words, one has to be *sure enough* that the interpretation invoked in the listener by some utterance is the one intended by the speaker insofar it has the *intended effect on the listener's knowledge, train of thought, and ultimately her actions*. Note that this explanation includes the notion of *certainty* as discussed in Section 2.2.3.

3.2.7 Describing Language Meaning –and Beyond

If we view meaning as essentially based on individuals' description of meaning (Section 3.2.4), this does not mean that some individual can effortlessly produce any

such description. In this respect, there is a strong relation between meaning representation and *knowledge representation*. For their hypotheses regarding knowledge creation, (Nonaka and Takeuchi, 1995) draw heavily on the distinction between explicit and tacit knowledge made by (Polanyi, 1958). Polanyi assumes knowledge to exist on a spectrum where all knowledge has a tacit dimension. 'Tacit' is opposed to 'explicit' or 'conscious': "We can know more than we can tell" (*ibid.*, p.95). The reason for this is that knowledge, according to Polanyi, is first of all acquired by our body through the daily activities we are engaged in. Therefore, it is not only often impossible to externalise this knowledge – think of an attempt to explain someone how to ride a bicycle – but also not desirable (for more on this, see Section 3.2.9).
Knowledge is something that resides within us, and manifests itself through our actions, and we therefore do not need to document it for our own sake. As was already suggested in Section 3.2.2, this also holds for linguistic knowledge (Hoppenbrouwers and Weigand, 2000), and it implies that someone's ability for describing language meaning is not something that can be taken for granted even if she is, say, a skilled linguist.

To recapitulate, we consider language meaning to be essentially lodged in the individual's mind, and linking it to 'objective reality' is not trivial. In addition, we observe that (knowledge of) language meaning is mostly tacit, and describing it is also not trivial.

Having thus embraced the idea that 'reality', whatever it may be, lies outside the immediate grasp of symbolic/representational modelling (including linguistic representation), and realising that the models and theories we put together to understand the world are typically representational, one might wonder whether we should not just give up on describing meaning altogether. But discussing language and meaning, problematic though it may sometimes be, is not just an academic exercise. It is, after all, a common and useful activity among language users in all walks of life, and in some cases has severe practical implications.

A way out is offered by, indeed, giving up the quest for the complete and precise representation of 'true' meaning and first turn to an approach that looks deliberately and explicitly at the "meaning games" people play without describing particular conceptual representations as such. In other words, we might look at language and conceptualisation *processes* without necessarily connecting them to the actual symbols of language (while leaving this option open). Indeed, note that in our focus on language as presented in Section 2.1.1, concepts are abstract notions consisting of *meaning* and *form*, that are not further specified. Only once we have a good idea what we want to conceptualise, and what we want to conceptualise it *for*, we can start thinking about the *how*. Depending on the situation at hand, we may then decide to engage in explicit meaning games (and perform appropriate kinds of meaning description), but alternatively we may choose to acknowledge particular meaning games as they occur 'in the wild', and then either leave them to their own devices, or guide them.

A more practical phrasing of the above is the following. While doing our best to understand the world and express our thoughts through language, we should understand and accept the limitations of representing meaning and prepare to cope with them as best we can. In other words: we should use our (meta)linguistic toolbox skillfully, yet to our ability, without confusing our (meta)communicative toolbox with the things we can experience and achieve *through* it. We may do anything from leaving natural language mechanisms free to roam to imposing strict linguistic restrictions (requiring solid description and perhaps also *enforcement* of form and meaning). Metaphorically put, in the various language and meaning games that take place around us, we may want to sit back and watch, or decide to become active players, or perhaps we may take the role of a referee. And maybe, we may even try and change some rules.

Consequently, in order to make the necessary decisions about how we choose to deal with language and meaning games we should first develop a reasonably comprehensive meta-perspective on the processes and goals that make meaning representation a necessity. On the basis of that perspective, we can then work towards guided *facilitation* of appropriate meaning description and representation where required. This would include the *development and implementation* of tools for meaning description (meta-languages, formats, repositories, approaches, protocols, methods) but also the *facilitation of responsible selection and use of such tools*. As argued in Section 2.3.2, we believe that the 'tool development focus' has thus far prevailed over the process-oriented 'tool selection and use focus', and we hope to help remedy this.

Note that in the above argumentation, we address meaning description, not 'freezing language'. Meaning description (for example, definition) is only one of the activities that may constitute the process of freezing language (see Section 5.2). However, there is a dependency between the *selection* of concepts and the *definition* of concepts in that concept selection implies explicit isolation and use of 'bits of meaning'.

3.2.8 Context and Meaning

We have seen that meaning is carried by form, and that the linguistic combining of form and meaning to form signs entails the creation of conventions. Conventional meaning is typically the subject of the field of semantics. However, it is clear that context heavily influences the meaning of actual utterances in language use. Contextual aspects of meaning are the subject of the field of *pragmatics*, which in many respects is complementary to semantics. Semantics and pragmatics together cover linguistic meaning. (Cruse, 2000, p.16) defines "linguistic pragmatics" as follows: "pragmatics can be taken to be concerned with aspects of information (in the widest sense) conveyed through language which (a) are not encoded by generally accepted convention in the linguistic form used, but which (b) none the less arise naturally out of and depend on the meanings conventionally encoded in linguistic forms used, taken in conjunction with the context in which the forms are used".

Clark, who's work has to a fair extent influenced our view on language and meaning, further develops the notion of context (Clark, 1992, p.6). He observes that "in the study of language use, investigators appeal time and again to "context" to explain this or that phenomenon. The problem is that they almost never say what they mean by "context" even when it is essential to their explanations." Clark lists six features of context common to most accounts (*ibid.*, pp.64-5):

1. Information Information about (or rather, knowledge of[14]) objects, events,

[14]We have taken the liberty of talking about "knowledge" instead of "information" in our rep-

states, or processes. It may be generic or particular, and it may come from direct experience, from being told, or from inferences based on these sources. It may include, but is not limited to, a person's knowledge, beliefs, or suppositions.

2. **Person relativity** Context is usually relativised, not to people in general but to each particular person. In other words, context is in the eye of the beholder.

3. **Process relativity** Context is relative to a *process* a person is carrying out. In a sentence, the context of a word is really knowledge a person has relative to his interpretation of that word.

4. **Occasion relativity** Context is knowledge a person possesses in the carrying out of a particular process *on a particular occasion.*

5. **Availability** Context is only that knowledge that is *available* to the person carrying out the particular process on that particular occasion. People's memories are full of many things, but only the part of that knowledge that is available (in an immediate, psychological sense) for the task at hand would be considered part of the context.

6. **Interactability** For knowledge to be called "context", it must interact with the process at hand. It has to be relevant in the sense that it must 'make a difference'. Not just any available knowledge is part of context.

The definition summing up these features is the following: *Context is knowledge that is available to a particular person for interaction with a particular process on a particular occasion* (Clark, 1992, p.65).
Clark further distinguishes between *intrinsic context* and *incidental context* (*ibid.*, p.66). "Intrinsic context is the part of the context that, a priori, has the potential of being necessary on some occasion for carrying out the process in question. [...] The incidental context is what remains, the parts of the context that never need to be consulted[15]". To be able to isolate intrinsic contextual knowledge is not only important for psychological studies, but also if we want to include context in our thinking about efficient and certain communication and meta-communication. It may give us a clearer sense of purpose in what we do or do not want to specify and convey in order to safeguard effective communication.
Clark (p.67): "When a listener tries to understand what a speaker means on some occasion, it would be advantageous if the process he uses could limit what it retrieves from memory to some portion of the total information that could be made available" [(i.e. the intrinsic context)]. [...] *The intrinsic context for a listener trying to understand what a speaker means on a particular occasion is the common ground that the listener believes holds at that moment between the speaker and the listeners he or she is speaking to* [Clark's italics]. [...] [Clark is concerned with] how a listener

resentation of Clark's features of context, which seemed more appropriate in view of our previous discussion, and our particular interpretation of 'knowledge' –an individual's "picture of the world".

[15]Note that the difference between 'Intractability' and 'Intrinsic context' is that the former refers to the use of some knowledge in a particular act of interpretation, whereas the latter refers to a *type* of knowledge, in a more persistent sense, that is *potentially* used in interpretation.

tries to determine what the speaker intended him to determine, in part by means of his recognition of the speaker's intentions. [Clark's concern is] *not* [with] the further inferences that a listener carries out on the basis of what the speaker meant; that is, it is about the "authorised" and not the "unauthorised" inferences made by the listener [...]".

Note again that the above description of contextual knowledge is presented as existing in the individual, but that in communication, a substantial amount of knowledge needs to be *shared* between communicating parties: it needs to be or become *common ground*. Indeed, Clark goes on to replace the notion of "context" in language use with "common ground of the participants", which covers "mutual knowledge, mutual beliefs, mutual assumptions, and other mutual attitudes". "Context", in its traditional sense as distrusted by Clark, effectively serves to express a contrast with "context free" (i.e. strictly conventional) structures and meaning; it separates "semantics" from "pragmatics". Clark, who is interested in both semantic and pragmatic games that people play involving and bringing forth shared meaning, considers the notion of common ground a better, safer basis for the understanding of linguistic interpretation than the rather battered notion of "context". He concludes that "in language use the relevant context is simply the common ground of the participants".

3.2.9 Common Ground and Linguistic Knowledge

All linguistic knowledge, being of a conventional nature, is part of a language community's common ground. However, we have previously mentioned linguistic *skills*. It is not entirely clear where knowledge stops and skills begin. In this context, some researchers distinguish *knowledge that* from *knowledge how* (Ryle, 1949; Bechtel and Abrahamsen, 1991, p.152; Hoppenbrouwers, 1994, p.67). 'Knowledge that' is typically theoretical; 'knowledge how' involves activities or skills like riding a bike or speaking grammatically. 'Knowledge how' is typically *tacit* in the sense of ((Polanyi, 1958)); also see Section 3.2.7. 'Knowledge that' seems to be more typically *representational*. We will refrain from further discussion of the distinction between linguistic knowledge and skills here, and further assume both to be part of the common ground.

The above view is essentially compatible with our action-based perspective on language. However (in line with earlier remarks), we should not throw away the child with the bath water and loose track of the fact that people *do* at some level use some representational means when they use language (cf. Harder 1996 p.116) but also, importantly, when they *talk about* language. The latter point will be crucial in our later discussion of linguistic meta-communication.

After initial conceptualisation (Section 3.2.5), linguistic knowledge comes in: *storage* and *sharing* of knowledge. Common human experience has led to the evolution of words and concepts over thousands and thousands of years, which is intertwined with the evolution of the human race as we know it. Communication requires constant re-adjustment of bits of the linguistic toolkit, in line with constant active use and interpretation: the toolkit needs skill and effort in order to be effectively used, and the tools need to be kept sharp and in working order. Humans therefore store linguistic knowledge, transfer it to each new generation, and adapt it as the conceptual situation requires. Such changes primarily take place on a situational level (in day-to-day language use), but in the long run they are a driving force behind the gradual change of the more stable collections of linguistic conventions such as national languages.

In preparation of our later discussion in Section 3.4, it is important to clearly distinguish two types of linguistic knowledge: one type that covers *all conceptual knowledge* and one that covers *socially marked conceptual knowledge*.

First of all, we assume that if someone knows that F means M, or even that F may mean, depending on the situation, M or N or O, the fact that a person knows that this is the case is not in any way a matter of choice. The same goes for knowledge that, for example, person A means M with F, and person B means N with F. Such "general" linguistic knowledge is comprehensive and represents all linguistic knowledge that might be used to come to an interpretation. If it is shared between members of a community, it is a *general linguistic convention*, which can implicitly, even unconsciously hold between people. General linguistic conventions concerning meaning thus relate to *everything some language form may mean, in any situation commonly imaginable.*

In contrast with general conceptual knowledge, we distinguish a type of linguistic knowledge that is particularly relevant for our further discussion: knowledge about what particular *choice* of concepts may be *typical* for use in a certain community and situation. This notion reflects the observation that in some community, particular concepts (words, phrases) are part of the language as habitually or standardly used in that community. Thus, this involves knowledge of which concepts are *part of the communal language*. We assume that such concepts are 'marked' as such in the knowledge of individual agents that are part of the community. We call such knowledge, when applied in a speech act or an interpretation act, knowledge of *communal linguistic convention*. Communal linguistic conventions concerning meaning thus relate to *what language forms are part of the typical language in a community, and what those forms are expected to mean within that community*. We will return to the general/communal distinction in Section 3.4.

The set of communal linguistic conventions is a subset of the set of general linguistic conventions; we have introduced the notion of 'general linguistic convention' mostly to enable us to emphasise that our 'communal linguistic conventions' do *not* cover all shared linguistic knowledge, but only that part of linguistic knowledge that somehow marks concepts as *belonging to some communal language*. One might test whether some concept is 'socially marked' as belonging to a community by eliciting from members of the community whether it is 'part of their own language' (as opposed to 'not part of their own language' or 'foreign'). So it is possible for a concept to be foreign to a community while some members of that community can fact interpret it correctly.

3.2.10 The Robustness of Language

There are a number of basic reasons why language as a whole goes a long way when it comes to communicating concepts. The most important of these reasons are briefly listed below.

- People are able to carefully and accurately use language the way that is best fit for their particular audience at the time. They select appropriate words and structures from the linguistic toolkit at large.

- People are very good at applying conversational *repair* techniques, at seeking confirmation of their mutual understanding, and at gently steering each other towards a common understanding through a delicate game of linguistic coordination.

- People are very good active interpreters, seeking and adjusting interpretations even under the toughest of circumstances. The value of the educated guess should not be underestimated in this respect.

- Language carries a fair amount of redundant information, thus allowing for some missing or faulty interpretation to be countered by "backup" information that is more in line with the process of sense making.

- Because of its complex and above all *recursive* structural properties (its grammar), it is possible for people to use language to create a literally endless variety of utterances, employing a wide array of forms to shape delicate nuances and conceptual shades. The complexity of language structure, when faced in all its detail, has left linguists in awe and wonder (and has at times driven them to despair).

- If the need arises, people can spend so many words and sentences on one subject or situation that through many complementary bits of skeletal meaning, a complex, dense web of semi-experiential meaning is gradually woven. Consider for example, a book. It needs time to be read, much more so to be written, and it is never the final word on its subject; still, it manages to create a conceptual picture that is immensely complex and can convey ideas that in richness and complexity go quite far beyond prototypical speech acts such as commands and requests.

- People simply consider the limitations of language a fact of life, just as they accept that one cannot fly without the use of specialised tools, and that even then flight has its limitations. Language is a great tool, but it is a limited one, and we know it; the limitations do not keep us from using it constantly, successfully, and happily.

Importantly, some of the properties that make language so powerful are severely hampered, if not ruled out, if language is used in its "frozen" form. Freezing language, being based on a representational outlook, leads to emphasis on a limited, "skeletal" approach to language as a communicative tool because it necessarily strips away many of the linguistic devices that compensate for the conceptual limitations. This is why the distinction between "open language" and "closed language" use (see Chapter 2) is so important.

3.3 The Language Action Perspective on Communication

Let us now turn to a discussion of the –to us– most relevant points of a framework that has been of great influence to our ideas about language and (meta) communication. The framework is known as the Language Action Perspective or LAP. Its main theoretical pillar is Speech Act Theory (Austin, 1962; Searle, 1969), which has already been introduced in section 3.2.3. An important influence to the LAP is the work of Jürgen Habermas (Habermas, 1981). Researchers working within the LAP apply speech act theory mainly in the context of organisation, communication, and information systems. Our presentation is mostly based on work by Van Reijswoud, who uses speech act theory as the basis for a framework for the modelling of business communication (van Reijswoud, 1996). We also include fragments of (Hoppenbrouwers and Weigand, 2000).

We have already seen that the 'speech act' is a theoretical notion that was introduced by (Austin, 1962) to capture the observation that language is used to *do things with*. (Searle, 1969) further developed Austin's ideas.

Quoting Van Reijswoud p.29 : "... the primary object of analysis in the philosophy of language is not, as traditionally has been thought, the sentence but the speech act, which he defines as the *uttering* of a sentence in a particular *context*. [...] The production of the sentence taken under certain conditions is the *illocutionary act*, and it is this act that constitutes the minimal unit of linguistic communication that is able to establish coordination of action (Searle, 1969)".

The 'language acts' we consider relevant within our focus are the following:

```
Propositional Act:  S says to H in C that so-and-so
Illocutionary Act:  S does such-and-such in C
Utterance Act:      S utters e from L to H in C
Perlocutionary Act: S affects H in a certain way
```

In the definitions, S stands for Speaker, H for Hearer, e for an expression in language L, and C for the context of the utterance (adapted version of classification as presented in van Reijswoud 1996, Bach 1979).

The propositional act is the typical level of concern for traditional, proposition-oriented linguistics. The 'payload' of the propositional act is often referred to as the *propositional content* P of a speech act.

The illocutionary act adds to this both *illocutionary point* and *illocutionary force*. Illocutionary *point* refers one of the main types of illocutionary act. What these types are depends on the categories distinguished, and many have been proposed (for example, see the use of some more specialised speech act types in Section 3.4.2). We stick to the distinction as previously presented in Section 2.1.2 (van Reijswoud, 1996; Searle, 1979):

Assertives commit the speaker to something being the case (e.g., stating);

Directives try to get the hearer to do something expressed in the proposition p (e.g., questioning and commanding);

Commissives commit the speaker to some future course of action (e.g., promising);

Declarations (or declaratives) bring about a (new) state of affairs by merely declaring it (e.g., declaring);

Expressives express the speakers attitudes/feelings about a state of affairs (e.g., apologising).

Illocutionary *force* adds a *degree* in which the illocutionary point is expressed. "This distinction can best be illustrated with the two utterances: "You will leave the room" and "Leave the room!". Both utterances have the same propositional content, namely that you will leave the room; but characteristically the first of these sentences has the illocutionary force of a predication or a threat, while the second has the illocutionary force of a command" (van Reijswoud, 1996, p.29). Note that it is quite common in general theoretical descriptions to only deal with illocutionary point and leave illocutionary force to more in-depth linguistic analysis.
The perlocutionary effect of either utterance may, for example, be that the Hearer indeed leaves the room. Note that a grammatically incomplete utterance like "Leave!" may have a similar perlocutionary effect, even though the propositional content does not include an explicit reference to "the room". Nor is it considered to be a 'complete sentence' (generally the result of a complete propositional act). This is where the complex game of active interpretation comes into play, which involves both the propositional and illocutionary level.
As we have seen, communication is a social game, and language reflects this by nature. Social structures are essentially based on values, norms, and rules; the game involves a number of forces that determine the relationships between people, and therefore the way they interact. Habermas, in his theory of communicative action, addresses related matters.
Most importantly to us, Habermas introduces the distinction between three action types (see table 2.4, van Reijswoud p.52). "The first type of action oriented to success is called *instrumental action.* [...] Instrumental action is oriented to success through the technical control of impersonal problems. In this respect it cannot be counted as social action because it is nor really communicatively mediated. It is in the typical case set in written and technical languages rather than ordinary language in spoken form (Pusey, 1987)." [...] The second type of action oriented to success is *strategic action.* [...] Strategic coordination of action depends on the way egocentric interests fit together. [...] Instrumental and strategic action are opposed by *communicative action.* The basic aim of communicative action is that at least two participants achieve a common definition, or consensus, of the situation in which they find themselves. So, the actions of the participants in the conversation are not coordinated by egocentric calculations, but through acts of reaching understanding (Habermas, 1984, pp.285-6)" (van Reijswoud, 1996, p.51).
"In both a strategic and a communicative exchange of communication acts, the speaker raises *validity claims*, next to the proposition, which bring coordination of action about. The action types determine the behavioural alternatives of the hearer. The dominant claim in strategic action is the *claim to power.* The claim to power leaves the hearer only the possibility to answer "yes" or "no", there is no possibility for an equal argument.

In a communicatively oriented communication act, three validity claims are raised by the speaker. [...] The speaker claims that what he states is true (*claim to truth*), that his utterance is right in relation to the recognised normative context (*claim to justice*), and that his manifest expression of intentions is truthful (*claim to sincerity*; see van Reijswoud 1996 p.52). When engaged in communicative action, the hearer has three behavioural alternatives. The hearer can answer "yes" and "no" and has also the option to reply with a "yes, but...". In the last alternative, the hearer disagrees with one of the validity claims that has been raised by the speaker. The contesting of validity claims is an essential characteristic of communicative coordination of actions." (van Reijswoud, 1996, pp.52-3).

Most importantly, Habermas's distinction between strategic and communicative action, i.e. action based on a claim to power versus action based on the other validity claims, sheds light on some fundamental aspects of social interaction. They should, therefore, also be able to help us understand social aspects of meta-communication, and see how meta-communication relates to primary communication. We will return to this in the next section.

In addition, since the option to *disagree* with a speech act is seen as fundamental, this can be taken as the basis of a useful division between types/levels of conversation. Van Reijswoud develops this basic idea into his Transaction Process Model (TPM; van Reijswoud, 1996, p.95). In this model, which depicts "the process structure of communication for the purpose of the coordination of activities in organisations", three layers of conversation are distinguished: the *Success Layer*, the *Discussion-and-Failure Layer* (further called "Discussion Layer" in this thesis), and the *Discourse Layer*. The Success Layer is where communicative transactions take place that do not suffer breakdown. If, however, breakdown occurs (or threatens), the Discussion Layer is entered. At this level, "the participants in a communicatively oriented transaction process discuss whether the claims incorporated in a communication act and laid down in a transaction state are true, justifiable, and truthful" (van Reijswoud, 1996, p.97).

"However, in some situations a discussion relates to the background conditions of a group of related transaction processes. These discussions, called discourses, are dealt with in the discourse-layer [...]" (van Reijswoud, 1996, p.100). "Discussions about validity claims, on the one hand, relate to a particular instance of a transaction type, e.g. the order with the order number #93154. Discourses, on the other hand, abstract from particular instances of a transaction type, and focus on the underlying rules of a particular transaction type" (*ibid.*, p.161). In Figure 3.1, a simplified depiction of van Reijswoud's TPM is given, showing its three layers and the basic order in which they may be engaged.

A final point we want to make concerns the crucial relation between all validity claims and the *comprehensibility* of the symbols used in the utterance. Up to a point, validity claims (the claim to power included) need the people involved to successfully comprehend the meaning of the signs uttered. Comprehensibility of utterances conveying the meanings which in turn 'serve to carry out some goal', is crucial to the communicative effort as a whole (Hoppenbrouwers and Weigand, 2000). This is acknowledged by Habermas, who introduces an extra validity claim: the *claim to comprehensibility* (Habermas, 1981).

Nevertheless, the status of comprehensibility in speech act theory varies. For exam-

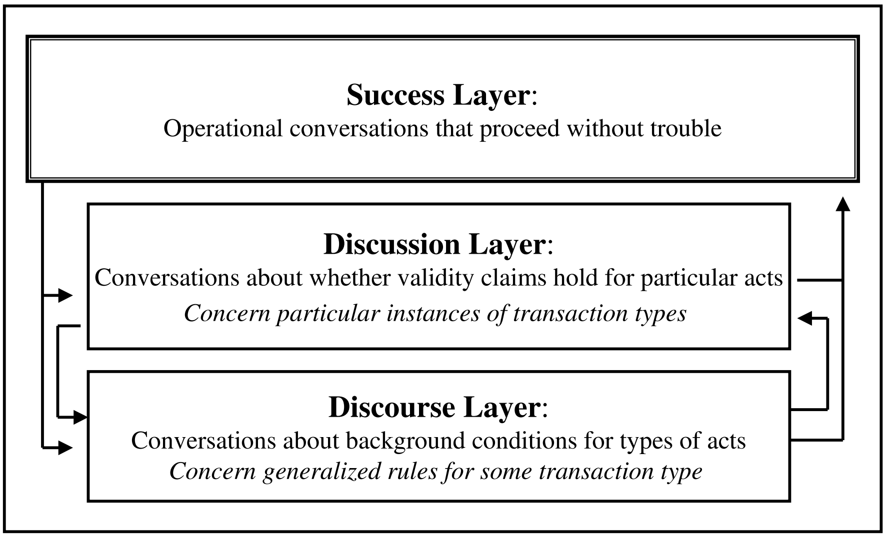

Figure 3.1: A Simplified Representation of the Transaction Process Model

ple, in some discussions of validity claims underlying transactions (e.g. van Reijswoud, 1996), the claim to comprehensibility is not adopted. Some others, however, do. For example, (Schoop, 1998) and (Schoop, 1999, p.65) embrace it; she argues that it is related to both Propositional Content and Illocutionary Force. We side with Schoop in adopting the claim to comprehensibility, which is relevant to our discussion of meta-communication.

Though we have necessarily left out many a theoretical point underlying the concepts presented (in particular, van Reijswoud's TPM has not been done full theoretical justice), we have now introduced the basic theoretical elements that are needed for our account of meta-communication below.

3.4 Linguistic Meta-Communication

We have arrived at what could be seen as the *piece de resistance* of this chapter: a discussion of meta-communication, which is the linguistic scene-setting for Freezing Language. We will revisit the theoretical assumptions introduced so far and in that light try to understand meta-communication and the freezing of language. This section is partly based on (Hoppenbrouwers and Weigand, 2000), in which an early version of our ideas was presented.

3.4.1 The Basic Mechanisms of Communication about Language

Let us start this section with a classic quotation (Winograd and Flores, 1986, p.66): "Conditions of satisfaction are not objective realities, free of interpretations of speaker and hearer. They exist in the listening, and there is always the potential for a difference among the parties. This can lead to breakdown [...] and to a subsequent conversation about the understanding of the conditions."
Most typically, if communicative transactions fail this is because there is disagreement about some aspect of the propositional content, which is at least successfully communicated. There is also the possibility that illocutionary load raises problems, and this too may be the subject of discussion. However, note that matters of illocution are generally expressed through intricate, complex symbols, and are not usually *explicitly* represented in information systems[16]. We therefore tend to pay most attention to matters related to propositional content and to relatively simple matters involving illocution.
We are in principle concerned only with *comprehensibility*, both of the propositional content and of expressions of illocutionary force (Schoop, 1999, p.65). We are thus interested in cases in which the language/utterances used notably fails to do its intended communicative job *because of some mismatch in linguistic or contextual knowledge between communicating parties*, leading to *misunderstanding*.
If comprehensibility is violated, three different situations may occur:

- Immediate breakdown occurs on account of clearly perceived incomprehensibility of some utterance; conversation for clarification may or may not be started.
- The listening/reading party *suspects* incorrect interpretation on her behalf, because of a perceived mismatch between what is said and what is to be expected in the wider context of the conversation. Again, she may or may not start a conversation for clarification.
- No misunderstanding is perceived and problems may arise only later, if at all.

We will focus for the moment on those cases in which the listener expresses or predicts the (suspected) occurrence of comprehensibility failure and thus enters into a 'conversation for clarification' (Winograd, 1988). Let us assume this covers all types of conversation that are prototypically initiated with the utterance "what do you mean with ... ?".
A crucial distinction now needs to be made. The most typical conversation for clarification involves discussion of the meaning of situated utterances, i.e. utterances that take place in a specific context (cf. Section 3.2.8). Such discussion primarily (but not exclusively) involves pragmatics. The other, rarer type of conversation for clarification deals with the *language system*, not utterances but the language forms and their (skeletal) meanings that are part of the persistent common ground. We can thus distinguish between *pragmatics-oriented* and *semantics-oriented* meta-communication.

[16] One of the main points of (Winograd and Flores, 1986) is that this should change in order for communication through ICT to be more successful.

Note that discussion of both general and communal linguistic conventions (Section 3.2.9) fall under semantics-oriented meta-communication. Also note that conversations about language systems may include the selection and definitions of concepts as they occur as *part of a medium system* (see Section 2.1.2).

As indicated before, both the semantic and the pragmatic level of meta-communication are of interest to us. Recognising which type we are dealing with in a particular meta-communication process is crucial for understanding its goals and how to support them. Just to realise for every 'frozen' concept to what degree it will involve pragmatic information in use, is a major step towards controlled, responsible conceptualisation. The degree of 'contextualisation' in concept *use* is (or should be) a crucial factor in determining the conceptualisation strategy that is to be followed.

To continue our basic theoretical discussion, there is a parallel between Van Reijswoud's TPM and the distinction introduced above. If we assume that meta-communication is covered by the TMP, it seems to hold that *pragmatic* matters (involving particular contexts) are usually dealt with in the discussion layer, whereas *semantic* matters (involving linguistic *conventions*) are usually dealt with in the discourse layer. Still, both semantics and pragmatics can be the subject of conversation in either layer. It is possible for a meaning-oriented conversation for clarification to originate in the discussion layer, mostly concerning pragmatics, and to move to the discourse layer only as the discussion deepens and touches upon differences of opinion concerning linguistic (semantic) convention. However, it is also possible that the discourse layer is entered immediately, for example if someone is plainly accused of violating a linguistic convention ("using a word wrongly").

In the previous section, we have seen that the discourse layer is associated with matters concerning validity claims. This raises the question of whether the standard validity claims (to power, truth, justice, sincerity, and comprehensibility) also apply to meta-communication. One would expect they do, so let us explore and exemplify. Consider the following (slightly adapted) citation from Lewis Carol's "Through the Looking Glass". HD is Humpty Dumpty, A is Alice; they engage in linguistic meta-communication.

```
HD:     There's glory for you!

A:      I don't know what you mean by ``glory''.

HD:     Of course you don't -till I tell you. `I meant ``there's
        a nice knock-down argument for you!'

A:      But ``glory'' doesn't mean ``a nice knock-down argument''.

HD:     When I use a word, it means just what I choose it to mean
        -neither more nor less.

A:      The question is whether you can make words mean so many
        different things.

HD:     The question is which is to be master -that's all.
```

Concerning the relation between the claim to comprehensibility on the one hand, and the claims to power, truth, justice, and sincerity on the other, we believe that in the case of meta-communication, there is more going on than a division in five claim types. Once the claim to comprehensibility is challenged and a conversation for clarification is entered ("I don't know what you mean by *glory*"), we enter into a new linguistic domain in which terminology but possibly also all other aspects of language are tuned to meta-communication. Let us consider the various validity claims within the context of meta-communication.

Claim to Truth

Whether or not the claim to truth holds in semantic meta-communication is mostly a matter of Ontology; we take a mentalist perspective and assume that indeed the claim to truth holds at the level of *general linguistic convention*: the basic knowledge that X *potentially* means Y to some existing someone in some hypothetical situation. So if meta-communication concerns semantic knowledge of general linguistic conventions, it involves a claim to truth, which might be proven empirically: by finding someone for who X means Y in some situation. For example, if someone believes that the symbol "Dog" refers to a type of quadruped that typically barks, bites, smells, and wags its tail, then it is *true* that "Dog" means *type of quadruped that typically barks, bites, and wags its tail* in at least one situation. Note that it may mean many, many other things as well, and that this may be expressed in numerous ways.

Claim to Sincerity

A possible exception to the possibility to assign a 'cognitive truth' to concepts concerns the *referential* meaning of truly *subjective* words. For example, it has often been observed that it ludicrous and impossible define "beauty" extensionally (i.e. defining which entities are beautiful and which are not). The claim to sincerity in linguistic meta-communication then applies to cases in which someone expresses something subjective by means of a word that is generally agreed upon as being a subjective word: a commonly agreed on semantic 'shell' designed to be filled with a subjective, emotion-based opinion, usually expressed about something situational (hence strongly pragmatic). Note, however, that at least *some* objective, shared part of the meaning of the word "beauty" belongs to (semantic) common ground, and thus is covered by the claim to truth and general linguistic convention.

Claim to Justice

The claim to justice concerns *communal linguistic conventions* about meaning, holding between a particular group of people. For example, knowledge of a communal convention of meaning may be represented as "I believe that in hypothetical situation S, person P belonging to group G would mean m with f". It involves meaning in a slightly different sense than that captured in general linguistic conventions: communal linguistic conventions are capable of *restricting* the semantic options offered by general conventions in a social domain. Such restriction is sometimes referred to as the use of a linguistic *register*. Note again that the type of knowledge involved is, in our view, still *semantic* and part of common ground (even if this is the common ground of a limited group of people). Consequently, discussion about meaning can concern the claim to truth as well as the claim to justice, depending on whether it relates to 'general linguistic convention', or to meaning selected through 'communal linguistic convention' applied with a domain.

Claim to Power

From the perspective of general linguistic conventions, power in meta-communication seems a difficult claim to substantiate, because thought control is not something that can be easily exerted, and general conventions are hard to 'control' in the best of circumstances. As we have seen, such knowledge is more fruitfully discussed in view of the claim to truth. However, from the perspective that language normally concerns *appropriate use* (depending on communal linguistic conventions and typically related to the claim to justice), the claim to power can certainly be made: someone may take a strategic (egocentric) instead of a communicative (common understanding) approach to word meaning. Our Humpty Dumpty example is a case in point.

If one of the communicating parties makes the claim to power (Humpty Dumpty: "The question is which is to be master – that's all"), and the other party (Alice, who is a polite English girl), does not want to trigger breakdown of the conversation, then the latter is forced to adopt the meaning of a word at least for the duration of the conversation; yet she may 'drop' that meaning the moment it ends. Note that a situation is possible in which power is exerted concerning linguistic conventions, while at the same time no such power can be exerted in other respects. Also note that Alice will probably remember Humpty Dumpty's definition and it will thus be added to her knowledge of possible meanings of "Glory", i.e. to her set of general linguistic conventions. Perhaps she will also remember that Humpty Dumpty is the one eccentric character who uses this meaning, thereby adding to her knowledge of communal linguistic conventions.

In the example, the claim to justice is, as it were, 'replaced' by the claim to power: a particular selected meaning is enforced, not agreed upon. Communal linguistic convention thus is respected in this case on the basis of fear for repercussions of some sort (HD getting nasty), not common understanding and agreement. Note that this implies that people may violate such a 'quasi-convention' when they believe they 'will not get caught'.

As we have seen, the claim to power seems to be applicable to socially restricted domains of vocabulary, not to general linguistic conventions. This implies that, though the latter can certainly be subject of discussion in meta-communication, communal linguistic conventions are more relevant to our current discussion because we focus on *agreements* about language (explicit or not), which quite emphatically involve the social dimension. Consequently, the claim to justice prevails in our further discussion of linguistic meta-communication; yet it may in some cases be replaced by a claim to power.

Texts can play a role in this in two ways: they may enforce an explicit, skeletal, decontextualised form of communal linguistic convention (meta-linguistic texts, for example dictionaries), or they may provide a more implicit conceptual background against which the common ground in all its complexity takes shape (more general, non meta-linguistic texts). *Interpretation in context*, taking place in the mind of the individual, cannot realistically be subject to the claim to power: only skeleton meaning seen as a communal linguistic convention can.

Since we focus on explicit, deliberate conceptualisation rather than intuitive (meta)linguistic processes, we conclude that meta-communication in its explicit form may indeed be based on a claim to power, however weak it may be. If, for example, agreements are made on terms to be used throughout a company, more often than not some form of *authority* is associated with them. In some cases, such exertion of power may not be strongly felt because it may not go against the grain of common language use. However, it may in fact be the case that 'communicatively agreed' language is replaced by 'strategically imposed' language: language that does not serve a common communicative interest (is not subject to common agreement), but is imposed because of a strategic goal of someone wielding power. One may thus be forced to replace an intuitive communal linguistic convention by a counter-intuitive one.

For example, a company may actively guard the use of legally correct terminology in its documents. The use of legal terms may clash with the terminology used in general in the company, and may even cause some frustration and communicative trouble. People at the work floor may dislike it and even avoid it if they think the will get away with it; still they will use the 'company terms' in documentation and (importantly) when they build terms into information systems.

In Figure 3.2, we illustrate the move from a challenge to the claim of comprehensibility to the specialised mode of 'a conversation about comprehensibility' (i.e. meta-communication), in which the claim to comprehensibility may again be challenged –ad infinitum. This echoes our observation in Section 3.2.7 that cognitive meaning cannot be described through language: we cannot break through the surface of our representations. We can only try hard to create some common ground, and then seek confirmation of there being *enough* common meaning for achieving our communicational goals in that particular situation.

Claim to Comprehensibility

Finally, let us reconsider the claim to comprehensibility. As we have seen, it is a violation of the claim to comprehensibility that may trigger meta-communication; as we focus on linguistic meta-communication, perhaps we should explicitly restrict ourselves to *linguistic* comprehensibility. Given the conventional nature of language, it can very well be argued that the claim to comprehensibility is in fact a wrapped up combination of the claims to truth, justice, sincerity, and power in a particular *mode*, namely that of linguistic meta-communication. Hence, it is questionable after all whether the claim to comprehensibility is a 'true', fundamental validity claim. However, we choose to maintain the distinction of the claim to comprehensibility in our discussion because it so effectively isolates a particular type of validity claim discussion; one that clearly and intuitively reflects our particular area of interest.

Interestingly, at the level of meta-communication, the claim to comprehensibility again holds. Since meta-communication is a linguistic domain in its own right, it is not impossible that this language, and utterances composed using it, include

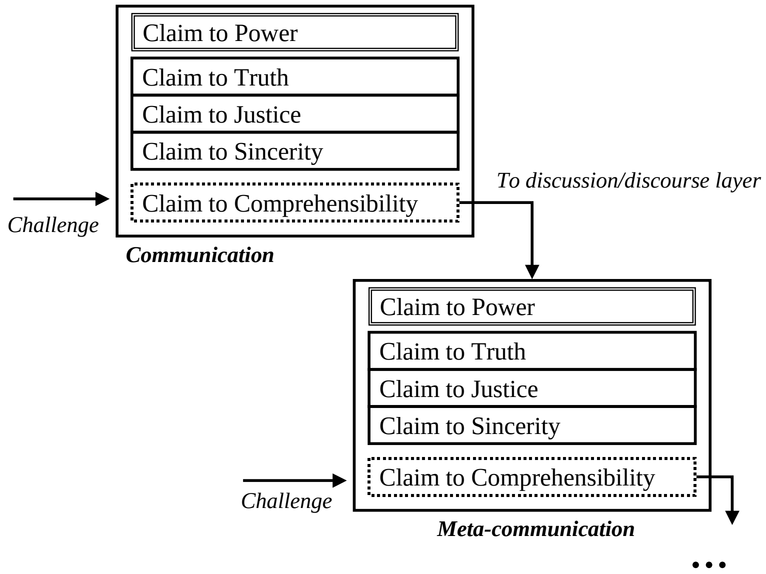

Figure 3.2: Meta-communication triggered by a challenge to the claim to comprehensibility

elements not shared by all parties involved, which may lead to breakdown, (meta-) meta-discussion, etc. For example: we have seen how problematic the notion of "meaning" itself can be.
Summarising the validity claims as applied to meta-communication:

```
truth:             X may means Y because that is what we know it
                   does to someone in some situation

sincerity:         X means Y because that is the way I feel about
                   it

justice:           X means Y because that is what some set of people agree it
                   does in some domain

power:             X means Y because I say so

comprehensibility: I believe you know that X means Y
```

So in our perspective, one claim is central in linguistic meta-communication, which is the claim to justice, applying only to communal linguistic conventions. It is this claim that is primarily (but not exclusively) dealt with in linguistic meta-communication that takes place in the discourse layer of the TPM. The claim to truth plays a role when general linguistic conventions are concerned. If subjective

aspects of meaning are discussed, the claim to sincerity may become relevant. The claim to power may also come in, typically replacing the claim to justice. The claim to comprehensibility really is a composite claim applied to the discussion of linguistic conventions.
To conclude: indeed linguistic meta-communication can be analysed in terms of (challenges to) validity claims and conversations about validity claims. This provides an important theoretical means for understanding and explaining the social nature of language (linguistic conventions). The distinction between the claims to truth and justice provide an important backing for the distinction we make between general linguistic conventions and communal linguistic conventions. The claim to sincerity hands us a link with deeply individual and therefore subjective aspects of language and meaning, outside immediate reach of the community but nevertheless embedded therein. Finally, recognition of the 'claim to power' as applied in linguistic meta-communication entails that authority is another fundamental aspect of language and languaging. Thus, the application of Habermassian theory as included in the Language Action Perspective contributes to our understanding of the essence of linguistic meta-communication, and of its relation to more generic forms of communication and meta-communication.

3.4.2 Definition from a Speech Act Perspective

We proceed by briefly presenting a speech-act related, functionalist view on *definition* as put forward by (Viskil, 1994), as a concluding part of his elaborate and in-depth discussion of definition. Some of the findings mentioned by Viskil (most relevant to this thesis) are:
"Precision of definition is always connected to a certain goal or a certain context. No statement can be expected to hold absolutely or universally; only that it is precise enough for the goal and context for which it was brought forward" (*ibid.*, p.3) [translated from Dutch].
"The break-off point of definition is determined intersubjectively: language users participating in communication together decide whether enough clarity has been achieved –for what they want to achieve" (*ibid.*, p.3).
Note that both statements are entirely in line with our argumentation thus far.
Viskil, after a thorough comparison of many analyses of definition, distinguishes four main types of definition:

Lexical definition Speech-act type = language use assertion; describes word meaning as attached to a word in usage by language users (observation). For example, "In Dutch, X means Y"; the definition concerns the meaning of a word as it is perceived by 'people in general'; de person defining takes an objective attitude. *For example, in an English dictionary a description can be found of what English speaking people generally think the word 'embarrassment' means.*

Stipulative definition Speech-act type = language use clarification; records word meaning with the purpose of clarifying (statement). For example, "With X, I mean Y"; the definition concerns a particular instance of use of a word; the person defining describes subjectively what the word means to her. *For* *example, we defined 'concept' as the combination of a linguistic form and a meaning. This definition differs from the one given in many other publications, but we* stipulate *the meaning we give to it* in this particular book, *clarifying our particular use of the term.*

Stipulative-lexical definition Speech-act type = language use clarification through language use assertion: combines stipulative and lexical definition in that it is an assertion (statement) about word meaning explicitly grounded in observed usage. For example: "X means Y in Dutch, and therefore it also means Y in this particular case"; the definition concerns a particular instance of use of a word, but the person coining the definition chooses to stick to the 'common' meaning; the subjective agent conforms to previous objective observation defined meaning, usually as expressed in some authoritative dictionary. *For example, we might feel it is necessary to clearly state the meaning of 'community', but decide to take a general dictionary definition and present this as the meaning we prefer to give to the word.*

Factual definition Speech-act type = assertion; empirical description of a fact; not meta-language. A definition-like description is used to explain some object, situation, state, process, etc.; however, such a definition has no primarily meta-communicational function. *For example, encyclopedias contain essay-like entries describing something that is usually referred to in the entry heading by a single word or a short nominal phrase. However, more than the word meaning is described: the entry is not a* linguistic *definition as such.*

Note that in organisational contexts, definitions are often heavily loaded with general, normative statements. Think about the definition of words like "employee", or of "staff member". If a definition influences the obligations and authorisations of the members of the organisation, then the definition makes a claim about the appropriateness or rightness of these obligations. A discussion about such a claim has to be distinguished from a conversation for clarification proper, i.e. it does not belong to linguistic meta-communication.

Viskil does not show any particular concern with operational matters of definition as part of a general language game as discussed in Section 3.2. In particular, he does not take into account that definition requires considerable effort and resources, and is not something that anyone can be asked to do in any situation. While this may not be a relevant issue within Viskil's research goals, for our current purposes we need to add such considerations to our view on definition.

Finally, if a definition has to conform to the opinion of a group, there has to be some sort of discussion about it; agents involved will have to come to an agreement. This process is not dealt with by Viskil, but it is relevant to our current effort. Without going into procedural matters with respect to groups reaching agreement on definitions here (for more, see Sections 5.2.1 and B.26), we observe that within the frame of our ideas on language and meaning as discussed in Section 3.2, the nearest we can come to 'agreement about the meaning of a concept' is to strive for *linguistic reassurance.* Individuals can only be *sufficiently sure* of what is meant by a language form in some situation, as a result of the observed use that has been made of it in context, or given meta-communication that may have taken place about it.

Even a carefully constructed definition can never do more than provide linguistic reassurance that is strong enough to be acceptable; absolute guarantees of proofs are *theoretically* impossible. Fortunately, linguistic reassurance is often quite good enough to enable *practical* agreement about language meaning.

3.5 Summary

As mentioned in the introduction to this chapter, Chapter 3 is aimed at answering research question Q1.4: “What are the linguistic mechanisms and factors underlying the description of, need for, and course of the processes of freezing language?”. The three sub-questions (explaining description, need for, and course of processes of freezing language) were not answered separately; we rather presented a fundamental framework working towards the topic of *meta-communication*, which is the main notion that enables us to explain processes of freezing language. We conclude the chapter with a summary, which is a reasonably concise answer to question Q1.4. However, in some respects the answer will continue in the next chapter, though our focus shifts to the ICT domain.

Language is a tool that helps people communicate, and thereby to coordinate their thoughts and actions. The tool of language is wielded largely intuitively: people are generally not aware of the intricate skills and knowledge they apply when they use language.

Language uses symbols to convey meaning: both elementary symbols (words) and complex symbols (sentences, even texts). The combination of a form with a meaning we call a *sign*. Our definition of *concept* is identical to that of sign: a combination of a language form and a language meaning. A word is the most typical, basic example of a concept, but an entire sentence is also a concept. Concepts need to be shared between people to make up a language; the agreements about a language between people are *linguistic conventions*. They are implicit by nature, but sometimes they are made explicit.

Mainstream linguistics focuses on describing structures in language, mostly rules of grammar (syntax). Some branches, however, focus on language form in immediate connection with *what it does*. We share this latter, functional perspective, but in our fundamental outlook on language meaning we take functionalism even further: we look upon meaning from an action perspective; we wonder what a word does when it ‘means’.

What utterances ‘mean’ is usually divided in two theoretically separated aspects: the ‘propositional load’ (regular, representational semantics and pragmatics) and the ‘illocutionary load’ (speech acts). We view *both* in terms of the action perspective. It is quite common to do this for illocutions, but not so for propositions.

We follow those thinkers who emphasise that language is not a carrier of meaning in the sense of ‘literally transmitting thoughts’. Language as such is capable only of conveying ‘skeleton meaning’. When language is *interpreted*, the interpreting person actively and creatively combines knowledge of the current situation with common knowledge (including linguistic knowledge) to arrive at a highly *contextualised* meaning. Such meaning cannot fully be expressed in natural language, certainly not in one brief utterance. ‘Real’ meaning, that is connected with the ongoing *experience* of living and not with just talking about it, only exists in the mind of individuals, and it constantly changes as we and the world around us change. We can only ask an individual to *describe* what meaning 'looks like' for her, at a particular moment in time, and the description can never be complete and perfect. So we can never really capture meaning in full by means of representations.

Words emerge (partly) through the need to single out some persistent idea or thought (cognitive concept) and then combine the meaning with a form in language (linguistic concept) so one can communicate about it and hence deal with some matter of interest. The concepts that need to be made explicit this way are therefore the ones that are needed for expressing some problematic or otherwise 'relevant' aspect of experience. Linguistic conceptualisation is driven by the *need* to have particular signs. Such need typically differs per situation, and may vary greatly between domains of experience. We should keep this in mind in our thinking about conceptualisation processes.

Language works by invoking changes in the 'mental picture of the world' of individuals. It does not convey 'real meaning', but triggers associations that are *similar enough* between people for them to come up with a useful interpretation of what someone has said to them. How this works exactly, on a cognitive level, is not our concern. We view the human mind as such as a 'black box'.

To describe meaning is difficult, even for a specialist, partly because knowledge of meaning is typically *tacit*. But talking about meaning is still important and useful. The problem this poses requires a fresh look at meaning description: perhaps it is better to first look at the whole process of meaning discussion and description –the whole 'language-and-meaning game', as it were– and only when we understand what is going on decide how we will describe meaning *for a particular situation*. It sometimes works better to keep away from meaning description altogether, but at times we do want to be very strict and precise in meaning description. Which choices to make, how to support making them, and how to carrying out the actual describing if called for: that is what we are after.

Language works because people share much knowledge, including linguistic knowledge. Such knowledge involves linking the context to interpretation (pragmatic knowledge) and linguistic conventions (semantic knowledge). The total body of shared knowledge of communicating individuals we call their *common ground*. This is what 'context' in a linguistic sense boils down to: the knowledge the speaker and hearer(s) share that is the basis of similarities in interpretation in language users, and hence of coordinated action (either mental or physical).

Linguistic conventions, that as we have seen are essentially tacit but can for a limited part be made explicit, include a special kind of convention concerning which other linguistic convention are linked to a certain *community*. For example, language users know which words typically *belong to their language* –or not. This special knowledge we call *communal linguistic convention*. Communal linguistic conventions help people make sense of what they hear because it makes it possible to select special meanings of words that hold only for a particular group of people.

Language is an impressive means to invoke meanings in a constantly changing, multifaceted world of concepts. There are a number of reasons for this. Perhaps most importantly, people are very good in active interpretation, as long as there is sufficient common ground. In addition, people are able to steer each other towards mutual understanding, and *repair* their conversation if intended meaning threatens to be lost. People also *adjust* the language they use to fit the situation at hand: they adapt both to the person they are communicating with and the situation in which the communication occurs. However, some of these powerful properties of language are damaged when language is *frozen*.

Our view on language is largely inspired by the Language Action Perspective. Within this frame of thinking, speech acts (of various types) are founded on *validity claims*. Different types of speech act emphasise different validity claims, e.g. a 'claim to truth' or a 'claim to sincerity'.

Also within the LAP, a Transaction Process Model is used that distinguished three layers of conversation: the 'Success Layer', the 'Discussion Layer', and the 'Discourse Layer'. Operational communication takes place in the success layer, discussion of whether the communication on the operational level satisfies the validity claims takes place in the discussion layer, and conversations about the background conditions of the validity claims are handled in the *discourse layer*.

We project the notion of meta-communication, which is *communication about communication*, on the TPM. We restrict our discussion to meta-communication *about language* (linguistic meta-communication). Conceptualisation is an important 'linguistic meta-communication process'.

We observed that 'pragmatic meaning' is typically (but not exclusively) discussed in the discussion layer, and 'semantic meaning' typically (but not exclusively) in the discourse layer. Recognising the validity claims helps greatly in understanding the social aspects of meta-communication.

The dominant validity claim in linguistic meta-communication is the claim to justice, which is related to the communal-conventional component of language. In addition, a somewhat controversial validity claim, the *claim to comprehensibility* is embraced by us as a useful notion with respect to linguistic meta-communication, which is seen as being triggered by a challenge to the claim to comprehensibility. However, on a fundamental level the claim dissolves into the four basic validity claims (justice, truth, sincerity, power); the claim to justice then typically prevails. Nevertheless, we do not dismiss the claim to comprehensibility because it applies the various validity claims to a particular type of conversation central to our discussion.

Finally, definition can also be analysed from a functional perspective. We distinguish four different types: stipulative, lexical, stipulative-lexical, and factual definition. Definition is not something that is done easily, and as we have seen can only provide a limited certainty concerning the meaning of a particular concept. Fortunately, this is often enough.

Chapter 4

Language and Meta-Communication in Organisations and ICT

Having advocated a general view on meta-communication in order to better cope with the frozen language issue, and having presented some theoretical underpinnings of (linguistic) meta-communication, we are now ready to turn our attention to the particular type of domain to which we apply our view on language and meta-communication: ICT-supported organisations. The freezing of language has been defined as "the process of specifying the set of concepts of the language to be built into a medium machine" (Section 2.1.2). The freezing of language is tightly related to the use and development of ICT, but also to the organisations the language is to be used in. After all, 'frozen language' needs to be (and remain) aligned to the organisational language environment. We will develop some theoretical concepts that should help us reach our main goal as previously formulated. Let us consider again the following research questions:

Q1.1 What are the processes involved in freezing language, and how can we describe them?

Q1.2 What are the environments in which the freezing of language may take place, and how can we describe them?

Q1.3 How do processes and environments for freezing language relate to and influence each other?

By the time we reach the end of this chapter the reader should at least have basic answers to Q1.2 and most of Q1.3. The answers to Q1.1 will be given in Chapter 5; further answers to Q1.3 will emerge partly in Chapter 5, but mostly in Chapter 6. Chapter 4 is structured as follows. We first explain our view on "the languages of organisations", and explore some notions that are crucial for identification and demarcation of domains relevant to our effort. We expand our previously introduced formalised conceptual framework accordingly. This provides some of the fundamental concepts we need to describe the environments "in which freezing language may take place" (Q1.2).

Next, we elaborately consider some types of language and language use to be distinguished, enabling us to analyse the various functional aspects (requirements) of language that can form a basis for goal-driven manipulation and even the design of languages. We thus extend our exploration of the environments of conceptualisation with an analysis of the sorts of product that conceptualisation may (or should) deliver.

We then once more discuss meta-communication, now focusing on how it fits into ICT-supported organisations. This should enable us to more thoroughly understand the language-and-meaning games played in and between language domains. We consider drives behind ICT-related meta-communication, and provide a typology of meta-communication that makes it possible to distinguish situational needs, possibilities, and approaches in the selection, creation, and discussion of language as it takes place in ICT-intensive working environments. In addition, some detailed examples of the particular types of meta-communication are presented, and their role as communicational problem-solvers is explained. This provides ample illustration concerning our conceptual framework, the language use typology, the typology of meta-communication, and the relations between them. It also demonstrates the real and influential presence of conceptualisation processes in everyday ICT practice.

The various concepts and notions introduced in this chapter will be incorporated into a coherent descriptive framework in Chapter 6.

4.1 Language and Organisations

When they agree to cooperate, people may decide to create and adhere to a system of mutual agreements about who does what, when, and coherently so: to *coordinate* their actions. Loosely speaking, a group of people involved in such a cooperative structure is called an *organisation*. Whereas it is imaginable for people to cooperate without communication or language, it is rather unlikely for any substantial, explicitly shaped organisation to be formed without it.

Whereas it is quite possible to view organisations as 'real' entities (as is usually the case), there exists another view on organisations that centres round communication. According to this view, communication is not just an important activity that takes place within organisations: organisations essentially come into existence, function, and are maintained by means of communication and language. A number of (quite different) versions of this basic viewpoint have been developed, for example (March and Simon, 1958; Taylor, 1993; van Reijswoud and Dietz, 1999). However, they all seem to have in common the idea that communication is so essential to organisations that it is the chief underlying mechanism of which organisations *consist*, and that the products of communication (utterances, texts, or even artefacts in a broader sense) are to be seen as the tangible reflection of organisation. We will not go deeper into this matter, but conclude that at the very least, communication and language are a crucial *enabling instrument* for organisational action. What is more, language is both a *means* to organise, and a *product* of organisations: not only are concepts used to perform organisational communication, they are also formed (conceptualised) within the frame of the organisation they are used in, in order to support them.

So language as a means of organisation is formed by and adjusted to the organisational situation. Consequently, organisations are often said to have 'their own language'. What is special or 'their own' about an organisational language is not that its employees deploy a special grammar or pronunciation: without exception, some standard *national* language like English or Dutch forms the background for organisational language, and conventions of grammar and pronunciation are drawn from those national languages. In virtually all circumstances, *organisational* languages are particular in terms of *words and word meaning*: specially selected or created (sets of) concepts may be used or preferred, usually with specialised meanings. Also, specific, deviant meanings may be associated with word forms that commonly occur in the standard language.

Sometimes it is possible to point out how it is that particular words become part of organisational language. They may, for example, feature in some authoritative text, for example in a policy statement or a company handbook. Such texts can be of great influence to the words used in an organisation. In rare cases, words may become explicit beyond their mere use in texts: they then become subject of meta-linguistic decision making and are *explicitly conceptualised*. Usually this means that a particular word is designated as the 'proper' term reflecting a particular concept, and in many cases such a decision is complemented by the authoring of a definition of some kind. Specialised words that have an explicitly described meaning we call *terms*, the collection of terms in some domain we call its *terminology* (Wright and Budin, 1997, 13-4).

4.1.1 Terminological Domains and Discourse Environments

To presume that a terminology typically belongs to a single organisation would be rather naive. Terminological domains are as prolific as they are diverse, and depending on the availability of standards and the presence of traditions, they may stretch out across many organisations, even internationally. For example, terminology to describe parts and materials in machine industry is largely standardised. In extreme (but not uncommon) cases, there are even ISO standards for terminology (Strehlow, 1997a). On the other hand, terminology *can* be organisation-specific, and even specific to a particular department or other group within an organisation. Though it is generally considered practical to talk about terminologies and *sets* of terms, this is not always the most natural perspective. "Terminology" is a typically meta-linguistic and rather abstract notion. It is quite possible for an individual word to belong to several terminologies, and also to be used extensively outside explicit terminology; this would not hinder the use of the term, though it could well make meta-communication about it more complex.

Yet on the other hand, words do not 'mean' entirely by themselves: they are part of a cognitive network of concepts with semantic relations between them (Cruse, 2000, pp.127-8). However, this view is typically related to the concept of *vocabulary*, which we define as the complete set of concepts known by some *individual*. Nevertheless, at times concepts can be optimally described (for meta-communication) as part of a 'concept system' (for example, a concept tree, or a concept network). Usually this involves some kind of graph (Wright, 1997; Roulin, 2001).

Apart from looking at terminological domains (focusing on a coherent set of terms related to some topic or theme), we can also look at particular *Discourse Environments* (DEs) and analyse the various terminologies that may be used in them. This enables us to capture the observation that various terminologies may be used alongside each other in a particular situation, and that terms from different terminologies may occur even in one and the same utterance.

In (Weigand, 1990, p.182), the theoretical concept of *Environment of Discourse* (EoD) is introduced, in contrast with the more traditional *Universe of Discourse* (UoD; *ibid.*, p.12; also Halpin, 1995). Put briefly, the UoD is 'the world (or universe) we are interested in talking (or discoursing) about'. It is a 'conceptual world': a world about which people communicate, and which may be described through the naming of entities and relations. This view implies that if one makes a model of the domain, one also (almost implicitly) makes a model of the language used to describe that domain. The Environment of Discourse adds to the UoD viewpoint in that it includes the "social agents involved [in discourse], the communication links between them, interaction patterns and the relationship between information and action" (Weigand, 1990, p.182).

Within our current perspective, which puts *meta*-communication central, we prefer to introduce a different but strongly related notion. At some risk of creating confusion with the 'EoD', we coin the concept of *Discourse Environment*. A DE is an environment in which the dynamic processes of communication and meta-communication take place. It includes agents, artefacts, vocabularies, terminologies, media, and various other entities that play a role in (meta-)communication.

On a general, intuitive level, there admittedly is little difference between an EoD and a DE: they both stand for 'environments in which discourse takes place'. The difference is a matter of *focus* and the way the concepts are used as part of a descriptive framework: DE descriptions reflect a broad, dynamic, 'architecture level' perspective on language, prominently including meta-communication, where the EoD as presented by Weigand presumes a more static, representational view on language. On the other hand, the DE as we present it shows far less detail than the EoD when it comes to description of operational, communication (agent networks; who communicates with who). We will further define the DE in the next section, and present a concrete model for DEs in Chapter 6. Ultimately, it seems possible to unify the notions of EoD and DE, but we refrain from doing so here in order to keep our focus clear.

Terminological domains or terminologies, then, are sets of *explicitly described terms that have in common that they are related to some particular topic, situation, or group*. In contrast, discourse environments are essentially communicative environments in which sets of *people* enter into organised communicational activity. Though it is not typical, it is quite *possible* for a DE to have its own, specific terminology, in which case the terminology concerns the group and situation that occur in the DE. In any case, we consider DEs to be the linking pin between individual people and terminologies.

It is of course not the case that because two groups of people are involved in a similar activity, they automatically speak the same language. For the latter to be the case, the groups also need to be in some way cooperatively related: to be both capable and willing to use, and if necessary learn, a certain shared vocabulary or terminology. If a situation occurs in which two groups come together to communicate (usually via some agents representing larger groups), a new discourse environment comes into existence, with its own dynamics and common goals. It may be the case that conventions or agreements in one DE then *carry over* to another DE. Such conventions or agreements may be made explicit or not, and meta-communication between DEs may be organised or not.

4.1.2 Discourse Environments and Medium System Language Communities

The observation that many DEs may occur in an organisation gives rise to the question of what theoretical concept might stand for the larger whole in which various DEs occur and interact. We prefer not to speak about 'organisations' but introduce the specialised notion of the Medium System Language Community or MSLC. An MSLC represents all environments for communication in an organisation that make use of the same medium system or play a role in its development or support. Thus, if a DE touches upon a medium system, the agents within the DE belong to the MSLC of that system. Within the MSLC, people may communicate *through* its information system as well as communicate *about* its information system. The combined DEs in an MSLC include environments of operational communication (typically by 'users', but actually between all imaginable stakeholders) as well as all types of meta-communication. The MSLC thus covers quite a range of people. Anyone who finds herself in a DE that somehow includes a concept that is also involved in the medium system that is used or specified in that DE, belongs to the MSLC of that system.

Note, however, that agents who are involved not with the *concepts* but strictly with the *forms representing them* (typically programmers or system operators) are in principle not part of the MSLC: they deal with bare *data* and thus are not involved with linguistic concepts in the purer sense. However, it is quite common for agent concerns and tasks to blend (for example, programmers often do get involved with functional design), so in practice it regularly occurs that technical agents do in fact play 'meaning games' alongside other members of the MSLC.

Finally, consider 'texts'. We view texts as 'artefacts representing language utterances'. A *language artefact* then is the persistent physical occurrence of a (complex) language form. A 'persistent physical occurrence' might be defined as 'a physical occurrence that remains intact after the time that is required for a single perceptive intake (by a reader/hearer) of the complete language form in question'.

4.1.3 Extending the Conceptual Framework

We have previously distinguished and defined a conceptual framework (Section 2.1.2). We repeat its summary in table 4.1, and extend the framework as follows.

The language community of an information system used to implement a medium is defined as the *set of agents who are in some way functionally involved with the information system*. As mentioned, this involvement may pertain to the use of the information system for communication purposes, or any activity related to the functional development or description of the information system.

Concept	Explanation	Formal Definition
$\mathcal{CO}$	all decontextualised linguistic concepts	–
$C_L \subseteq \mathcal{CO}$	all linguistic concepts in language L	–
$\mathcal{MG}$	all linguistic concept meanings	–
$\mathcal{FO}$	all linguistic concept forms	–
$\mathsf{Form} : \mathcal{CO} \rightarrow \mathcal{FO}$	a linguistic concept has a form	–
$\mathsf{Meaning} : \mathcal{CO} \rightarrow \mathcal{MG}$	a linguistic concept has a meaning	–
F_L	all concept forms in language L	$\{\mathsf{Form}(c) \mid c \in C_L\}$
G_L	all concept meanings in language L	$\{\mathsf{Meaning}(c) \mid c \in C_L\}$
$\mathcal{SA}$	all possible speech acts	–
$\mathcal{CN}$	all possible contexts for speech acts	–
$\mathsf{Concept} : \mathcal{SA} \rightarrow \mathcal{CO}$	a speech act has a decontextualised concept that it conveys	–
$\mathsf{Context} : \mathcal{SA} \rightarrow \mathcal{CN}$	a speech act has a context	–
$A_L \subseteq \mathcal{SA}$	all speech acts that may be produced through language L	–
P_L	all concepts that may represent a complete speech act in language L	$\{\mathsf{Concept}(a) \mid a \in A_L\}$
U_L	all forms of concepts (*utterances*) that may express complete speech acts in L	$\{\mathsf{Form}(c) \mid c \in P_L\}$
$\mathcal{MS}$	all medium systems	–
$\mathcal{ME}$	all media	–
$\mathsf{Language} : \mathcal{MS} \rightarrow \mathcal{LA}$	all medium systems have an associated language	–
$\mathsf{Medium} : \mathcal{MS} \rightarrow \mathcal{ME}$	all medium systems have an associated medium	–
C_S	conceptual bandwidth of a medium system	$C_{\mathsf{Language(S)}}$

Table 4.1: Overview of conceptual framework (repeated)

The concept of 'agent' on which the concept of language community depends is defined an individual (human, or possibly some other intelligent entity) whose use of language is somehow of interest to us. This set is denoted as $\mathcal{AG}$.
The set of users of a medium systems can be defined more precisely as the set of agents who perform or undergo speech acts via the medium system. As explained in Section 2.1.2, speech acts normally involve various agents, minimally an agent who is the speaker and one or more hearers. Let $\mathsf{Agents} : \mathcal{SA} \rightarrow \wp(\mathcal{AG})$ be a function yielding the set of agents that are involved in a speech act. The set of users of a medium system S may then be defined as:

$$\mathsf{Users}(S) \equiv \bigcup_{a \in A_{\mathsf{Language}}(S)} \mathsf{Agents}(a)$$

Let $\mathsf{Describers} : \mathcal{MS} \rightarrow \wp(\mathcal{AG})$ be a function that defines the set of agents who play the role of a describer of the medium system at some point in time during the system's life cycle. This allows us to define the concept of medium system language community (MSLC) more precisely as:

$$\mathsf{MSLC}(S) \equiv \mathsf{Users}(S) \cup \mathsf{Describers}(S)$$

A discourse environment represents a part of a particular kind of 'language-and-meaning game' in which we are interested. Within our current scope, a discourse environment is identified by the combination of a set of agents and a set of medium systems. Let $\mathcal{DE}$ be the set of discourse environments. The associated set of agents is presumed to be provided by the function: $\mathsf{Agents} : \mathcal{DE} \rightarrow \wp(\mathcal{AG})$, while the set of associated medium systems (being the selected scope of a discourse environment) is presumed to be provided by: $\mathsf{Scope} : \mathcal{DE} \rightarrow \wp(\mathcal{MS})$. The set of agents who are associated to a discourse environment should be within the scope of the specific domain environment. In other words:

Axiom 2 Let $d \in \mathcal{DE}$, then:

$$\mathsf{Agents}(d) \subseteq \bigcup_{s \in \mathsf{Scope}(d)} \mathsf{Users}(s)$$

In addition, we presume the *vocabulary* of an agent, being the set of (internalised, 'mental') concepts available to an agent (i.e. being part of her linguistic knowledge), to be provided by: $\mathsf{Vocabulary} : \mathcal{AG} \rightarrow \wp(\mathcal{CO})$. We overload this function with a more refined version, which takes into consideration the discourse environments in which an agent participates: $\mathsf{Vocabulary} : \mathcal{AG} \times \mathcal{DE} \rightarrow \wp(\mathcal{CO})$. This gives rise to the following three axioms.
If an agent is part of a discourse environment, then the vocabulary of this agent with regard to this discourse environment cannot be empty:

Axiom 3 $g \in \mathsf{Agents}(d) \Rightarrow \mathsf{Vocabulary}(g, d) \neq \emptyset$

Conversely, if an agent possesses a vocabulary with regard to some discourse environment, then she should be regarded as being part of that environment:

Axiom 4 $\mathsf{Vocabulary}(g, d) \neq \emptyset \Rightarrow g \in \mathsf{Agents}(d)$

Any concept belonging to the vocabulary of some agent with regard to some specific discourse environment should belong to the general vocabulary of this agent:

Axiom 5 Let $g \in \mathcal{AG}$ and $d \in \mathcal{DE}$, then:

$$\mathsf{Vocabulary}(g, d) \subseteq \mathsf{Vocabulary}(g)$$

Interestingly, we can now (re)define the notion of UoD within our own conceptual framework as: $\mathsf{UoD} : \mathcal{DE} \rightarrow \wp(\mathcal{CO})$ where:

$$\mathsf{UoD}(d) \equiv \bigcup_{g \in \mathcal{AG}} \mathsf{Vocabulary}(g, d)$$

Thus, the UoD is now rooted in the vocabularies of the agents in the DE.
Next we define *terminology*, which is the explicit, 'organised' counterpart of individual-based vocabulary. A terminology *may* belong to some specific discourse environment. Let $\mathcal{TM}$ represent the set of available terminologies. We presume $\mathsf{Terminology} : \mathcal{DE} \rightarrow \wp(\mathcal{TM})$ to be a function yielding the terminologies that are associated to some discourse environment. Each terminology consists of a set of terms and possibly a set of descriptions of the meaning of these terms:

$$\mathsf{Terms} : \mathcal{TM} \rightarrow \wp(\mathcal{FO})$$

$$\mathsf{Description} : \mathcal{TM} \rightarrow (\mathcal{FO} \rightarrowtail \mathcal{FO})$$

The terms that receive a description in a terminology, must indeed belong to that terminology. In other words, the domain of the function yielded by $\mathsf{Description}$ should be a subset of the associated set of terms:

Axiom 6 $\mathtt{dom}(\mathsf{Description}(T)) \subseteq \mathsf{Terms}(T)$

Note that the reverse does not hold. In other words, $\mathsf{Description}(t)$ does *not* have to be a total function from $\mathsf{Terms}(T)$.
In Table 4.2, the extension of our conceptual framework is summarised. This concludes the presentation of our basic conceptual framework, put together in order to enable description and analysis of the environments of ICT-supported conceptualisation processes.

4.2 Languages and Language Types in Medium System Communities

Languages are central to our account, and therefore we are interested in types of (domain) language we may distinguish, in order to approach meta-communication in a particular situation (Discourse Environment) *in accordance with the demands set by the languages to be deployed.* The typology presented is our primary means of capturing the kinds of 'language requirement' that should be met by meta-communication and conceptualisation processes.
Though we will discuss various common typifications of language, we are not primarily interested in any particular typology of languages and sub-languages. Such

Concept	Explanation	Formal Definition
$\mathcal{AG}$	all agents	–
$\mathsf{Agents} : \mathcal{SA} \to \wp(\mathcal{AG})$	speech acts involve agents	–
$\mathsf{Users}(S)$	all users of a medium system	$\bigcup_{a \in A_{\mathsf{Language}}(S)} \mathsf{Agents}(a)$
$\mathsf{Describers} : \mathcal{MS} \to \wp(\mathcal{AG})$	mediums systems have describers	–
$\mathsf{MSLC}(S)$	medium system language community	$\mathsf{Users}(S) \cup \mathsf{Describers}(S)$
$\mathcal{DE}$	all discourse environments	–
$\mathsf{Agents} : \mathcal{DE} \to \wp(\mathcal{AG})$	discourse environments have agents	–
$\mathsf{Scope} : \mathcal{DE} \to \wp(\mathcal{MS})$	scope of a discourse environment	–
$\mathsf{Vocabulary} : \mathcal{AG} \to \wp(\mathcal{CO})$	agents may have vocabularies	–
$\mathsf{UoD} : \mathcal{DE} \to \wp(\mathcal{CO})$	universe of discourse	$\bigcup_{g \in \mathcal{AG}} \mathsf{Vocabulary}(g, d)$
$\mathcal{TM}$	all terminologies	–
$\mathsf{Terminology} : \mathcal{DE} \to \wp(\mathcal{TM})$	discourse environments may have terminologies	–
$\mathsf{Terms} : \mathcal{TM} \to \wp(\mathcal{FO})$	terms are concept forms from terminologies	–
$\mathsf{Description} : \mathcal{TM} \to (\mathcal{FO} \rightarrowtail \mathcal{FO})$	term descriptions are (meta-)concept forms	–

Table 4.2: Overview of the extension of our conceptual framework

a classifications are highly domain-dependent; they will differ greatly in view of a particular meta-language as used in a particular domain. Comprehensive listing of language types in terms of use domains is therefore both impossible and futile. We are more interested in a deeper understanding of types of *language use*, and the generally applicable *properties and factors* that may be used to evaluate conceptualisation efforts against the result they should deliver, i.e. a language *fit for a certain use*. The concept of MSLC helps us to designate language types that are 'outside' some MSLC, thus providing a means of focus.

4.2.1 Community-Oriented Typification of Language Domains

In conversation, it is not uncommon for people to refer to languages, vocabularies, terminologies, jargon, etc. Essentially, various types of distinction boil down to identification of the *communities* the languages typically belong to.

Most commonly, languages are distinguished at the hand of their nation or area of origin (for example, English). However, national languages are not of much interest to us here. More specifically terminology-oriented sub-languages are often referred to in terms of the (distinctive) group of professionals or specialists using them. For example, 'legal language' is used by people in the legal professions, 'medical language' by medical personnel, and 'IT language' by computer experts. Obviously, the general domains mentioned can be subdivided in sub-domains and sub-domains of sub-domains, and the demarkation, purpose, and contents of terminological domains may be (and often are) subject to fierce discussion.

Apart from a categorisation in terms of professional fields, other common bases for categorisations include organisational units ('IBM', 'South-East Branch', 'Project X'), operations or tasks that are to be performed (instruction, negotiation, administration), specific medium systems that are to be used (e-mail, database, long-distance radio), or working environments (office, operation room, ship's bridge). Sometimes there may be an overlap between types of categorisation (accountancy, administration, IT, sales: all professional fields but also, commonly, specialised departments in organisations). Note that some types of language categorisation are more closely related to particular DEs than others.

In any case, directly or indirectly the above mentioned distinctions can be related to groups of people that make use of them[1]. The purpose served by such categorisation is to actively and successfully invent labels for terminologies, and use these to single out some subset of terms they consider relevant, for whatever reason. Notably, the underlying type of categorising may help us understand or express the perspective of the person or group making a certain distinction. Which particular terminological domains are important in a particular DE is not interesting *per se*, but such information may serve as a lead to identify and determine further factors as discussed in this section.

[1] They are therefore closely related to communal linguistic conventions.

4.2.2 Computational versus Communicational Language Use

An alternative way of categorising language is by focusing on the use a language is put to, or is intended to be put to: its *functionality*. For example, we distinguish 'programming languages': languages primarily designed for use in programming machines (though as we shall see, programming languages may in fact cover several functions in parallel).

Within our perspective a vital distinction between types of language function is the one between *the construction of symbolic representations as part of a computational machine* and *the use of symbols for human-to-human communication*. It has already been discussed in Section 2.1.1. Perhaps the clearest example of a 'language for technical use' is indeed a programming language. However, even for this typically 'technical' language, two linguistic functions can be observed to be active: the use that a programming language is primarily designed for, i.e. constructing an executable program that can be run on a computer, and the secondary use of (aspects of) that programming language: as a language for human-to-human communication.

In Chapter 3, we have elaborately discussed our view on the nature of 'human language semantics'. In contrast to this view, which centres round cognition-based interpretation by humans, the semantics of programming languages can be seen as *operational semantics*: "the changes that occur in the machine's state when it executes a given statement define the meaning of that statement" (Sebesta, 1999, p.132). However, in our perspective we are primarily concerned with human language semantics, and hence focus on *what a program means to some human.*

Take an early, 'low level' type of computer language like Assembly. It consists of symbols that have a direct relation with the inner operations of the computer's CPU and memory. Apart from the use of characters borrowed from natural language script, hardly any concept labels are shared with an actual natural language. Consequently, a layman will 'understand nothing of assembly': the concepts of assembly are fully tuned to the machine, and only people understanding the machine understand the language.

However, if we consider 'higher' programming languages, more symbols (words) appear that have been borrowed from natural language. They roughly occur on two levels:

- they may be built into the language itself as labels for computational instructions (commands)
- they may be used as labels (for certain variables, files, routines, etc.) that are part of the program, but not of the programming language as such.

In either case, labels are usually not chosen arbitrarily. By somehow relating them to relevant existing words in Natural Language (NL), the labels 'make sense': their meaning in NL corresponds roughly with a certain item or functionality in the program that the expression is related to (Sebesta, 1999, pp.8-20). In other words, the NL meaning is used as a descriptive heuristic for remembering and understanding the code's functional load, and thus has an explanatory and documentational function.

In addition, remarks in open, natural language (heavy on technical jargon) are commonly added to the programming code. For a particularly advanced approach to the the use of natural language descriptions alongside programming code, see (Knuth, 1984; Sewell, 1989).

It is important to see that resemblances between a natural language and a programming language help programmers cope with the code, but that that code still is, in essence, part of a machine, and consequently needs to be 'precise' and 'formal' in nature. In the mean time, it cannot be denied that when programmers write or read code that resembles NL in some way, they *also*, at the same time, participate in human-to-human communication, either with themselves or with colleague programmers. Note that this double use of language extends to any situation in which someone (even a user) operates a machine by feeding it symbolic representations.

Along similar lines, it regularly occurs that symbols (words) used in the interface of an information system, i.e. belonging to the UoD on which its data structure is based, appear literally in the code. Consequently, such a symbolic representation simultaneously plays multiple roles in very different language games. Assuming that different language functionality may require different language, it is therefore important that the differences between uses of language symbols are clearly distinguished to prevent confusion between their various meanings and uses (and the *stakeholders* and *concerns* that underly them).

Note that all this implies that the notion of "natural language" or "programming language" is better replaced for our purposes by "natural use of language" and "computational use of language". Overlap between various functions of some language form can most certainly have its use, but it helps if such overlap is well understood and treated with some caution. This may prevent situations in which concepts are imposed on agents (human or automated) for use in situations they were not intended for, and for which they may be inappropriate or impractical. Also, it may encourage agents (typically human) that are responsible for the selection, creation or adaptation of concepts and languages to actively consider the particular use or uses those concept may be put to.

It is of course quite common for technical terminology (borrowed from the domain of some programming language, software library, or even the domain of a particular program, system, or application) to enter into the language and conversation of ICT professionals. Programmers, system operators and the like, working persistently on a particular system or piece of software, naturally develop a highly peculiar vocabulary concerning that system or program, *on top of* the more widely spread technical jargon tied up with a programming language.

Confusingly, terms from the UoD of the user may thus enter technical discussion in a technical DE and technical language may become part of the user UoD: user language then finds its way into the computational machinery (in the form of variables, data structure, etc.) and technical language enters the user UoD because users need it to work with the system. If such terminological cross-over occurs, and an actor from the technical DE also participates in another DE alongside, for example, users of the system, chances are that within the latter DE, people *think* they use the same concept while in fact they are not. For example, in an Object-Oriented (OO) programming environment, an automated business process like 'payment' or 'order' is an 'object' just like any other software component. This may lead to a situation in which people talk about 'business objects', while others may still prefer to talk about 'business processes': the technical domain influences the user domain, possibly creating conceptual confusion.

By now it should be clear that our focus excludes computational use of language as such, but that confusion in human-to-human communication as the result of the intermingling of differently oriented agents in some DE is quite relevant. Though in an idealised situation, the engineers responsible for implementation of a system should perhaps not occur at all in an MSLC, such a situation is not very realistic in current ICT practice. At the very least, there will be people who 'translate' communicational terms from the UoD into computational labels. This is, in fact, an almost inherent part of system design.

Finally, there are many domains of language that are strictly meant for human-to-human communication and yet have a very technical background and goal: those concepts that help users operate machines. This type of language use differs from computationally used language in that symbols involved are no part of the mechanical workings of the machine as such: the language is strictly meant to confer to the users how the machine is to be operated. Perhaps the simplest example is the "ON/OFF" label accompanying a power switch: the switch works fine without the label, but the user is helped by some guidance as to the consequences of turning the switch, which is provided through the use of language. A more complicated extension of such language can be found in on-line user manuals, and arguably in any manual. Users (both productive and receptive) of this type of language (let us call it 'technical instruction language') should perhaps not in essence be seen as part of an MSLC, but in practice it is at least quite relevant to distinguish technical instruction language as a basic type (in the sense of 'communal domain typification' as discussed in Section 4.2.1). Most users of 'technical instruction language' will also use the operational communicational language that occurs in the UoD (see next section), and in fact technical instruction language and work process related instruction are often blended together in work process documentation (see Section 7.3.2 for an example).

4.2.3 Operational versus Auxiliary Language Use

Another use-based perspective that might be used to distinguish between language types concerns the type of organisational communication process the language is used in. In line with (March and Simon, 1958), van Reijswoud and Dietz (1999, pp.17-8) distinguish five different groups of organisational communication. They use this distinction as one of the pillars of their theory of business communication, which underlies their DEMO method for modelling business transactions and processes.

1. **Informal communication** describes all the communication that is not directly related to the business processes in an organisation.

2. **Precondition creating communication** is the communication that sets the organisational conditions for people to work together.

3. **Information providing communication** comprises all the communication

that is used to convey information from one place in the organisation to another place in the organisation.

4. **Action initiating communication** is used to start an operational business process in an organisation.

5. **Action reporting communication** is used to state the result of the performance of a particular action relating to the production of a good or service.

Interestingly, if we follow the above categorisation, linguistic meta-communication as discussed in this thesis belongs to precondition creating communication. This would put meta-communication firmly in the realm of 'secondary' or 'auxiliary' communication: important though it may be, it does not belong to what we might call 'primary' or 'operational' communication (types 4 and 5).
However, van Reijswoud and Dietz focus solely on business communication. This does not match our own perspective: we prefer to let the primacy of some type of communication (and hence also the associated language function) depend on the *particular* DE we look at. For example, van Reijswoud and Dietz consider *management language* a generally tenable example of auxiliary language use (since it does not directly concern business process communication of type 4 or 5). From our perspective, *what is auxiliary or operational language use depends the* DE *that is considered*. Thus, if the main operational goal within a DE is to manage, 'management language' is operational, not auxiliary, and if writing a data dictionary is the main goal in a DE, the writing as such concerns operational communication if seen from that DE yet meta-communication if seen from a DE in which the dictionary is eventually used. We therefore use a simpler distinction more fitting for our linguistic point of view:

Operational communication is directly related to performing work (work processes) in a DE

Auxiliary communication comprises all the communication that is not operational in a DE

The distinction between operational and auxiliary language use is mostly important because it helps estimate the urgency of providing resources and infrastructure to *support* a particular kind of language use in a particular DE. Note that auxiliary language use can be supported just as well as operational language use, and that this may even include the support of language used to talk about language (linguistic meta-communication). In what way resources for conceptualisation are allocated is a matter of management; it all depends on the kind of DE communication takes place in, and how important particular support is deemed to be.
In addition, allocation of resources will depend on how important the task performed in a DE is in view of the goals of the organisation at large (alternatively put, the goals that drive the people in the MSLC; also see next section). For example, if it is unlikely that anyone will ever use a certain conceptual manual, it makes little sense to spend much time and money providing it. Thus, the DE with as operational task the writing of manuals will probably be poorly funded, and this relates to resources for explicit conceptualisation and meta-communication in general.

4.2.4 Language for Intra-DE Communication versus Inter-DE Communication

A fourth basic distinction we make is between language used for communication *within one* DE and for communication *between* DEs (in which case a specialised DE may be installed). This distinction is also mostly relevant from the point of view of the individual DE: 'looking out from within a DE', as it were. It is particularly important in view of the effort that might be needed to either carry over some language to another DE or adjust a language so it may better fit some DE (or possibly various DEs). For example, 'management language' may be part of a DE and be accepted as 'local' (see 4.2.5). In that case, management terminology will not pose problems. However, if the management domain is effectively disconnected from some DE, management terminology becomes 'foreign', which has obvious consequences for meta-communication and conceptualisation.
Note that it is quite common for DEs to be called into existence for the specific purpose of communication between two domains (also see Section 4.1.1, last paragraph); think for example of inter-department meetings, but also of dedicated meta-communication sessions (for example, unifying, comparing or translating terminologies). In fact, it can well be argued that every time communication takes place between two agents based in separate DEs, a special 'intermediate' DE is created. However, it seems best to restrict the distinction of DEs to 'relevant' situations that at least exist beyond the span of a single conversation; i.e. are either permanent or occur repeatedly and predictably. Nevertheless, discourse environments can in principle be described for any instance of discourse taking place, and meta-communication and conceptualisation may in principle occur in any DE. Whether in some DE, meta-communication and conceptualisation need to be explicitly *supported* is another question.

4.2.5 Further Distinctions in Language Use

There are a number of additional factors that may be usefully introduced as part of a typology of language and language use. However, we believe they are best presented as *properties* (perhaps typical of some terminology or DE) and not necessary as "types of language and language use". A number of these properties have already been touched upon, and some of them are strongly related, but in an attempt to be comprehensive within the boundaries of relevance, we list and briefly describe them all below.

Global versus Local Language use

Positioned between two extremes of a continuum, language may be intended for use by a few (even one), isolated agents (local), or it may be intended for widespread, even truly 'global' (i.e. world-wide) use. Though 'globality' will usually be related to 'geographical space', it may also relate to 'cyberspace': perhaps two agents on different continents can be quite 'close' if connected by a medium system. Alternatively, a notion of 'cultural space' might be used here (the larger their cultural differences, the larger the distance between them in 'cultural space'). Therefore, globality is more linked to the scale of linguistic differences between the vocabularies of stakeholders in a DE than to the physical distance between agents. The degree of 'globality' intended has consequences for various other factors and aspects. To mention a few:

- The more people are included, the more the use of a specialised vocabulary becomes a problem.
- Also, it will be increasingly difficult to accommodate for the needs of everyone involved.
- Any kind of concept control or management becomes more difficult if more different people are involved.
- If globality becomes high enough, areas may be involved that require different 'background languages' to be taken into account; for example, different national languages may come into play.

Note that the global/local distinction is related to the agents who use some language, and that therefore this distinction is strongly *community-based*. The difference between this distinction and that of more regular community-based differences as discussed in Section 4.2.1 it that here we do not look at *what* people in the community have in common (e.g. living in France or being a lawyer), but rather at *how much* they have in common (for example, a community may include both professionals and non-professionals).

General versus Specialised Language Use

This property is related to the 'global/local' property, but differs from it in that it emphasises not the linguistic distance between stakeholders in a DE, but their underlying degree of *expertise*. In other words: general language use assumes little specific expertise; specialised language use assumes considerable expertise, in some operational task which taks part in some DE. A high degree of professional expertise often entails the willingness and capacity to *invest* in acquiring, using, and supporting a specialised language. This distinction is community-based.

Persistency of Utterances Created by Language

If language use applies to a single, volatile interpretation (say, a single speech event), this is quite different from language use for persistent representation (text: e.g. documentation, specification, a database). In the first case (and depending on some other factors), it is usually better possible to estimate what the audience will be. However, it is sometimes hard to predict what life a text may lead and who will eventually get to read it (and in which context). Also, as representations of utterances last longer or are distributed more widely, the flexibility issue becomes an increasing problem. Knowing how long the product of some language use will last can, therefore, be of great importance. Note that this distinction is not community-oriented, but more typically *function-oriented*.

Contextualised or Decontextualised use of Language

This factor is knowledge-oriented (and thus related to the general-specialised distinction). As has been elaborately explained in Chapter 3, context is important for interpretation. Depending on the availability and importance of *contextual knowledge* in the DE, a higher or lower degree of pragmatic information should be provided alongside (or part of) language description; this has substantial consequences for the requirements on meta-communication.

Productive versus Receptive Use of Language

It is important to consider whether or not a particular language is used productively (e.g. in speech or writing) or receptively (listening or reading). In relation to this, it is important to know which type of stakeholder (possible, which *role*) will involve productive/receptive use. Note that therefore, this is a typically function-oriented distinction, which may be *linked* to a user group (roles, stakeholders, etc.).
For receptive use, little or no active control or management has to be exercised, since the language user (hearer/reader) is simply confronted with a particular language and will need to deal with it. For example, if for some agent the fields of a database are 'read only', the 'use' of the concepts involved is hardly optional for that agent. For productive use, language users (speakers/writers) may actively select (or refuse to select) particular concepts; such selection is much harder to control and manage. For example, a particular (closed language) database field may be considered 'useless' to some agent, and consequently she may simply never fill anything in, or leave it on its default setting. Enforcing correct active use of a data field arguably is harder than just passively confronting people with some concept (merely forcing them to draw on general linguistic conventions known to them).

Social Properties of Language

It is often important to know whether a certain word or terminology is socially marked (is subject to some communal linguistic convention; see Section 3.2.9). For example, a term may be either 'fashionable' or perhaps 'unfashionable'; it may be 'standard' or 'idiosyncratic'. Depending on the influence the 'social factor' (also see Section 4.3.2) has within a DE, the social 'value' of a word may either boost or frustrate attempts to introduce or support it. The existence in a DE of a strong tradition, for example, will increase the importance of the social properties of a concept or language considerably. This distinction is community-oriented: it concerns the question how strongly a language is communally marked in the knowledge of community members.

Practical Properties of Language Use

A property of language use already introduced in Section 2.2.3 (and one that will be further discussed in Section 4.3.2), concerns the 'practical factor'. This distinction is directly related to the communicative functionality of language. Especially in an operational context, people may highly value the use of a certain word to get a communicative job done, even if this violates communal linguistic conventions or

even official decrees. Efficiency, certainty, and technology-based restrictions are the typical sub-factors here. They may well overrule the social factor.
However, the contrary also occurs: some language use may be highly unpractical in operational use, to the benefit of some other aspect or goal. Whether or not the use of such impractical language is the 'best' solution depends, as always, on the DE in which the particular language use takes place. In addition, the question of what is 'practical' rapidly becomes more complex if in some instance of language use, several communicative functions are combined (see section 4.2.2, and also Section 7.3.1).

Official Language Use

We define 'official' language use as the use of the language that some *authority* prefers to be used (for some particular DE and/or task). This distinction is community-oriented. Arguably the property is strongly related to the social properties of a language, but we prefer to distinguish the 'official' status of a word from it being 'strongly socially marked': it has to do with *authority*. For example, it may be the case that a word is highly unpopular ('negatively marked'), yet is strongly advocated by the Management[2]. In most cases, if a concept is 'official' his means it is part of a terminology. When language becomes official and its use is considered important enough by the authority in question, there may be reason for that authority to *wield power* in order to *enforce* a particular language to be used.

Flexibility of Language

This property refers to the ease with which a language can be changed. It is not community-oriented, nor functionality-oriented. It represents a category in itself. An important case in which flexibility is low is when a language is a standardised, or at least is standardised. Standards are both common and important to language use, in particular when we consider terminologies. Standards restrict but may also facilitate communication (practical factor). Whether or not standards are helpful depends on the situation in the DE under consideration. For example, if in a DE many different stakeholders participate, freely combining many different vocabularies, standardisation may be a huge problem. Also, if the language in a DE is rapidly changing, standards may be difficult to keep up (or up to date).
When standards are concerned, in fact several other properties of language and language use are touched upon: 'officialness', global/local, and general/specialised. A language may be flexible or not, but for it to be standardised it must be *made* inflexible, which might for example be achieved by making it official; etc.

Language Built into a Closed Medium System

Finally, there is the property that lies at the heart of our discussion of Frozen Language in an ICT context: whether or not the language used is actively imposed, and thereby restricted, by a medium machine. Obviously, this relates to the type of medium system that is used to communicate, but also *whether a particular language*

[2]It is in fact not uncommon for a term to be highly unpopular *because* it is advocated by the Management.

is associated with it (which need not be the case). Under our rather strict definition, for language use to be 'closed', it needs to involve a *linguistically restricting medium system*, and therefore it is not enough for explicit agreements on language to exist. We have opted to classify this property as 'function-related'. However, it is also closely related to the 'flexibility' property, to 'officiality' of language, and to diverse aspects of the practical factor. It is indeed a highly complex factor. Not surprisingly, decisions about whether to freeze a concept or not should depend on the weighing of all previously mentioned properties in combination with further demands and resources that occur in a particular DE.

4.2.6 A Typology of Language Use

To sum up, we can typify language and language use in terms of a number of main types and properties. Without exception, this typification can be applied to complete terminologies as well as explicit descriptions of individual concepts (or *terms*). In most cases, the typification can also be used for implicit, 'mental' concepts or vocabularies, with the exception of computational language use, official language, and closed language use in medium systems (these properties require language to be explicitly described or specified).
In Figure 4.1, a boiled-down presentation is given of the language typology we suggest, including a rough classification of the properties distinguished and a short, heuristic question attached to each property.
This concludes our discussion of typifications of languages and language use as relevant in Medium System Language Communities and Discourse Environments. The types discussed should enable us to distinguish between the different purposes that may be served by meta-communication and freezing language, i.e. the sort of product that should come out of some particular conceptualisation process as required by its communicational environment.

4.3 Linguistic Meta-Communication in ICT-Supported Organisations

We introduced and illustrated our perspective on linguistic meta-communication in ICT in section 2.2, in context of our initial analysis of the Frozen Language Issue, and suggested some directions for solutions involving meta-communication in Section 2.3. We also provided a theoretical analysis of linguistic meta-communication in Section 3.4. We will now elaborate on the topic.
We define meta-communication as *the process of communicating about communication.* In meta-communication, any aspect of communication can be discussed in principle, for example physical means of communication, matters of protocol, matters of authority, politeness, etc., but also meta-communication about the *language* used to communicate with. Note that for meta-communication to take place, the parties involved must share some basic means (medium, language) that enable it. Also note that meta-communication can in itself be subject to meta-communication[3].

[3]To call this meta-meta-communication seems, however, quite pointless in our current discus-

Community-related	
Common classification	*Who use it?*
Topic	*Who talks about it?*
Professional group	*Who needs it for this line of work?*
Organisational unit	*Who need it in this organisation?*
Involved in activity/task	*Who need it for this medium system?*
Using medium system	*Who need it for this task?*
…	
Global or local use	*How many different "whos" use it?*
Social properties	*Does it belong to "us"?*
For intra or inter-DE communication	*Need "they" use it?*
Official status of language	*Do the authorities want us to use it?*
Function-related	
Computational or communicational use	*For talking or for building?*
Persistency of utterance in use	*Staying or fleeting utterances?*
Productive or receptive use	*Active or passive?*
Practical properties	*Does it work?*
Priority-related	
Operational or auxiliary use	*How important is it?*
Knowledge-related	
General versus specialised use	*How knowledge-intensive is it?*
Contextualised or decontextualised use	*Pragmatic knowledge involved?*
Change-related	
Flexibility of language	*How fast does it (need to) change?*

Figure 4.1: A Typology of Languages and Language Uses

It should be made clear first of all that communication that serves to directly fill some general information gap concerning an utterance is *not* linguistic meta-communication. Linguistic meta-communication strictly concerns the language system as such. The distinction may seem vague, but it is essential. For example, if a teacher tells his students to "hand in an essay", a question for extra 'general information' might be, "when do you want it?". This is not linguistic meta-communication, since it does not focus on language but on operational communication. However, if the response were "what do you mean with 'essay'?", then this is the initiating of linguistic meta-communication. If in the remainder of this thesis we use the term *meta-communication*, we strictly mean *linguistic meta-communication* unless we explicitly indicate otherwise.

We approach linguistic meta-communication primarily as a process; in an organisation, it will in fact be a cluster of processes. Most if not all occurrences of meta-communication will be embedded in organisational processes that do not have meta-communication as their main goal, and hardly anyone involved will be aware that they are partaking in linguistic meta-communication. This is part of what makes our perspective interesting: we try to bring to the surface an essential functional aspect of languaging and communication in ICT-supported communities that has always been an integrated part of information system design and use but that has not been explicitly scrutinised. We hope to thereby deepen our understanding of (trouble with) communication through information systems, directly reflecting aspects of *information system functionality*. More concretely, linguistic meta-communication is the main concept that enables us to link language and language use with information system design, maintenance, and use.

4.3.1 Meta-Communication and Texts

In addition to languaging, people have developed the trick of making language utterances persistent by means of *texts*. Much meta-communication takes place through non-persistent media, and in its explicit form is not around for long. However, matters are different when we consider dictionaries, written definitions, grammars, etc.: they are texts. Once meta-communication texts are around, we are close to dealing with frozen language. Textual language descriptions represent a potential link between implicit language conventions and closed medium systems (which require an explicit specification of the language they may convey; see Section 2.1.2). In addition, when a language description is explicitly linked up to the language forms used in a medium system (system documentation; see Wettengel and van de Weyer, 2001), it becomes part of it and arguably has become part of frozen language.

Operational information systems are also a form of text. They represent symbolic utterances, which in many cases are persistent. For example, the data stored in a database is a text. This type of text should not be confused with the explicit specification of the language *to be built into* an information system (i.e. the medium system language).

Importantly, there is a difference between problems involving frozen language as primarily discussed in this thesis, and issues that are the subject of the field of

sion; see Section 3.4.1.

hermeneutics: the study of (text) interpretation. Hermeneutics is about the interpretation of written utterances, whereas in this thesis we are mainly concerned with the language conventions used to *put together such utterances*. We very explicitly deal with freezing language (an activity), with frozen language (which is the result of linguistic meta-communication), and even with the *use* of frozen language, but we are *not* primarily interested in the recorded *products* of the use of language, whether frozen or not. Note that the vast majority of texts are in fact not authored using a closed language.

4.3.2 Drives Behind Term Use and Meta-Communication

The development, introduction, and use of concepts or terms takes an effort, and we assume that people are only willing to put in such an effort if there is some drive behind it. That drive then is what encourages the use of certain terms, but possibly also meta-communication about them. It underlies the willingness to discuss terms, find out about their meaning, and perhaps also to define them or 'freeze' them.

From the perspective of an individual agent, the drive to acquire and use a certain word roughly depends on two (related) factors: the attitudes and culture of the group she belongs to (social factor), and the tasks or situations she has to communicate about (practical factor). The social factor relates mostly to what we have called 'communal linguistic conventions' (Section 3.2.9). The practical factor was presented in Sections 2.2.3 in order to explain why people might want to freeze language. In this section, our focus is wider: we are concerned with drives behind term use and meta-communication in general. The social factor may play some role in freezing language, but does not appear to be the primary driving force behind it.

The practical factor is most typically related with 'general linguistic conventions' (Section 3.2.9) that enable a language user to recognise or wield the meaning of some term in some context even if this is not considered 'socially appropriate' (i.e. falls outside the regular and familiar language use of the user, yet "works"). However, if practical-driven use goes against the communal conventions that the user expects to hold in the system DE, this increases the risk of misinterpretation as the intention of the speaker is less 'predictable'. This may in fact reduce practicality, *certainty* to be precise. In other words, letting practicality prevail over communal linguistic conventions can be counter-productive; a balance needs to be struck.

The fact that a DE may have its own vocabulary and terminology is not only a result of social coherence between individuals, it is also an important expression of that same social coherence and may be a binding factor within the organisation. Common languages are part of what defines a group; particular language use can signal membership of a group, but also social 'distance' from it (Trudgill, 1983; Crystal, 1987, pp.38-). For example, a certain vocabulary (but also pronunciation and even grammar) can mark someone as belonging to a certain class or age, coming from a certain region or country, etc. On the work floor, similar differences can be observed (however, they mostly concern vocabulary/terminology). The differences in language use between different groups in a working environment will usually be related to the particular job the members of that group fulfil. For example, (Schoop, 1999) describes crucial differences between the language of doctors and nurses in hospital.

Choices to adopt a particular word (and possibly to engage in meta-communication about it if necessary) are often the result of a combination of the two factors; one 'language act' (the use of some concept) thus has a social and a practical aspect. In most cases, both the social and the practical factor most directly influence the choice of *individuals* to adopt certain words, a choice which may or may not be approved of or joined in by other members of the group. Only in some cases will there be an *explicit* agreement between the members of the group about the use of some word, making it a *term*.

So the social and the practical drive are often intertwined, and yet we believe that distinguishing them helps us to account for the preference for the use of particular words by particular people in particular situations, as possibly reflected in shared language use within some group or DE but also potentially leading to an explicit agreement about this. The two factors need to be distinguished primarily because in relation to language use, they may lead their own lives; they may even become opposite forces. Though either of the factors may, in some situation, outweigh the other, we suspect that the practical factor tends to outweigh the social one in professional, cooperative discourse environments. For example, use of a word that in a situation in the past was born out of immediate, operational need (practical factor), but has now been firmly established as part of the language of a group (social factor), may outlive its practicality yet remain deeply ingrained in the social identity of the DE. Generally, however, the practical need will eventually overrule the old communal linguistic convention, and the more practical word will take its place.

Where the use of socially agreed, 'appropriate' concepts is most prominently backed by the claim to justice, the practically driven use of concepts may be more strategic in nature: it may confront people with a concept that they can cope with (drawing on general linguistic conventions, essentially backed up by the claim to truth) but that is not appropriate (violating social linguistic conventions). Such use may even involve the 'linguistic claim to power' (Section 3.4). However, if all agents involved see the practical need for the use of some concept, what may initially have been a case of 'conceptual power play' can rapidly become a communicative situation based upon mutual understanding: the agreement that some concept is the most practical one to use. Within a DE, this will usually lead to such a term quickly *becoming* appropriate.

The drives behind word use do not necessarily lead to meta-communication, but for meta-communication to occur, the parties involved must at least have an interest of some kind in using (or not using) some term. Since such an interest will involve the social or practical factor (or both), it seems reasonable to state that these factors also underly the need to perform meta-communication.

It is also possible that a stronger form of strategic action is introduced in meta-communication. Some authority-related agent or group may impose a term upon the DE at large, and thereby upon all agents communicating in it. (Note that to impose implicit *vocabulary* is very hard; it will typically first have to be identified and spelled out, hence made explicit (*terminology*)). Whether or not such an imposed term will be successful then depends not only on the social and practical factors at play in the DE, but also upon the non-communicational (for example, disciplinary) consequences suffered by members of the group if they do not adhere to the imposed

terms.
Note that there is a difference between someone imposing a terminology, and someone being a leading figure that the group follows *voluntarily*, in which case communicative action still is a driving factor and the social factor will be activated to back up the terminology. Also note that if a terminology is imposed on a DE, for example by 'The Boss', this may be socially well accepted but *practically* raise objections.
To conclude, whether or not some term will become an accepted part of the vocabulary of individual agents in a particular DE depends on the dynamics between two main factors:

Practical factor reflecting practical needs of individuals to use a certain term

Social factor reflecting social preference among a community of individual agents

The two main factors each underly (or bundle) three aspects:

- Practical factor
  - Efficiency
  - Certainty
  - Technology
- Social factor
  - Individual language preference
  - Group language preference
  - Official language preference

We assume that these factors and aspects also constitute the motivation behind linguistic meta-communication.

4.3.3 A Typology of ICT-Related Meta-Communication

In this section, we discuss four main types of meta-communication that we distinguish for the analysis of meta-communication in ICT-intensive domains. The typology is based on related but distinctive aspects of meta-communication in an ICT context. As will be explained, distinguishing between these aspects is required as a basis for a general understanding of how meta-communication and conceptualisation relate to essentially different situations and problems concerning language in ICT-supported environments.

Meta-Communication Before or After Operational Communication

This aspect is especially important in view of the development cycle of information systems, and is quite directly related to engineering issues.
From the perspective of a medium system and its development, linguistic meta-communication may occur for two main reasons. First, to discuss means of communication because something has gone wrong, i.e. to solve a *language breakdown*.

This we call *ex post meta-communication.* Second, meta-communication may take place to discuss means of communication *in anticipation* of some communication situation: ex ante meta-communication[4].

We take the point of view that discussion of communication-functional aspects of (information and) communication systems, most in particular discussion concerning language-related matters, is a form of meta-communication. In this light, ex ante meta-communication most often occurs in the actual design phase of a medium system, but it is also possible that ex ante adaptation of the medium occurs after the initial design stage (perhaps iteratively, in regular revisions of the medium machine). In any case, to adapt a system, specification of the mutations will somehow have to be included: ex ante in anticipation of future needs, ex post in direct response to encountered conceptual problems in instantiated processes.

From a naive point of view, it may seem that ex ante meta-communication is preferable to ex post meta-communication, since it does not imply the occurrence of a breakdown. However, in complex, dynamic environments it seems impossible to accurately predict all occurring needs and changes in language and language use. Hence, in such environments it is not realistic to assume breakdown can be avoided in all cases (despite ex ante efforts), and therefore a 'backup mechanism' should available: ex post meta-communication.

Changing the Medium System or Not

This aspect is also linked with a system development perspective. It distinguishes between those cases in which the actual medium system, and possibly also related artefacts, are changed, and cases in which meta-communication is purely informative.

If meta-communication is strictly aimed at repairing or clarifying communication, some alternative (open) communication channel will usually be engaged to sort things out between the agents involved. This we call *non-intervening meta-communication*: it does not attempt to change the medium system (or related artefacts). If, however, an attempt is made to adapt (improve) the medium system, a process of *intervening meta-communication* is initiated. This will often involve a relatively elaborate series of activities, in which several agents may play a role. Note that non-intervening and intervening meta-communication may occur independent of each other, but usually the second will occur after the first.

Relation to Operational Communication

Another relevant distinction is that between *non-mediated* and *mediated* linguistic meta-communication. It concerns *who is involved* in a particular act of meta-communication. The *operational communication process* (see Section 4.2.3) is the communication process in the 'success layer' of the TPM that may or may not be supported by some meta-communication process. The parties involved in it are the *operational communication parties.* However, it is quite usual for some third party (one that is not involved in the operational communication process), to be somehow involved in meta-communication, even if this is merely through some *text*

[4]The terms ex ante and ex post meta-communication were suggested to us by Goran Goldkuhl.

(for example, a general dictionary, the lexicographers then being a rather distant and abstract meta-communicating party, with no access to contextual knowledge) or as an aid (facilitator) in the meta-communication process. This we call *mediated meta-communication*. Obviously, meta-communication between the parties also involved in the operational communication process is then called *non-mediated meta-communication*.

Thus, the criterium that determines whether we deal with non-mediated meta-communication is whether it strictly involves agents that also play a role in the operational communication event that the meta-communication supports. This distinction is very helpful in determining the knowledge and goals of agents involved in meta-communication. It assists, for example, in the estimation of the 'pragmatic-linguistic distance' that a particular party in meta-communication might have with respect to the operational communication situation.

The possibility of mediated meta-communication brings in mind an observation by Putnam[5] concerning what he calls the "linguistic division of labour": "the principle that in any linguistic community some people have more expertise in applying certain terms than others do. For example, in our community some people know more about trees than others and so can tell which trees are, for example, beeches and which are elms." (quotation from Searle, 1983, p.201). The idea thus is that in some cases terms are known and used that require assistance once the meaning of those terms needs to be closely examined. This entails that sometimes, people are willing to go with *expert advice* on the language they use rather than *discussing of some term meaning amongst themselves*. A hybrid form is of course a situation where discussion among the users of a term does take place, but either the discussion as such is mediated or an expert advisor takes part in the discussion. All cases of meta-communication except 'discussing some meaning amongst users' are instances of mediated meta-communication.

Any of the agents/roles within the MSLC can in principle be involved in mediated meta-communication, but so can MSLC-independent standardisation authorities, dictionary publishers, law makers, or what have you. So it is possible for some agent outside the MSLC to participate in mediated meta-communication, but it is by definition impossible for someone outside the MSLC to participate in non-mediated meta-communication. Consequently, language use of an 'outsider' can be described and analysed within our framework, but only if it concerns mediated meta-communication.

Language-Oriented or Machine-Oriented

Parties that are involved in linguistic meta-communication typically come in two flavours: *language-related* and *implementation-related*. Language related meta-communication agents play a role in determining the *medium system language* of the medium system S (see Section 2.1.2). Implementation-related meta-communication agents are strictly involved in the construction or maintenance of the closed-communication medium machine M, including the means for providing information

[5]Putnam makes this observation as part of a philosophical argumentation concerning the nature of meaning (Putnam, 1975). We do not aim to contribute to or to even touch upon that particular discussion; we merely borrow the idea of 'linguistic division of labour' and use it as an illustration.

about the concepts used (documentation used in mediated meta-communication). They could be 'data structure implementors', or possibly 'medium machine builders' or even 'documentation managers'. As stated in Section 4.2.2, they are not part of the MSLC in principle, unless they are *also* involved in some operational communication process or linguistic meta-communication process, which is in fact not uncommon.
Language-related and implementation-related agents may or may not be united in the same party or authority. It may also occur that in some situation, either of the two agent types is not explicitly represented or known: though a closed language may be present as well as a medium system including it, it may not be clear who is responsible for introducing or maintaining them. The distinction between language-related and implementation-related participants in meta-communication is therefore primarily intended to provide focus and keep apart potentially confusing roles of agents. Note that language-related meta-communication may be intervening just as well as implementation-related meta-communication.
In sum, we distinguish the following types of meta-communication:

Ex ante versus ex post meta-communication which is related to the difference in time of meta-communication occurrence relative to some instance of primary communication, i.e. either before (anticipating communication) or after (looking to repair communication in some specific instance)

Intervening versus non-intervening meta-communication related to whether or not meta-communication involves the actual changing of the medium system or related artefacts

Non-mediated versus mediated meta-communication which depends on whether or not agents participating in meta-communication are also directly involved in a operational communication process that the meta-communication is supporting

Language-related versus implementation-related meta-communication which we believe speaks for itself

4.4 Discussion of Meta-Communication in ICT-Intensive Environments

Having presented four different types of meta-communication, we will now illustrate and discuss these aspects at the hand of a number of examples. We do this in order to show that the types distinguished indeed occur regularly, and serve a concrete analytical purpose. We also hope that exemplification will simply increase the reader's understanding concerning the differences between the kinds of meta-communication that may be encountered in the ICT domain. We structure the discussion by dividing it into two parts: one on non-intervening and one on intervening meta-communication, as this has proved the clearest way of presenting. Other types of meta-communication are dealt with throughout both parts. Language-oriented and implementation-related meta-communication are not discussed explicitly, as the distinction between them seems clear enough.

4.4.1 Non-intervening Meta-communication

If an open medium system is involved, ex ante meta-communication will only marginally occur, and ex post meta-communication will usually be embedded in normal conversation between two or more communicating parties. It may or may not involve a third party for mediated meta-communication. When a closed language medium system is concerned, however, the situation is more complex. If meta-communication is non-intervening, it will usually be ex post. Such meta-communication is typical of any repair situation following 'communication breakdown'. If normal embedding of meta-communication in the operational conversation is impossible (because operational communication takes place through a restricted medium system), a dedicated meta-communication process has to be initiated. The simplest way to do this would be to open up a non-mediated meta-conversation between primary parties involved, either *vis a vis* or trough some technologically supported open communication medium (e.g. telephone, e-mail). However, some form of mediated meta-communication may also be involved (for example, documentation).

For example, take the simple case of supplier-client communication via a computerised standard order form. Suppose the client wants to send in a form, but is not satisfied with the expressiveness it offers given its standard (fixed choice) conceptual framework, which fails to provide the means to express a particular order she wants to place. Not wanting to use the standard 'further remarks' field provided by the system because she wants both immediate and interactive response (because of both *certainty* and *efficiency*), the client picks up the phone and thus opens up an extra conversation, engaging in meta-communication. The conceptual detail is discussed over the phone, and it turns out the required order can be specified after all within the fixed framework, though in slightly different terms. The client puts down the phone and sends in the form.

However, it may well be impossible to engage in any sort of non-mediated meta-communication at all, perhaps for reasons as mentioned in Section 2.2. In such cases, the only remaining immediate course of action is to resort to (non-intervening) mediated meta-communication: inquire with a knowledgeable colleague, call a helpdesk, or look things up in some piece of documentation. Unfortunately, this being the only way of clearing things up, the need for effective and elaborate support of mediated meta-communication (for example, documentation) is now much more urgent. And even if mediated communication appears successful for one party, there is no way of verifying whether everything communicationally went well on the other side of the virtual wall.

Now to make things worse, suppose there is no particular person the ordering form is sent to, and the supplier is located in another country. Non-mediated linguistic meta-communication then is seriously hampered. The client can perhaps call a helpdesk, but the person at the desk may not be fully aware of the details of the ordering form, or may not be able to convince the client that this is the case. Communication fails, the client does not risk placing a faulty order, and thus the order is not placed at all. Perhaps the client will ask for the ordering form to be updated; in this example, she probably will not.

Ex ante non-intervening meta-communication looks ahead in order to prevent conceptual trouble, but does not change the medium machine (or related language descriptions, which we view as part of the medium system). This type of meta-communication simply boils down to all sorts of *instruction*. The better users are instructed, the better they will be able to use the medium system. There is also a variation on this situation that is perhaps best compared to a 'language-oriented functional bug report' addressed to the user, that may even inspire solutions involving some sort of *by-pass* of the medium system to be adopted. If no such bypass is possible, this type of meta-communication functions merely as a conceptual danger sign. It may also play a role in decisions about whether or not to use the medium system in some situation. Consider the following example. It might be the case that the designers of the computerised standard form foresee trouble with closed concept specification, for example of some customised articles. A warning might be issued that such items are available from the supplier, but cannot be ordered through the form. One either contacts the supplier via an open channel, or refrains from ordering the items in question.

The use (not the authoring!) of *preconstructed definitions* (someone looking up a concept she is not sure about) typically belongs to ex post, mediated meta-communication. A common exception to the one-way scenario are cases in which two or more communicating parties use, for example, a dictionary for back-up of their initially non-mediated meta-communication. A more radical exception is formed by cases in which a helpdesk employee is called in, who may well be capable of two-way communication, but still performs mediated meta-communication. Note that mediated meta-communication via a helpdesk or dictionary might also occur complementary to fully open communication (i.e without a closed language medium machine being involved).

An important factor in the (receptive) use of definitions through documentation or a helpdesk is whether the communicating parties are *able* to do this, but even more importantly whether they are *willing* to. In many situations, ample information is available in principle, yet still nobody uses it. An important reason for this is the limited effectiveness and availability of information retrieval capacity: people often refuse to spend any serious length of time looking up things if this is not the primary task at hand. The helpdesk, with its intermediary human 'search agent', should be able to solve this problem, but in our experience (confirmed by the case study presented in Chapter 7) helpdesks are often not considered very helpful in meta-communication. An additional reason for people to ignore documentation is cultural rather than practical: "you just do not spend time looking up nitty-gritty things –you carry on with things". These limitations on actual use should be taken into consideration in any effort to provide effective definitions.

If for any reason some sort of communicational trouble occurs because of restrictions on meta-communication, this may trigger an *intervening* meta-communication process. Initiation of such a process is far more likely to occur in cases not allowing non-mediated meta-communication than in cases where open, non-mediated meta-communication with an identifiable counterpart is possible.

Before we turn to intervening meta-communication, let us conclude that absence of the means to perform open communication alongside closed communication increases both the need for appropriate concept-oriented information (mediated meta-communication) and the chance that intervening action is taken. If either documentation or adaptation of the system fail, communication runs an increased risk of being/becoming ineffective or faulty, which affects operations as well as the agent's opinion about the functionality of the medium system, perhaps even of the quality of the primary communication process itself. Indeed, we believe that many of the current problems with closed communication systems can be better understood if viewed in this light.

4.4.2 Intervening Meta-communication

Intervening meta-communication refers to attempts by the user (or another stakeholder) to change the medium machine (or have it changed). In many cases, especially if intervening meta-communication is triggered by ex post meta-communication, this will raise problems because systems cannot be changed 'just like that'. Such problems are part and parcel of contemporary ICT practice.

Both ex ante and ex post intervening meta-communication are of interest in an ICT context because they directly concern the *specification* of information and communication systems (i.e. medium systems), as well as their *documentation* (which is conceptually related). As we have seen, once we have said goodbye to ex post, open, non-mediated meta-communication, the importance of the quality and communicational aptness of both the medium system and its documentation becomes much higher. Changing the documentation is perhaps not the clearest case of intervening meta-communication, but we categorise it as such because it usually occurs alongside specification of the medium system, and works to actively shape and consolidate the language functionality of the concepts built into the medium system.

Let us first consider intervening meta-communication that stems from a need for change in the mediums system voiced by a user. This may be triggered by some particular incident involving failing communication (perhaps an unanticipated, acute need that emerges, leading to typical ex post meta-communication), but also by a number of experiences with the existing system which lead the user to form a more general opinion about the changes needed (leaning more towards ex ante meta-communication). This will also depend on the gravity of the incident: if it is serious enough, a single incident may trigger far reaching yet immediate ex post intervening meta-communication.

Importantly, the fact that a user wants something to change does not at all mean such change will in fact be accomplished. In many cases, there simply are no means for intervening meta-communication of any kind. In other cases, meta-communication as such may be available in principle, but no implementation follows because the meta-communication channel fails (for example, due to bureaucracy), or because the authorities reject the change proposal, or because of technical problems. Alternatively, it may be the case that the user has the authority and means to implement the desired changes herself, or perhaps can provide specifications that are immediately (even automatically) implemented. In the simplest of cases, this may solve the problem. However, even if implementation of changes in the medium system is no problem whatsoever, we are still faced with very serious problems originating in the *cooperative nature of communication*.

Put simply, if a medium system is adjusted only according to the perceived needs of one party, or even of only the directly involved parties, this may of course have highly

undesirable effects on communication with and amongst other parties using the system. Without going into this deeper here, we emphasise that the possibility for users to engage in intervening meta-communication raises questions of who should be involved in some decision concerning a medium system and how an ensuing specification process should be set up. Clearly, to give unlimited and unguided power to individual users for specifying and changing a communal medium system language may well be counterproductive. Apart from matters of implementation, the communicative needs of all players need to be taken into account. For this and other reasons, it is quite reasonable that there is some form of centralised control over the medium system specifications.
This brings us to the current mainstay of intervening meta-communication: centralised system design involving a number of stakeholders. Ex post meta-communication may trigger it or not, but in most cases, centralised, multi-party meta-communication is likely to have an ex ante character. In current ICT practice it is highly unlikely that a direct link with a single, instantiated primary communication process is maintained while action is undertaken to change the medium system. The typical type of meta-communication involved in medium system design can therefore be classified as ex ante, mediated, intervening meta-communication. The typical context in which well-constructed *definitions* are produced is als ex ante, mediated, intervening meta-communication, aiming at one-way communication through a text. For example, the writing of a manual describing the work processes supported by the medium system often includes this type of meta-communication activity.

4.4.3 Meta-communication as a Problem Solver in ICT

In sections 4.4.1 and 4.4.2, we aimed at providing clear illustrations of the various types of ICT-related meta-communication in organisations. We now change our focus to meta-communication as related to some typical activities and phenomena in ICT-supported organisations, and discuss some speaking examples of how they relate to and help solve *problems with frozen language*.

Open Communication as a Backup

Enabling open, non-mediated meta-communication between parties as a backup for (not a replacement of) closed language communication is perhaps the most basic non-avoidance approach for dealing with frozen language use. It is quite recommendable, and fortunately it is often possible. Apart from the availability of full-scale open communication channels such as telephone, e-mail, or even shouting across the office, open communication is also enabled regularly in medium systems (including traditional ones). For example, consider 'remarks' fields or other 'free choice' fields in databases and on electronic forms (as specified in data structure and interface design). Arguably, open communication in general is the best lubricant for the cog wheels of a closed medium system.
However, availability of open communication may also have its drawbacks. For example, it may encourage people to avoid use of the closed medium system language altogether. Particularly problematic in this light are cases in which the medium system is used for other than purely primary-communicational purposes, for example

for the obtainment of statistically derived management information. If concepts are to be used for this purpose alone, then no immediate, operational communicative drive exists for the user to use (some concept built into) the medium system, and therefore there is limited willingness to make an effort to properly use it at all. Automate record keeping related to particular database transactions is a good example of this.

Documentation and Expert Advice

Lacking the option of open (meta-)communication, mediated meta-communication is the straightforward alternative. It requires an investment in language-oriented information sources and perhaps training. Traditional system documentation usually mostly concerns technical aspects of the system, not conceptual aspects of its use and context. Even if the documentation includes detailed, user-oriented operational scenarios or protocols, chances are that if specific terminology is used in such documentation, it is part of the conceptual problem instead of an answer to it.

Documentation has one great advantage: it typically involves *open language* meta-communication, and if it is written well, by a knowledgable person, it will often be able to *somehow* clarify particular concepts, even if this is not the main aim of the documentation (Wright, 2001b, pp.498-501). Documentation is still one of the most powerful instruments in coping with frozen language.

Documentation that directly aims at clarification of concepts is relatively rare. Even a comprehensive list of medium system concepts is seldom available. Coherence between documentation concepts and medium system concepts, though perhaps successfully achieved by consistent and careful use of concepts, is hardly ever supported by structured, dedicated procedures or instruments. To our knowledge, the only field that has brought forth a mature collection of such instruments is that of translation. Dictionaries (involving lexicography), recorded occurrences of terms in context, cross-reference mechanisms, and advanced lexical databases and search mechanisms are available. However, all this support is strictly aimed at conceptual matching between *national* languages; surprisingly, no serious connection appears to have ever been made between translation support and with conceptual problems in ICT (with the exception of *automatic* translation, which lies outside our current focus).

Mediated meta-communication may also involve dynamic, two-way communication with some person instead of the consulting of a text. Helpdesks have so far been the main example of this, but of course any knowledgable individual may be of help. In fact, an experienced co-worker may well be of more help than an officially appointed helpdesk employee: complex contextual knowledge, acquired through hands-on experience, usually plays a vital role. The importance of the presence of experienced colleagues that may lend a helping hand should not be underestimated (Taylor, 1993; Nonaka and Takeuchi, 1995). In addition, a facilitator or mediator may help parties in their discussion of some language item, while perhaps also serving as an expert informant. Finally, there is instruction. This is the planned (ex ante) form of mediated non-intervening meta-communication.

Customisation

Now let us turn to situations concerning the user-based changing of a medium system. Procedures for this are of course commonplace, but they rarely involve any explicit, focused considerations involving language-related factors. Cases in which (groups of) users can themselves introduce or modify a limited number of concepts in the medium system are common enough (customisation: the setting of certain software 'preferences'). For example, in most electronic diary systems, entries can be categorised: 'business', 'personal', etc. It is often possible to freely add categories, like 'thesis', 'housekeeping', etc. This type of categorising (and we do not strictly refer to electronic diaries here) also occurs in systems used by groups, which places it in our immediate focus.

Note that in the illustration above, communication about the concepts involved as such is not supported by the system: only implementation of changes in the medium system is. Though there may be some procedure that is followed if ex ante or even ex post conceptual adjustments are to be implemented, conceptualisation will usually have an ad hoc character as long as the set of categories is straightforward enough, is considered useful by everyone involved, and can be freely extended. However, even in the simple exemplary case, excessive growth of the set of categories used threatens to occur, and so does conceptual confusion and even disuse that may result from it. And yet, serious guidance of the conceptualisation process would normally be considered an exaggeration in a situation like the one sketched. Only if the group of users becomes large and varied enough, and the dependence on the system is high, this situation might change. If so, one would expect a fixed set to be defined and imposed by some authority, and that would probably be the end of it.

A relatively small number of researchers has shown explicit interest in the question whether user-based *ad hoc* system adjustment might be structured and even guided, and even fewer have attempted to make *conceptual* adjustment part of the system. Medium systems that feature functionality that is actually *dedicated* to intervening concept-oriented meta-communication do not, to our knowledge, exist[6], though some (experimental) 'groupware' systems come close in providing them (de Moor, 1999), and indeed the field of Computer Supported Cooperative Work (CSCW) has a relatively open eye to some of the problems addressed here. But even there no guided conceptualisation or active concept management occurs. The most far-reaching development in this respect is the introduction of some procedure (usually democratic, often involving the casting of votes) that determines some basic settings of the system. Such a procedure may be supported by the use of groupware (typically, an 'Electronic Meeting Room'); see (Vermunt, 1996).

A more organised, ex-ante form of the customisation of software is involved in the activity of *localisation*: the translation to some national language of software written in a foreign language (Corbolante and Irmler, 2001). With the large-scale international distribution of software packages, localisation has become a real concern, and as a result also an activity for which complete infrastructures are devised. For those

[6]However, most implementation-related, intervening meta-communication is normally included as part of general communication about the system; in our case research (Chapter 7) we actually came across a dedicated communication system to this end, though admittedly it was hardly used as such.

companies who can afford it, efficient and largely automated localisation (covering software updates and consistency across product families) has become possible. Note that localisation may concern a blend of language types as distinguished in Section 4.2.2, notably user language, technical language, and technical instruction language.

4.5 Summary

In the introduction to this chapter, we promised basic answers to two questions: Q1.2 ("What are the environments in which the freezing of language may take place, and how can we describe them?") and, up to a point, Q1.3 ("How do processes and environments for freezing language relate to and influence each other?").

In Section 4.1, the basic (and vital) relation between organisations, communication, and languages was discussed. In addition, some interrelated analytical concepts were introduced and defined that we deem essential for insight-producing description of organisational language domains. Central were the notions of *Discourse Environment* and *Information System Language Community*. We also discussed vocabularies, terminologies, and language artefacts. In addition, we related these concepts to earlier defined concepts. We thus have laid the foundations for describing and analysing the environments of conceptualisation processes (Q1.2).

However, for a complete description of such environments, we also need to look at the languages that may be subject to or product of conceptualisation processes. We therefore provided a typology of language and language use that can be used to analyse and formulate the required targets of conceptualisation. This included a number of relevant aspects of language and language use in ICT-supported environments. We distinguished four main categories of distinction: community-oriented categorisation of language domains, computational versus communicational language use, operational versus auxiliary language use, and language for intra-DE communication versus inter-DE communication. We also identified ten additional aspects of language and language use relevant to the analytical enterprise at hand. Thus, we provided an initial part of our answer to Q1.2.

We went on to discuss our view on meta-communication processes as they occur in ICT-supported organisations: defined them, classified them (again by identifying relevant factors distinguishing them), and provided examples and discussion. We introduced distinctions between ex ante and ex post meta-communication, intervening and non-intervening meta-communication, non-mediated and mediated meta-communication, and language-related and implementation-related meta-communication. We also sketched the typical environments in which particular meta-communication processes take place. We thus continued our answering of Q1.2, and also partially answered Q1.3. In addition, provided an illustration of meta-communication in context of some common ICT activities intended to solve problems with frozen language. This is actually a first beginning of an answer to Q1.5 ("How are processes and environments for freezing language related to information system development?").

What we did not provide was a more detailed analysis of explicit conceptualisation processes (which are a subset of meta-communication processes), though quite a few examples of conceptualisation have by now been given. Consequently, a crucial part of the answer to Q1 ("Which theoretical framework enables us to describe and analyse processes and environments in which language is being frozen?") still needs to be provided. This is our immediate aim in Chapter 5. Also, a more precise and explicit discussion of the relation between conceptualisation processes and their context seems in order (Q1.3). This topic will be briefly revisited in Chapter 5, but it will be most directly addressed in Chapter 6.

Chapter 5

Processes of Explicit Conceptualisation

We now turn to processes that concern explicit conceptualisation: the *explicit* and usually *purposeful* description of language form and (possibly) meaning. This implies that the main focus lies on Q1.1: "what are the processes involved in freezing language, and how can we describe them?". In addition, we are interested in relationships between our general view on conceptualisation on the on hand, and concrete exponents of conceptualisation that occur in various fields of industry and research on the other. This issue is linked up with Q1.5: "how are processes and environments for freezing language related to information system development?". Within the thesis at large, this chapter serves the purpose of providing some core theoretical concepts for describing and analysing conceptualisation processes and activities, in isolation from their wider meta-communicational context. Most prominently, a *phased reference model of conceptualisation processes* will be presented. This and other results will later be placed in context of an analytical framework for conceptualisation processes and their environments (Chapter 6).

As a main inspiration and information source for this chapter, we have studied various fields of application that prominently include conceptualisation-related practices. An elaborate overview concerning the most relevant of these fields can be found in Appendix A. Relevant conclusions of this study are presented in Section 5.1. In the chief section of this chapter (Section 5.2), we present a phased description of the generalised process of explicit conceptualisation based on insights gained through the conceptualisation practices study. We also provide and an overview of types of input and output of conceptualisation, and of properties that can be used to further describe conceptualisation processes and sub-processes (Section 5.3). Finally, we give an overview of main categories of techniques for conceptualisation (Section 5.4).

5.1 Current Conceptualisation Practice

We derive the following general observations from our study of conceptualisation in various fields (Appendix A).

User-oriented approaches to conceptualisation often show respect for language, but also a one-sided view on conceptualisation; they often lose sight of the fact that for language just as for any other aspect, information system development involves various stakeholders, all with their own concerns and priorities. Language-related priority conflicts (communication-oriented, analysis-oriented, implementation-oriented) are thus insufficiently taken into account. There is a lack of comprehensive, in-depth approaches combining conceptualisation-related goals, drives, means, and processes (i.e. approaches reflecting understanding of the complex language game and how to play it). This confirms the indicated need for a well developed functional view on conceptualisation in order to evaluate and make decisions about conceptualisation processes.

Our study of conceptualisation practices has shown that conceptualisation as an explicit topic has hardly been addressed in terms of processes, and thus, in view of our current effort, that such a process-oriented approach indeed deserves more attention. In the rare cases in which finer points of conceptualisation as a process are discussed, they are generally considered in isolation and at an abstract level, not in relation to operational conceptualisation practices. They are not reflected in operational approaches to conceptualisation in ICT. However, the *representation* of concepts is quite elaborately supported by tools and methods (languages, formats, editors, repositories). In other words, current practices in conceptualisation are generally result-oriented instead of process-oriented. In those rare cases where conceptualisation processes are explicitly guided, they typically involve highly elaborate and methodologically sound, formally oriented methods, that unfortunately are too cumbersome and expensive for across-the-board application in ICT practice.

Notwithstanding its essentially language-oriented view on conceptualisation, even the UoD approach has not strongly influenced *techniques* for conceptualisation. Little attention is paid to activities of capturing the UoD and selection of concepts *in view of their specific communicational functionality.* Generally spoken, analysts go out of their way to find the right words and representations of language to fit their envisaged conceptual domains, but they hardly ever construct or even evaluate their work through asking: “will the words *work* in communication?”, or “how can I keep track of the ever changing language in the UoD?”.

Not surprisingly, investment in conceptualisation efforts is currently restricted to cases in which satisfactory ‘return on investment’ is evident. If we believe that conceptualisation should be performed in a controlled and optimal manner wherever this is needed to achieve good language functionality in information systems, then we will have to find ways to make controlled and guided conceptualisation ‘cheaper’ (more efficient and to-the-point), and make requirements, resources, and results more visible so that conceptualisation can be managed accordingly. Consequently, we should try and find approaches to operational conceptualisation that are as thorough or ‘quick and dirty’ as required in various operational situations. In view of the frozen language problem and the observed need for ex post linguistic meta-communication, however, we should also try to enable conceptualisation in heterogeneous, dynamic language environments, and integrate conceptualisation in information system evolution.

5.2 Phased Process Description of Explicit Conceptualisation

In an ICT context, the most tangible, concrete extension of meta-communication processes centres around *conceptualisation processes*, covering activities which result in some representation of concepts used in the MSLC, and that may be subject to 'freezing'. Based on our study of conceptualisation practices, we have put together a generic *reference model* model for processes of explicit conceptualisation. An important handhold for identification and analysis of phases of conceptualisation have been the many *conceptualisation techniques* (including 'methods' and 'approaches'; see Section 5.4) that occur in those fields.

Below we present a fairly elaborate overview of the phases of explicit conceptualisation. Importantly, the description involves many phases that are implicit or even optional in practice. We distinguish five basic phases of conceptualisation:

1. Acquiring raw material
2. Capturing the Envisaged UoD
3. Selection of Relevant Concepts
4. Naming and Defining Selected Concepts
5. Quality Check

In many cases, the activities involved in first three steps are performed on a strictly intuitive basis. Even if several agents take part, it is not uncommon that suggestions are put forward *ad hoc* (by whoever takes the initiative), and are then discussed and changed if need be, until agreement is reached. This is not an approach that should be viewed as inadequate in principle. People may draw very successfully on their natural capacity for language creation and adaptation, and their intuitive knowledge of conceptual needs in context is perhaps the most powerful mechanism for symbol selection available to us. Nevertheless, we attempt to add some methodological detail to this intuitive 'black box' activity, and categorise some sub-activities. This should be especially helpful in cases that are too complex or opaque to be solved by intuition alone.

The reference model of conceptualisation presented involves quite a number of steps and activities, but please note that *actual, controlled implementation of all individual steps and activities distinguished is not likely to occur in any but the most elaborate of conceptualisation efforts*. Our model is put together for analytical purposes and is meant to be comprehensive rather than prescriptive.

5.2.1 Cooperative Conceptualisation

Before we present the complete phased reference model for conceptualisation processes, some words need to be devoted here to the important issue of *cooperation* in conceptualisation. We have seen that within our theoretical framework, conceptualisation is seen as a form of meta-communication, dealing with agreements and conventions concerning concepts and concept meaning. Consequently, questions arise concerning who are involved in some phase of conceptualisation; who they agree, negotiate, or cooperate with. Also, we will need to establish how an acceptable degree of communal agreement may be reached in some situation.

We observe that the manner and nature of cooperation, even in a fully 'democratic' conceptualisation effort, will seldom be performed democratically every little step of the way, mostly for practical reasons. Most viable seems an approach in which a small group of representatives work together in order to propose some solution they hope will be embraced by their fellow community members, and if it is rejected they will return to the job and try to improve the solution according to the responses of the community. In other words, usually only results (including intermediate ones) will be presented to some larger group who judges them. Consequently, many subphases in the process may be performed by small numbers of people or even by specialised individuals.

In any case, we believe that an analysis and description of the matter of cooperation in conceptualisation (sub-)processes can be realistically distinguished from a description of the phases of conceptualisation as presented in this section. In fact, we believe this is a better approach, since it allows for the recognition of a dedicated view on cooperation which can then be *applied* to conceptualisation processes, as it can also be applied to any other construction process. Indeed we assume that conceptualisation processes are generally similar to other cooperative construction processes.

In the case conceptualisation takes place in a computer supported environment (i.e. if the conceptualisation process itself is supported by ICT), it may well become a subject of study within the field of Computer Supported Cooperative Work: 'how to do CSCW in conceptualisation' (see section A.2). In fact, we see this as fertile ground for further research (Section 8.2).

5.2.2 Phase 1: Acquiring Raw Material

This preparatory phase involves the collection or creation of material that is to assist actual, purpose-driven conceptualisation (as part of the development of a medium system for the DE in question). Raw material provides input and inspiration for actual explicit conceptualisation. While many of the concept forms that later become 'explicitly conceptualised' first surface in this phase, taking them for granted is a bad idea since one cannot easily be sure what the conditions and context were in which the language material was produced.

Conceptualisation has to start somewhere, and has to be based on something. Often this will simply be the current view of a situation as it exists in an analyst's mind (typically, tacit and subjective knowledge). Attitudes to such an intuitive basis of conceptualisation vary strongly: whereas most textbooks simply assume that basic human intuition, insight, and analytic skills are the primary drive behind the analysis process, some emphasise that a more controlled approach is called for, because systematic, generalised analysis and validation of the domain description is required. This inherently means that other sources than just the analyst's (or informant's) intuition and experience should be involved, and such sources we generally label *raw material.*

We roughly define the relation between raw material and language artefacts as follows:

- raw material concerns language artefacts, that
- do not contain any expressions of *explicit* linguistic meta-communication, and
- the utterances contained in which have not been *intentionally* produced by a speaker/writer with explicit conceptualisation in mind.

The material in question may take many shapes. Samples of communication taken from the DE under analysis are perhaps the most typical form (database entries, forms, documents, even recorded conversations) but included may also be process descriptions, scenarios, case descriptions, product descriptions, scripts, diaries, diagrams, etc. Some important techniques used to acquire raw material for conceptualisation are listed in Section 5.4 and described in Appendix B.
Choices made concerning raw material can be essential for the entire conceptualisation process that follows. The following main questions should therefore be addressed:

- What is the domain involved; how can it be demarcated?
- What is the nature of operational communication in the discourse environment(s) involved?

However, it may well be that some of these questions can only partly be answered in this preparatory phase, and must be revisited in, or even postponed until, phase 2; in that case, raw material should be gathered with comprehensibility in mind rather than focus, in order not to miss out on some conceptual aspect or sub-domain. There is a trade-off between, on the one hand, drawing premature conclusions about the conceptual domain and discourse environment, and on the other the ad hoc acquisition of raw material. It seems best to first perform a 'quick scan' in Phase 1 and consolidate or reconsider the preliminary results in Phase 2.
Some questions typically related to the acquisition of raw material are the following:

- What sort of artefacts already exist in the discourse environment (if any), and how do they relate to the conceptualisation process?
- What is the nature of concepts that are to be found in each artefact, and how does this relate to the conceptualisation goals set?
- What sort of knowledge is associated with various agents/roles in the DE, and how does this relate to their role in conceptualisation (and in creating raw material)?
- Is the effort of acquiring raw material justified in terms of resources available weighed against actual necessity of using it in conceptualisation?

The reader may find it hard to distinguish between the creation of raw material (Phase 1) and initial description of the activities in the discourse environment (Phase 2). The main difference between the phases lies in their respective *purpose*: providing exemplification and inspiration (Phase 1) versus performing active initial conceptualisation (Phase 2). In the latter phase, the material is produced in a consciously performed conceptualisation effort and in a controlled context. Nevertheless, if raw material is specially created in order to aid a conceptualisation process (i.e. not created at an earlier date for another purpose), chances are it can readily be used as material for Phase 2. It all depends on how sure one can be of the context in which it was produced (chiefly, with what purpose in mind). Note that the output of neither Phase 1 nor Phase 2 conceptualisation processes involves expressions of explicit meta-communication (which is what defines the main difference between Phases 1-2 and Phases 3-5).
In view of the above, we distinguish the following three sub-phases in Phase 1:

1.1 Perform preliminary evaluation of conceptualisation domain, DE, and communication goals

1.2 Decide which existing sources of raw material to use

1.3 Acquire the raw material

5.2.3 Phase 2: Capturing the Envisaged UoD

If conceptualisation aims at carefully *copying* the vocabularies used in the discourse environment under analysis (i.e. the present UoD), techniques can be used that record actual communication (a Phase 1 activity discussed in Section B.10). If factors come in that require more complex considerations (practical factor, foreign terminologies, etc.), this approach will not do. In such cases, the capability of individual humans to somehow *envisage* some communication situation and suggest concepts accordingly needs to be harnessed. Yet of course this introduces some major risks: if informants are biased, ill-informed, or fail to mentally relate to (even imagine) the communicative situation intended, and the analyst is unaware of this, then using the material rendered by such informants results in a very bad start of the conceptualisation process indeed. However, these risks are part of the game when we set out to freeze language, and we will have to cope with them. The effort of achieving this is central in Phase 2.
First, some idea must be conceived of the situation(s) the concept will be used in, and the communication that will be performed through the medium system under development. It is important to make sure the DE (and its boundaries) becomes clear, and that various situational factors and insights that apply to the *whole* conceptual domain of the DE (as opposed to some individual concept) are noted and charted. As a result of the analysis, it may well be necessary to distinguish sub-DEs not hitherto identified. The input for this phase consists of the general knowledge of a DE and its UoD as contained in the minds of the people involved in this phase, complemented with any raw material available.
Next, it will generally be useful to produce a 'mentally processed', conceptualisation-oriented description of the particular tasks or activities that belong to the DE. Such

a description can be expressed using natural language descriptions or diagrams (or both). This is where the first, preliminary representations of individual concepts take shape, unless it is decided that 'raw material' should be copied; yet in the latter case, the relation between the raw material and the envisaged communication situation should at least be carefully considered.

With 'mentally processed' we mean 'actively (re)phrased by a person who is aware of the context and focus of the conceptualisation effort at hand.' In other words, the situation description initial to the conceptualisation process must be *purpose-made* and the result of *context-specific interpretation and phrasing.* Though existing descriptions ('raw material') may well be used as inspiration, accepting such material as direct input for phase 2 concept description is risky and we advised against it unless its context and the goals for which it was originally produced are well known.

As an illustration of why 'mental processing' of the situation and activities may be important, consider the use of video or audio recordings. These are exact, mechanically captured representations of domain activity; they are not mentally processed. It may seem that since such material is 'objective', it is an excellent basis for user-based conceptualisation. Whereas it is certainly true that recordings may provide excellent data for study of some communication domain (see Section B.10), explicit conceptualisation, once it is engaged in, often includes rational consideration: a degree of considered artificiality so to speak. While a choice *may* be made to strictly keep to user language as originally encountered in the UoD, it may also be the case that various situational factors (see Chapter 4) require a more engineering-like approach to conceptualisation. Choices in this respect should ideally be made deliberately and on the basis of a complete situational analysis. Of course, raw material may *underly* UoD description as performed in Phase 2, and recordings are a good example of such raw material.

We have defined the UoD as follows:

$$\mathsf{UoD}(d) \equiv \bigcup_{g \in \mathcal{AG}} \mathsf{Vocabulary}(g, d)$$

In plain English, the UoD of some DE is "the combined vocabularies of all agents relevant to the DE" (Section 4.1.3). Consequently, we cannot realistically expect for the complete UoD to be captured by the descriptions produced in Phase 2. We prefer to explicitly suggest the notion of 'satisfactory vocabulary coverage within the DE', instead of pretending that a complete description of the vocabulary used in a DE (UoD) is achievable.

Through choosing representative informant-agents ('domain experts') and the activities and situations they are to describe, it may be that only a fragment of individual vocabularies from the DE is actually available for explicit conceptualisation. Consequently, particular concepts may from the onset be excluded from the description, along with their 'owners'. While this seems unavoidable, we should at least be conscious of this, try to minimise damage, and assess the risk involved. Consequently, we consider it highly important that due attention is paid to the following practical points:

- Make sure that the representatives of the agent community in the DE are realistically capable of describing or producing the vocabulary to be used in the envisaged DE.

- Make sure that the perspective from which the representatives look at the DE does not exclude other important perspectives.
- In Phase 2 as well as Phase 1, strive for completeness rather than relevance (though necessarily within the boundaries of what is practically possible).
- If clear choices are made to include or exclude some vocabulary, make sure this is at least done for a clear reason, and document that reason.
- Do not hesitate to confront representatives from the DE with examples of vocabulary use as present in raw material (if available) in order to make sure that no important perspectives on the DE and UoD are missed

In many cases, the DE will already include various medium systems. Note that they may have a considerable influence on the existing UoD, and that vocabulary stemming from use of such medium systems is highly relevant, even if the goal is to change or abandon some medium system(s). However, it is recommendable to use the description of the actual system interface merely as raw material, and not to confuse the medium machine itself with the impact it may have on the vocabulary of *agents* in the DE, and hence the UoD.
Summary of Phase 2:

2.1 In a controlled context, form an idea of the situation(s) in which the concepts that are to be conceptualised will be used

2.2 Capture the language as used in the envisaged situation(s) and activities

5.2.4 Phase 3: Selection of Relevant Concepts

The primary activity in explicit conceptualisation processes is the selection of concepts (tangibly: concept forms) that are to be a definite part of the language to be specified. Such selection may involve form-related considerations, but underlying it are usually also considerations of which 'isolated bits of meaning' should be covered by the language.
At this stage in the conceptualisation process, it is quite possible that it is decided that *no closed communication is required after all*, for example because on the basis of the UoD study (Phase 2) it is found that a particular communication activity is superfluous, or because there is no need (or possibility) to resort to closed language communication in the DE. Quite differently, it may also transpire that the conceptualisation process will take up too many resources, and is therefore to be abandoned (note that this decision may in principle occur in earlier phases as well).
Once Phase 3 is actually entered, the 'mentally processed' descriptions of the envisaged DE situations, or possibly recordings from the user domain, can be now scanned for *relevant concepts*. If we look at this process in more detail, we observe that two approaches are usually combined here: form-oriented analysis and meaning-oriented analysis. The first concentrates on visible representations of concepts, the second on their underlying meaning. The two approaches are sufficiently co-dependent to present them as two coinciding sub-phases: 3.1a and 3.1b.

Ideally, each word in the description should be evaluated in particular with respect to its *communicational functionality*: whether it can be used to convey information that is expected not to be part of common knowledge. How strictly this criterion is applied should depend on the risk one is prepared to accept (also in view of resources available) that some meaning will not be expressible (or some speech act not performed) through the medium system under development. Hence, how Phase 3.1 is performed should depend on the communicative situation as analysed previously (Phase 2.1).

Next, perhaps the most essential step needs to be taken: someone has to actively suggest which concepts will constitute the preliminary closed set of concepts that will eventually be built into the medium system. This arguably is the core Phase in conceptualisation, possibly alongside Phase 4.3.3.5. (see below). Analytically, it may involve a whole range of intertwined considerations (also see Phase 5), yet we propose to let a careful but creative, intuition-driven approach prevail at this point. Note that to propose concepts does not yet mean to propose *definitive terms* (see phase 4), but that the conceptual description is expressed through *candidate terms.*

Summary of Phase 3:

- 3.1a identify forms of potentially relevant concepts
- 3.1b consider which concepts are needed for communication through the medium system
- 3.2 isolate those concepts per situation by listing and describing them

5.2.5 Phase 4: Naming and Defining Selected Concepts

Obviously, this phase is closely related to Phase 3. However, in Phase 4, matters of *concept representation* (about both form and meaning) are to be taken into account, whereas Phase 3 focusses on the *selection* concepts. Thus, Phase 4 involves a different set of considerations than the isolation of key concepts (Phase 3).

First, the concepts selected in Phase 3 may be *represented* in a number of ways by what we called 'definitive terms'. The main question here is: which terms are the best possible 'labels' for the concepts that are being frozen? Inter-domain considerations are particularly important here: how to combine the different conceptual sub-domains, the languages used in them, and matters of translation ('mapping') between them. It may be desirable to use different terms for the same meaning in different situations (synonymy), or allow for various meanings to be covered by one term (homonymy), and so forth.

The next step in Phase 4 concerns the definition of concepts (linguistic meta-communication). We refrain from discussing this sub-phase here, and devote a separate section (5.2.7) to it.

Summary of Phase 4:

- 4.1 consider appropriate labels for the selected concepts, for each communicational situation but also in view of combined situations
- 4.2 put together a list of definite terms per communicational situation
- 4.3 define the concepts, if necessary

5.2.6 Phase 5: Quality Check

By now there should be available a carefully considered, clearly described set of concepts (possibly including various sub-sets for different situations). If the situation calls and allows for it, this potentially complex whole should be thoroughly checked. Depending on the situation, a number of factors (many of them linguistic in nature) should be taken into account. However, note that the quality check by nature takes place in hindsight and *cannot fully replace the appropriate procedural choices made in phases 1-4*: the check is performed on concepts made explicit, but cannot include those concepts or expressions thereof that have been missed or rejected in the analysis and selection process.

There are two types of consideration involved: those related to the coherence and quality concerning the set as a whole, and those related to the description of individual concepts. Some factors relevant to the quality check will be further discussed in Chapter 6. However, note that the *choice of relevant factors* for a certain DE largely depends on the outcome of general analysis of the DE performed in Phase 2.1.

Summary of phase 5:

5.1 systematically check factors determining the quality of the set of (interrelated) concepts

5.2 systematically check factors determining the quality of individual concept descriptions (including their definitions, if present).

5.2.7 Phase 4.3: Definition of Concepts

We continue by describing the general process of the definition of concepts, inspired mostly on a lexicographical perspective but with parallels in information analysis. This discussion is a continuation of the previous section. We have already presented definition as Phase 4.3, and keeping to this the numbering of definition process phases starts with Phase 4.3.1; for clarity's sake, we will sometimes represent this as '(4.3.)1'.

The definition process description below is derived from (Viskil, 1994), as earlier discussed in Section 3.4.2, but we have adapted his findings to our current discussion and have changed, added, or left out certain elements as we though appropriate.

As is the case with selection of a closed set of concepts, the formulation of a definition can be presented as an elaborate process involving many possible steps, but it also has its intuitive natural language counterpart: in open meta-communication, every language-using human being has experienced explaining a word to someone else. It may go too far to call this "definition" (therefore we often use the weaker notion "meaning description"), but it is clear that such cases concern a conversation about word meaning. Again we emphasise that this natural human talent should not be played down, but embraced as a powerful resource, though perhaps guided and backed up (when called for) with additional check lists, tools, and techniques.

Though the process of defining a concept is in principle optional in explicit conceptualisation, it is often required and can then be difficult and labour-intensive. Though intuitive, ad hoc meaning description will work well enough in a number of

cases, we emphasise that definition may also be a tricky business, and when engaged in in a sensitive context (e.g. legally, operationally, technically) it should be carried out with great care. Consequently, in many cases it seems recommendable to keep definition *minimal*, and base the procedure followed on a good understanding of options available, their 'cost' (capabilities, resources), and their implications for future language use.
The definition phase is divided in four sub-phases:

4.3.1 Determine the goal of the definition

4.3.2 Analyse the meaning of the concept

4.3.3 Construct (author) the definition

4.3.4 Quality check

Phase (4.3.)1: Determining the Goal of the Definition

Phase (4.3.)1 as a whole is especially important if the communicating parties will have only the provided language documentation to help out in case of concept-related trouble (see Section 4.4). The goal of a specific definition is related to the intended use of the concept defined, but also to the goal of the definition itself. It is therefore related to both operational communication (language use, Section 4.2) and meta-communication (Section 4.3.3). As an example of the latter, whether or not the definition will be used in a legal context has implications for the appropriate style of the definition, apart from the goal of the concept in contextualised use. Definition also depends on who the definition is written for: whether the nature of the users is known at all, how well-specified the user group is, and what kind of definitions they are used to, demand, or can cope with.
Overview of Phase (4.3).1:

4.3.1.1 Choose symbol/concept that is to be defined

4.3.1.2 Establish for which goal the definition is needed

4.3.1.3 Establish for who the definition is written

4.3.1.4 Establish what type of definition is needed

Phase (4.3.)1.1, the choosing of a concept form (word) to define, may seem simple enough, but note that it entails a decision about whether or not to engage in definition. Once a definition exists, the word has firmly become a *term*, and, in a manner of speaking, it has lost its innocence: the definition attached to it may be used to cut short discussions about meaning and may well carry some authority. Coining a definition, specially in the context of ex ante meta-communication, involves the taking of some responsibility. Also, a definition may be (or become) a burden: it may awkwardly or even needlessly fix interpretation of a word that would otherwise be treated more 'liberally' (and perhaps very successfully so), or the definition may

overrule other useful meanings (homonyms) of the word. Also, a definition may *outlive* its use[1]. Finally, some thought should be spent on the form chosen to represent a concept, but this mirrors considerations already touched upon in 4.1 ('naming concepts').
Phases (4.3.)1.2 and (4.3.)1.3 are best performed in parallel, since they are closely linked. Questions that need to be answered include the following:

- For what purpose will the definition be used?
- What authority will the definition bear?
- What is the risk if the definition is insufficiently precise?
- What is the risk if the definition is *too* precise?
- Who will use the definition, will all stakeholders be able to understand and use it, and will they benefit from it or be burdened by it in some respect?

The choice of *definition type* is important because it influences the further definition process. Recall that the three[2] main types of definition distinguished in our framework are the following (see Section 5.2.7):

Lexical definition describes what a group of people can be *observed* to mean with a word.

Stipulative definition describes what a person or group '*chooses* to mean' with a word. It asserts a meaning rather than an observing it.

Stipulative-lexical definition asserts a meaning, but uses an observed (lexical) meaning as object of stipulation.

Phase (4.3.)1.4 involves the choice between the definition types mentioned above. Other important choices, most typically involving *representation*, come later. The choice of type is related (answers to) to the following questions:

- Does the definition aim to describe an existing convention shared by many?
- Or does the definition aim to make clear a particular meaning to be used in a particular DE or involving some particular task?
- Will the lexical knowledge of agents confronted with the definition be changed substantially by it, or will the definition mostly confirm what the agents already know?

It would go too far to elaborate here on strategies to follow when choosing the 'right' definition for some situation. The phases and criteria described should not become a burden in themselves, but are meant to suggest some important points for consideration and to make clear that whether and how one engages in definition may have consequences for meta-communication and thereby also for communication in general.

[1]In this respect, consider that it may at times be wise to *abandon* a definition and perhaps very explicitly 'set free' or 'unfreeze' a word.

[2]Factual definition is 'not meta-linguistic'.

Phase (4.3.)2: Analysing the Concept

This sub-phase, even more than the previous one, may seem cumbersome and over-detailed. Again, we do not want to suggest that it should be followed by the letter; it is merely intended as a guideline and as an overview of the kinds of considerations that might be made.
The sub-phases in (4.3.)2 are strongly related to matters of *meaning description*. Since we have repeatedly made the point that we want to keep away from detailed matters of meaning representation, this may raise some eyebrows. Indeed, we make an exception here to our general discarding of techniques for meaning representation. However, the activities related to meaning description mentioned below are primarily *analytical*; indeed it may be a good idea to apply them to the terms under analysis even if the findings are not explicitly recorded in the eventual definition. Also, the list of techniques under (4.3).2 is exhaustive nor prescriptive. It is merely illustrative of typical considerations in lexicography and terminography.
We sub-divide Phase (4.3.)2 in (4.3.)2a and (4.3.)2b. Analysis as sketched in (4.3.)2a is particularly useful if it is not clear what the meaning of a word is, and an *explorative* analytical process is called for, going beyond a particular context. Description of meaning as intended in a particular context is focused on in Phase 2.b.

First option: Phase 2.a, concerning possible meanings related to the word form

This option is associated with the lexical approach to definition rather than the stipulative approaches.

4.3.2a.1 Consider all major existing meanings of the word

4.3.2a.2 Consider existing derivations of the word, or of the stem form of the word; also consider compound forms including the word

4.3.2a.3 Check for synonyms of the word in other languages or domains

4.3.2a.4 Find out which collocations the word regularly has[3]

4.3.2a.5 Find out what are the 'relata' of the word[4]

4.3.2a.6 Check for existing antonyms, hyperonyms, and hyponyms of the word

4.3.2a.7 Find out the etymology of the word

[3]Collocations concern combinations in which words often occur, for example "blond" collocates with "hair". Importantly, collocations are not the same as compound words: they may be distributed over a phrase but still 'belong together': "Ms. Schiffer's hair could quite definitely be classified as blond".

[4]Relata can perhaps best be described briefly as 'case slots'. For example, for the verb "cut", we can determine that *someone* cuts *something* with *something*. As such, relata are an important connection to contextualised meaning.

Second option: Phase 2.b, concerning concept meaning as intended in a certain situation

Phase (4.3.)2b homes in on meanings in specific domains or contexts, thereby including those *pragmatic* aspects of word meaning that can be usefully described at all, and also the particular *semantic* aspects of a word as related to its use in the envisaged DE. Though lexical definition may certainly play a role here, it is expected that stipulative definition (and stipulative-lexical definition) are generally more relevant in this phase.

4.3.2b.1 Describe possible meanings of the word in the intended situation, by the defining party or by others (stipulatively or lexically), as relevant

4.3.2b.2 Find which concepts are *related to* the intended concept in the intended situation (charting its semantic field)

4.3.2b.3 Determine to which semantic categories the intended concept may belong *in the intended situation*

We include the determination of semantic category under Phase 2b instead of 2a because we consider the assignment of semantic categories to be a subjective, situation-dependent matter rather than an instrument of 'general lexical definition'. However, it cannot be denied that it might also have been included under Phase 2a. Many different systems of semantic categorisation can be (and have been) conceived. The one Viskil uses, for example, distinguishes between *property* concepts and *system* concepts; the first are further divided in *absolute, relative, gradual,* and *comparative* concepts, the second in *process-concepts, product-concepts, concrete concepts*, and *abstract concepts*. More commonly, categories are based on relations between concepts, such as 'IsA' (hyponymy) and 'PartOf' (meronymy). Refined categorisation is particularly useful if the definition is related to formal or semi-formal analysis, for example for information analysis or database design; it also occurs regularly in ontology building (see Appendix A). In many other cases, it will be superfluous. Yet the hyponymic or 'IsA' relation mentioned above plays a special part in a standard approach to definition known as the description of 'genus and differentia'. For example: 'a dog (definiens) is an animal (genus) that has a tail, barks, and bites (differentiae)'. In other words, the genus stands for the generalised type of the concept described, the differentia for what distinguishes the 'definiens' (concept to be defined) from other concepts that fall under the same generalised type (for example, 'cat'). Determining the exact genus and differentiae of a concept can turn out much harder than anticipated (see for example Lakoff, 1985), and depends greatly on both the general conceptual system (ontology) used by the defining party and the context (concept domain) to which the definition should apply.

Phase (4.3.)3: Constructing a Definition

Having made decisions about the nature and goals of the definition and the concept involved, and having explored the possible and intended meaning of the concept, we now need to actually *choose a particular meaning* and then *create and represent* the definition. Therefore, some previous considerations are revisited. Note that phase

(4.3.)3 is often the only explicit step taken in definition authoring; we hope to have made it clear that preparation should optimally involve more than just the *ad-hoc* authoring of a meaning description.
It is important here to realise that a definition has a status of its own, that in some respects it stands apart from the essential ('cognitive') meaning of a word as such. Choices made in the definition are part of meta-communication, even if it is mediated or ex ante, and essentially serve to support language use of some sort. By nature, definitions can never capture the full richness of meaning, not can any linguistic definition be 'definitive' or 'universal' (see Chapter 3).
Having paid due attention to the elaborate warnings and suggestions above, we can then finally proceed and author a definition.
Overview:

4.3.3.1 Consider the extension the concept is to have in the target domain

4.3.3.2 Determine which existing meanings are incorporated in or compatible with the concept to be defined

4.3.3.3 Determine which existing meanings are different from the concept to be defined

4.3.3.4 Determine in which semantic category the concept falls

4.3.3.5 Roughly represent the definition of the concept

4.3.3.6 Write the definitive definition, taking various factors of *presentation* into account

By focusing on the *extension* of a word aimed for in the conceptual domain (practically spoken, a listing of exemplary items covered by some term), it is emphasised that we should not blindly assume the correctness of *intentional* description (a generalised description of the attributes shared by the items covered), however carefully crafted it may be. There is a real risk that the authors of a definition fall in love with some elegant or well-versed definition that however fails to be meta-communicationally functional. Given the outlook of the relevant stakeholders within the target DE, we should keep an open eye for what actual occurrences, references, and situations the defined meaning is to cover. Often, this cannot be an exact process; it is more a matter of perspective and insight. Nevertheless, to evaluate a definition from the point of view of the actual extension it should have *in the domain at hand* can serve as a refreshing 'quality check', and is highly recommended. Fact-type related techniques in relation to phases 1 and 2 may be helpful here (see Section B.13). Note that considering extension may also be used as a reflective device in both phase 4.3.2a and 4.3.2b.
Once we have some idea of the extension of a definition as intended within the context at hand, we can turn to selecting the intentional descriptions that should cover the extension. If the results from previous phases are explicitly available, we may systematically determine which intentional descriptions should be included or excluded: weed out the unwanted elements. In this process, the intended *level of detail* of the definition should also be taken into account. It is not uncommon to explicitly mention synonyms or antonyms of a word in its definition, as well as

examples of its extension, or even of items explicitly *excluded* from its extension. In principle, all devices listed under (4.3.)2 may be used, but in many cases this would be an exaggeration, counter-productive given the limitations of the audience.
With the goal and particular application of the definition in mind, the preferred meaning of the word can finally be selected from the results of Phases (4.3.)2a and (4.3.)2b. Through selection of the definitive descriptive method or paradigm for meaning description is related to Phase (4.3.)3.6, admittedly preferences in this respect will have considerable influence on the whole of Phase (4.3.)3, perhaps even on Phase (4.3.)2.
Some definitions are associated with a particular system of categorisation (see Phase (4.3.)2b). Note that such a system typically applies to a large collection of words rather than to individual definitions. 'Categories' can cover anything from domains of use in all their variation (see Section 4.2.1) to semantic categories, and often also *syntactic* categories (for example: noun, verb, adjective), reflecting relational (combinatory) potentiality of the concept in question.
Two steps within (4.3).3 now remain: rough description ('sketching') of the definition, focusing on organising its final, comprehensive content, and definitive authoring. Sketching a preliminary definition probably is the most creative activity of conceptualisation, one that has been intuitively performed though the ages without help of the substantial apparatus presented in this entire section. In combination with Phase 3.2, it is arguably the the core phase in the process of explicit conceptualisation. In the final step, representational factors come in that go beyond pure description of meaning: meta-language as used in comparable definitions or the dictionary framework, the medium through which the definition will be published, even matters of graphical design. We will refrain from further discussing them here; in some cases, choices made will relate to goals and intentions determined in earlier phases of the process.

Phase (4.3.)4: Definition Quality Check

In this phase, the factors determining the quality of the definition proposed (strongly related to the considerations involved in in Phase (4.3.)1-3) are systematically checked. The place of the quality assessment (4.3.4) is perhaps misleading in that the quality check of a definition, like the one on concept selection, is performed in hindsight. Though it is certainly recommendable to perform a final quality check after a definition process is finished, we more specifically refer here to a quality check that may occur some time after the initial authoring of the definition, perhaps iteratively. Therefore, this phase can also be seen as a part of phase 5.2 in the general conceptualisation process. Once again, we emphasise that not so much the *order*, but the *nature* of the phases is important.
The crucial difference between the Definition Quality Check and the General Quality Check (Phase 5) is that in Phase 4.3.4, the emphasis lies on certain meta-communication related quality factors, involving the *choice of definition type*, *approaches to meaning description*, and even the *appropriate representational languages*. As mentioned, in this thesis we do not discuss the representational possibilities at any length; however, the framework we provide could be used to position and even evaluate particular representational languages (Section 8.2). Still, we can

list some basic 'quality check points' specific to definition as a process (derived from the process description above):

- What style of definition is appropriate in view of its use(s) in meta-communication?
- Which stakeholders or agents in the DE can be realistically expected to ever use the definition in meta-communication?
- What authority does the definition bear, and is its form and quality appropriate in view of this?
- Has due attention been paid to (alternative) existing meanings of the word defined that may exist within the MSLC?
- Is the *level of detail* at which meaning is described in the definition suitable?
- Has due attention been payed to possible synonyms and homonyms of the word defined, and should these be included in the definition explicitly?
- Has due attention been paid to (the difference between) general and situated meaning, and if necessary, has adequate differentiation between these factors been made explicit in the definition (and selection) results?
- Does the definition include categorisation according to an appropriate category system, and is such categorisation useful in the first place?
- Have matters of presentation of the definition been properly taken into account?

This concludes our discussion of a phased description of conceptualisation processes. The phases may be referred to in relation to particular kinds of activity involved in conceptualisation, either in context of specific techniques for conceptualisation (belonging to particular phases) or in view of infrastructure provided to support conceptualisation.

5.3 Conceptualisation Processes: Types and Properties

We have been developing elementary means for the description of conceptualisation processes. After describing the environment they take place in (Chapter 4) and after describing the general phases of conceptualisation (previous section), we can now finish by presenting a number of descriptive distinctions that we can make with respect to conceptualisation processes and sub-processes. This provides a useful instrument for the identification and evaluation (in context) of conceptualisation processes.

In addition to the identification of process phases, we provide a description of each main process phase in terms of its input and output (I/O). This will enable us to link particular language artefacts and expertise we may encounter in discourse environments to specific (phases of) conceptualisation processes. We also discuss general properties of processes as applied to conceptualisation processes.

5.3.1 Conceptualisation Process Reference Model (CPRM)

The phased model of conceptualisation processes as described in the previous section takes a central place in our descriptive framework for conceptualisation processes. It should be possible to position every conceptualisation process, at least by approximation, with respect to the phases of the model as concisely presented in figure 5.1.

5.3.2 Input and Output of Conceptualisation Processes

A common approach to the description of processes is to describe their input and output. Such a 'black box' description (discarding whatever happens inside the black box in order to transform input into output) is especially helpful if the descriptive situation calls for abstraction of the transformation process as such. This is, however, not our incentive. We are most certainly interested in what happens inside the black box (as has been elaborately discussed in the previous section), yet we believe it is simply also useful to look at input and output of conceptualisation processes, since this provides the crucial link with their environment. In addition, we consider input and output not only of a 'generalised conceptualisation process' but also of relevant sub-processes (phases); hence, the black box is opened up and becomes a Chinese box.

We do not engage in a classic data-oriented I/O description as is commonly performed in fields like data analysis and software engineering, since this would require (and here we go again) discussion of actual conceptual representations (or types of representations). This is not to say that data-like I/O description concerning conceptual representations would not be a productive approach; in fact, it is very useful indeed on an operational level of conceptualisation. Leaving it out is only a matter of focus.

We instead present a 'higher level' description, focusing on *artefact types* and *knowledge types* as input and output, and also consider *means and resources* required to achieve processing. Consequently, our discussion has ingredients in it that may remind the reader of such fields as 'document management', 'knowledge management', and even 'information architecture'.

Language Artefacts

Artefacts (more in particular, *language artefacts*) have been introduced in Section 4.1.2 as "persistent physical occurrences of a (complex) language form". Note that these may be used for both operational communication and meta-communication (an invoice is a language artefact, a dictionary too). Language artefacts 'carry information'. We have elaborately discussed our views on this in Chapter 3. We could discuss input and output of conceptualisation processes in terms of 'information'. However, we prefer to take a more practical approach and instead discuss the 'carriers' (artefacts and agents), as can be more readily identified in real-world situations. We do therefore not concern ourselves with detailed description of artefacts outside the scope of their functioning as input and output of conceptualisation processes. For completeness' sake we list some properties of artefacts that may also be of interest to the general situational (DE) analysis (for more see Chapter 6):

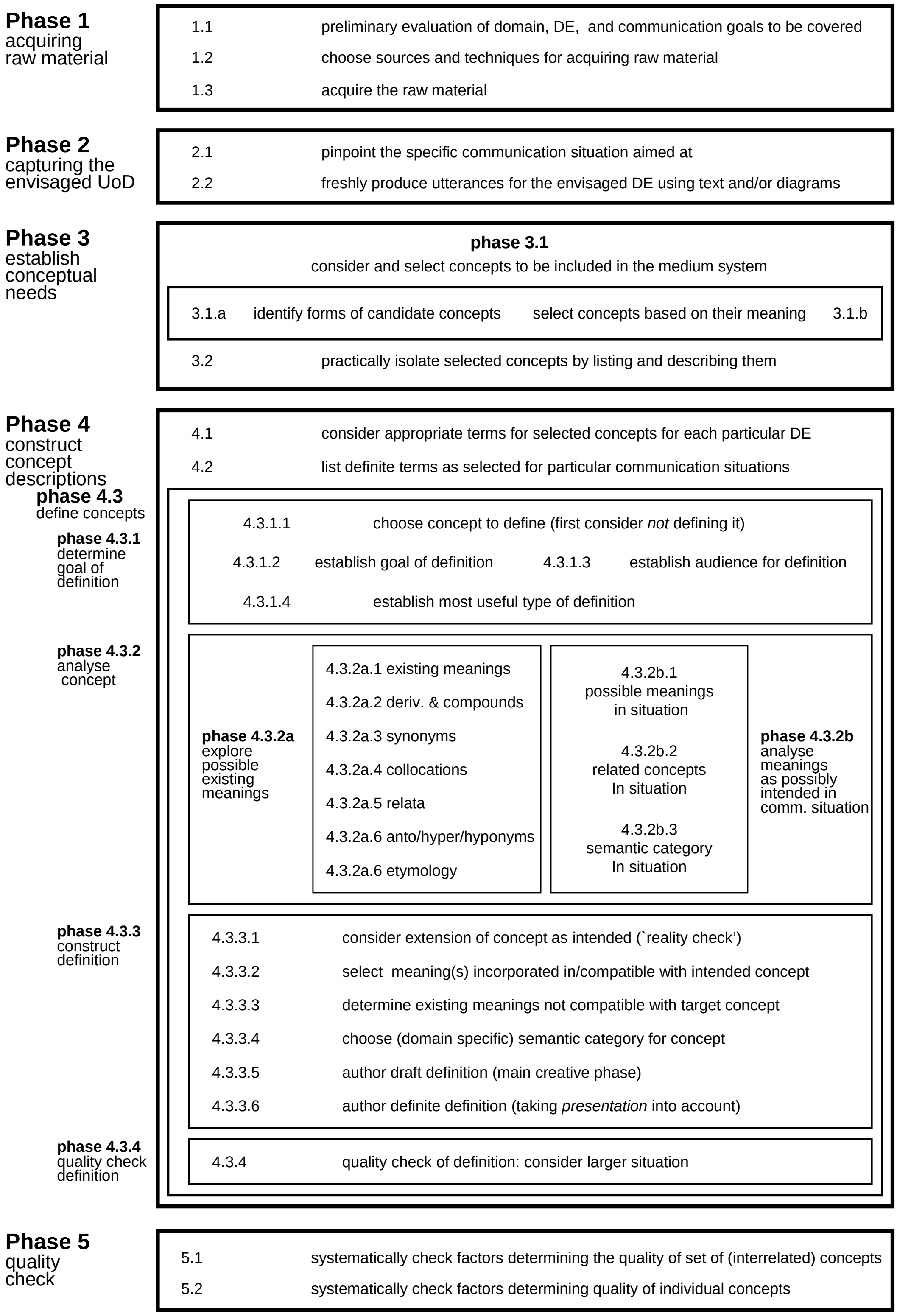

Figure 5.1: Conceptualisation Process Reference Model

- Medium: on what medium is the artefact based? (e.g. paper, web page, etc.)
- Source: how was the artefact acquired, where was it stored or found?
- Author: who created the artifact?
- Organisation: what organisation is responsible for the artifact?
- Authoritativeness: how authoritative is the artifact? (is it a guideline, or a 'law'?)
- Distribution/availability: how widespread is the artifact in the organisation? How many agents use the artifact?
- Version: how frequently is the artifact renewed? How can a new version of the artifact be obtained?

Helpful within the scope of this section (and in view of our general approach to process description) is a typology of language artefacts, tuned to input and output of conceptualisation processes. In table 5.1 we present an overview of the I/O requirements of particular phases in our reference model of conceptualisation processes. We have skipped those phases that need or render no relevant artefacts.
It is quite simple to derive from the detailed overview in Table 5.1 a listing of relevant input/output artefact types for some instantiated conceptualisation process, provided that such a process can be analysed in terms of the phased reference model. If a particular phase is explicitly performed, the related artefact types are relevant and are to be added to the general collection of artefact types for the conceptualisation process in question.
To conclude our discussion of language artefact types, we list the main types that can be derived from Table 5.1 and provide some additional comment for each type.

Descriptions of DE tasks and activities typically take the form of NL texts or (informal) diagrams. They include scenarios, case descriptions, procedures, organisational descriptions, etc. We distinguish three sub-types: *existing* (previously authored outside the context of the conceptualisation effort), *selected* (probably 'existing', but nevertheless selected as useful input for conceptualisation), and *new*: specially authored for the conceptualisation effort.

Communication samples from the DE are written or even audiovisual records or recordings of communicational activities. In other words, they represent utterances actually performed in the DE. If communication samples are *selected*, this means they have been screened for usefulness in conceptualisation, and passed. If they are *new*, they are specially created for the conceptualisation effort.

'Mentally processed' UoD descriptions are not unlike 'descriptions of DE tasks and activities', but they are especially written to reflect an envisaged UoD or conceptual sub-domain. They should be produced by one or more expert informants who realise to a reasonable extent what the description they produce will be used for, which should make it particularly well-tuned to contextualised, purpose-oriented conceptualisation.

Phase	Phase description	Input/Output
1.3	acquire raw material	IN: existing descriptions of DE tasks and activities, communication samples from DE; OUT: selected descriptions of DE tasks and activities, new descriptions of DE tasks and activities, selected or new communication samples from DE.
2.2	capture envisaged UoD	IN: descriptions of DE tasks and activities, selected communication samples from DE; OUT: 'mentally processed' UoD descriptions.
3.2	listing and describing selected concepts	IN: 'mentally processed' UoD descriptions; OUT: lists of selected candidate terms, meaning descriptions for reference purposes, observed homonyms/synonyms across sub-domains.
4.2	selection of concepts to be 'frozen'	IN: list of selected candidate terms, meaning descriptions for reference purposes, observed homonyms/synonyms across sub-domains, lists of 'foreign' concepts to be considered; OUT: list of definite terms, approved homonyms/synonyms across sub-domains.
4.3.2a	explorative analysis of selected concepts	IN: meaning descriptions for reference purposes, meaning descriptions of 'foreign' concepts, approved homonyms/synonyms across sub-domains; OUT: explorative sketches of concept meaning.
4.3.2b	purpose-oriented analysis of selected concepts	IN: meaning descriptions for reference purposes, approved homonyms/synonyms across sub-domains; OUT: indicative sketches of concept meaning.
4.3.3	definition writing	IN: explorative sketches of concept meaning, indicative sketches of concept meaning; OUT: definitions of concepts
4.3.4	definition quality check	IN: definitions; OUT: –
5.1	general concept description quality check	IN: lists of concept descriptions; OUT: –
5.2	individual concept description quality check	IN: concept descriptions; OUT: –

Table 5.1: Artifact types as I/O description for conceptualisation phases

Lists of selected terms are simply lists of words (without meaning description) that are either considered as *candidates* for inclusion in the closed language of a medium system, or alternatively have already been selected (*definite* terms). As a third subtype, a list of words may be considered that comes from outside the current DE and is suggested for inclusion in the medium system language. Such *foreign* lists are typically (but not necessarily) standards imposed by some authority.

Meaning descriptions indeed describe the meaning of some term; however, they are not considered to be 'definitions' since they are not approved as such within the DE. *Foreign* meaning descriptions will usually have the shape of finished definitions. Meaning descriptions *for reference purposes* may or may not be full-fledged definitions; in case the are not, they are probably informal, sketch-like descriptions either of the meaning of some term or even meaning descriptions without a clear term to label them. For example, meaning descriptions for reference purposes may be used to refer to some newly encountered object or situation a definition and term for which is yet to be created.

Descriptions of homonyms/synonyms across (sub-)domains are rather specific language artefacts that provide a listing of homonyms and/or synonyms, either inside one conceptual sub-domain or covering various sub-domains. They can therefore roughly be seen as 'translation tables'. *Observed* descriptions of this kind describe homonyms and synonyms as encountered (i.e. related to lexical definition); their *approved* counterparts have obviously received official approval (i.e. are of a stipulative nature), and may even be used in relation to a set of definitions.

Sketches of concept meaning are notes that are the result of concept analysis. In line with Phase 4.3.2a and 4.3.2b, respectively, sketches can be *explorative* (i.e. lexical in nature) or *indicative* (i.e. stipulative in nature).

Definitions of concepts are meaning descriptions that have been purposely constructed and approved for use in meta-communication, and reflect a carefully constructed and evaluated view on the meaning of a term, which often is approved as such by some authority. Definitions are the most refined product that may come out of a conceptualisation process (but also the most expensive one).

Concept descriptions are combinations of terms or words and meaning descriptions (the latter including definitions). Meaning descriptions or terms by themselves are not concept descriptions.

Knowledge-Carrying Agents

Apart from language artefacts, input and output of conceptualisation processes can also be expressed in terms of concept-related knowledge that may be drawn from people. As with language artefacts, we prefer to focus on the carriers of such knowledge as can be identified as concrete entities in real domains: knowledge-carrying agents. Knowledge-carrying agents too should be positioned and described within the complex context of a DE. Questions that may rise in relation to knowledge-carrying agents include the following:

- Degree of expertise in DE –how much knowledge does a person have about the DE and/or its sub-domains?
- Degree of expertise in conceptualisation –how much expertise with explicit conceptualisation processes, methods and techniques does a person have?
- Degree of expertise in concept management –how much expertise with concept management processes, methods and techniques does a person have?
- Authority in operational matters –strategic, operational, no authority, etc.
- Authority in conceptual matters –how much power does the person have to allocate resources to conceptualisation?
- Amount of resources allocated to person for conceptualisation - how much time does she have available for conceptualisation?

Note, however, that the kind of knowledge we are concerned with in this section is not that of conceptualisation skills in one form or another. It concerns knowledge directly related to linguistic concepts as used in a DE under investigation. As discussed in Section 3.2.7, such knowledge is normally tacit.

In Table 5.2, an analysis of required types of knowledge-carrying agents is presented, again in relation to relevant phases of the reference model for conceptualisation processes. We realise it is a bit unorthodox to view agents (i.e. people) as input or output. Please note again that we are of course chiefly aiming at knowledge description, but do this with reference to the tangible carriers of such tacit knowledge: humans.

Admittedly, the contents of Table 5.2 are considerably less concrete, and perhaps less useful, than those of Table 5.1. Nevertheless, we feel that something has to be said about tacit knowledge as opposed to conceptual information carried by explicit meta-linguistic artefacts.

Once more we finish with the description of the main types of knowledge-carrying agent that can be derived from Table 5.2. Unlike the types of language artefacts, the distinction between types of knowledge-carrying agent is somewhat marginal. One might say that in the end it all boils down to ‘good knowledge of concepts and concept use, in particular in the domain’. We have attempted to be somewhat more precise.

Conceptual knowledge of de tasks and activities is the most general type of knowledge; any agent in a DE that is able to work in it should possess it in sufficient measures.

Knowledge of communicational functionality in DE is a more specialised kind of knowledge that enables an agent to estimate what the importance is of the (correct) use of some term in performing communication (and consequently, some task or activity). Perhaps more typically, it entails awareness of

Phase	Phase description	Input/Output: agents with...
1.3	acquire raw material	IN: conceptual knowledge of DE tasks and activities; OUT: –
2.2	capture envisaged UoD	IN: conceptual knowledge of DE tasks and activities; OUT: –
3.2	listing and describing selected concepts	IN: knowledge of communicational functionality in DE; OUT: –
4.2	selection of concepts to be ‘frozen’	IN: insight into usefulness of language forms in DE; OUT: –
4.3.2a	explorative analysis of selected concepts	IN: knowledge of general range of concept meaning; OUT: increased sense of some general concept meanings.
4.3.2b	purpose-oriented analysis of selected concepts	IN: knowledge of general range of concept use; OUT: increased sense of some intended concept meanings.
4.3.3	definition writing	IN: good sense of some intended concept meaning; OUT: knowledge of explicit description of concept meaning.

Table 5.2: Types of knowledge as I/O description for conceptualisation phases

what may go wrong if a term is used or understood incorrectly. In addition, such knowledge may include to the ability to suggest what *other* term might work better.

Insight into usefulness of language forms in DE This type of knowledge is strongly related to the previous type. It requires that agents can judge the functionality of a word *form* (in context of its intended use). Note therefore that this type of knowledge has little nothing to do with concept *meaning*: it relates to matters as diverse as medium system implementation or retrieval techniques, typography, or cognitive ergonomics (perception). In addition, it may concern the question of whether there is cumbersome similarity with other concept forms in the domain (or even outside the domain).

Knowledge of general range of concept meaning can be distinguished from 'conceptual knowledge of DE tasks and activities' in that it entails a relatively broad knowledge of general language. In other words, the general vocabulary of the agent in question should be considerable. We might suggest a need for having enjoyed higher education, but this is of course no set requirement nor does it provide a guarantee. A general reference to 'linguistic ability' seems more in place.

Knowledge of intended concept meanings is another more advanced variety of 'conceptual knowledge of DE tasks and activities'. It particularly concerns awareness of the specific meanings as intended in some particular task or conceptual domain. As with the previous type, it may involve better than average vocabulary knowledge. An extra refined version of this type of expertise is required if *explicit descriptions* or *definitions* are to be (co-)authored by the agent in question.

5.3.3 Quality Attributes of Conceptualisation Processes

A helpful general approach to the description of conceptualisation processes, especially in view of their analysis and evaluation against the background of their functioning as part of a larger whole, involves considering those processes in terms of *functional and non-functional qualities.* We will present a standard list of quality attributes informally known as "the -illities", and discuss how they relate to processes of explicit conceptualisation as we see them.
The quality attributes presented originated as quality attributes of software systems (ISO/IEC, 2001; ISO/IEC, 1996). Given the ICT context of our general effort, and also our 'process engineering'-related approach which views conceptualisation processes largely as non-technical systems, it seems an appropriate and useful list. We have taken the liberty of slightly rephrasing the original descriptions of the attributes to make them more process-oriented, but a system-oriented flavour remains present at times. We consider this helpful rather than a disadvantage, in view of our functional outlook on meta-communication and conceptualisation.

Functionality bears on the existence of a set of process functions and their specified properties. The functions are those that satisfy stated or implied needs.

Sub-characteristics of functionality are: suitability, accuracy, interoperability, compliance, security, and traceability.

Functionality of conceptualisation (which within the frame of our general discussion is actually linked with the functionality of concepts) is of course a central matter in our general discussion. We come particularly close to specifying 'process functions and properties' in our conceptualisation process reference model (Figure5.1); whether or not particular instances of conceptualisation processes 'satisfy stated or implied needs' is a central question in our research effort (though answering it on a yes-no level may yet prove to be beyond us).
As for the sub-characteristics mentioned: suitability and accuracy we trust speak for themselves. Interoperability and compliance are complex matters, respectively related to compatibility of goals, formats, and procedures between separate conceptualisation processes/systems, and matters of authority in (super-procedural) concept management. Security mostly concerns matters of access and mutation rights. Traceability involves the possibility to record and retrieve logs of what has happend in view of a process, and who has performed a certain action.

Efficiency is the relationship between the level of performance of a process or a subprocess and the amount of resources used, under stated conditions. Efficiency may be related to time behaviour (response time, processing time, throughput rates) and to resource behaviour (amount of resources used and duration of such use).

Efficiency of conceptualisation processes should not be confused with efficiency of communication as discussed in Section 2.2.3. It concerns the balance between resources used (time, money, personnel, technical support, but also existing meta-conceptual information, conceptualisation expertise, even appropriate metalanguage) in relation to some quantified measure of transformation of conceptualisation input into output (as described in Section 5.3.2). How the latter might be expressed is no trivial matter, and we further refrain from discussing it as it falls outside our current scope (but see Section 8.2). As for time related behaviour, this is perhaps a concern more easily understood. Placed in context of particular metacommunication situations, the time needed for some conceptualisation activity to be initiated or completed, or the amount of 'conceptualisations' that can be handled through it, is a crucial influence on whether the activity can be realistically included in some process of information system development (or evolution).

Usability bears on the effort needed for participation in the process (or one of its sub-processes) by the actors in the DE of the process. Sub-characteristics of usability are: understandability, learnability, operability, explicitness, customisability, attractivity, clarity, helpfulness and user-friendliness.

Usability is related to functionality, but is more user-oriented and focuses on agent-specific matters which in some cases are highly subjective. Properties like understandability, learnability, operability, clarity, and helpfulness are closely related to *expertise, skills, and capacities* of users concerning conceptualisation. Note that this kind of expertise must be distinguished (where possible) from domain expertise as discussed in Section 5.3.2. Explicitness is a central notion underlying our discussion: indeed we distinguish between 'explicit conceptualisation processes' (which are recognised as such and perhaps even managed) and 'implicit conceptualisation processes' (which take place on an ad-hoc basis and without much awareness that there is a language-related activity going on)[5]. Customisability of conceptualisation processes is related to how well their functionality can be 'made to fit' some situation or use. User-friendliness and attractivity are linked to attitudes of individuals and groups to how easy, how interesting, or how much fun a conceptualisation activity is. They will greatly depend on the way such activities are implemented and presented.

Maintainability bears on the effort needed to make specified modifications to the process (or sub-process). Modification may include corrections, improvements, or adaptation to changes in the environment, the requirements, and the (higher levels of the) design. Sub-characteristics of maintainability are: analysability, changeability, stability, testability, manageability, reusability.

Maintainability applies largely to process characteristics that involve management and process engineering. It is therefore a quite relevant factor in 'conceptualisation management' (not to be confused with 'concept management'). We trust the sub-characteristics speak for themselves.

Reliability is the capability of the process (or a sub-process) to maintain its level of performance under stated conditions for a stated period of time. The mean time to failure metric is often used to assess reliability of processes. The reliability can be determined through defining the level of protection against failures and the necessary measures for recovery from failures. Sub-characteristics of reliability are: maturity, fault tolerance, recoverability, availability and degradability.

Admittedly, this property is so much a software system property that it is somewhat hard to apply it to conceptualisation processes. 'Failure' of such a process would probably mean a breakdown of its capacity to 'produce' acceptable output due to matters other than its basic design. In other words, reliability is mostly a matter of implementation. However, the more conceptualisation becomes embedded in an ICT-supported environment, the more reliability gains relevance.

Portability is the ability of the process (or one of its sub-processes) to be transferred from one environment to another. Sub-characteristics of portability are: adaptability, installability, conformance, replaceability.

Even more than reliability, portability is a property that is chiefly related to computationally implemented systems. We refer to the discussion of reliability above.

[5] A note of caution: please be aware of the treacherous difference between 'explicit conceptualisation processes' and 'processes of explicit conceptualisation'.

5.4 Conceptualisation Techniques

We now present an overview and discussion of existing techniques for explicit conceptualisation. Through this, we first of all hope to provide a reasonable insight into the concrete activities that may constitute 'conceptualisation'. This adds an important component to the answer to Q1.5: "how are processes and environments for freezing language related to information system development?". In addition, we consider it important that concrete conceptualisation techniques are included in our descriptive framework, so that it is not limited to abstract organisational and communicational structures and phenomena, however important these may be.

Throughout the literature consulted (see Appendix A), the terms 'method' and 'technique' are often confusingly similar in meaning. We have opted to use 'technique' as a common denominator, but arguably many of the techniques discussed here could as well have been named 'methods', and even 'approaches' or 'instruments'. Broadly speaking, what we call a 'conceptualisation technique' in this chapter (and in appendices A and B) can be *any concrete means to assist some phase of conceptualisation*. Importantly, as in previous chapters we do not include any discussion of representational paradigms and (meta-)languages, which in the literature are often also viewed as 'techniques' and are quite regularly tied up with more process-related approaches. Paradigms of representation of language and meaning are highly important in the general 'language and meaning game', but they are very deliberately kept out of our current process-oriented scope.

Due to the size of the area covered in the overview, it is quite impossible to be comprehensive in our coverage of individual conceptualisation techniques, and of variations and combinations thereof. We therefore present what we consider to be the chief categories of technique encountered, and provide examples and references without claiming to be complete or to refer to the earliest origins of particular techniques. The main result is a list of the chief conceptualisation techniques that we encountered in our literature study (Section 5.4.1). A description of each category of technique can be found in Appendix B. Some techniques may be used in various phases, as will be indicated. Some (sub-)phases are supported by quite a number of techniques whereas for other phases, supporting techniques seem underdeveloped.

5.4.1 Overview of Conceptualisation Techniques

The techniques presented in this section do not cover all the activities in the CPRM, nor all the output types presented in tables 5.1 and 5.2. The techniques are not like a comprehensive set activities covering all of conceptualisation activity; they are mostly dedicated, focused instruments that may help perform *parts of* or *aspects of* the general conceptualisation process. They have inspired the CPRM, but the CPRM is more than a framework for positioning the techniques, and is intended to be comprehensive.

We presented basic, relatively 'elementary' techniques: in most applications, various techniques will have to be combined in some way or another. For example, within one MSLC we may observe adaptation of existing concepts combined with an automated questionnaire-like technique combined with dialog construction combined with template filling, including feedback by means of ex post, intervening, mediated

meta-communication (i.e. checking by using).
In Table 5.2, we present an overview of the techniques discussed in this chapter. The overview is concise and is aimed at comparison of basic properties rather than on representation of complete and detailed issues.
The items covered in Table 5.2 are the following:

Ph 1-5: conceptualisation phases corresponding to those in the CPRM; "x" = 'used in this phase'; "xx" = 'particularly relevant to this phase'

Auto: possibility for automated support; "x" = 'some support', "xx" = "considerable support"

Eff: efficiency; low-medium-high; "dep" = 'depending on situational factors', in other words: 'it is impossible to generalise about this property here'

Usab: usability; dito

Main: maintainability; dito

Rel: reliability; dito

Port: portability; dito

Strong point: speaks for itself

Weak point: dito

In Table 5.2, some rudimentary indications are included that should help make it possible to *evaluate* conceptualisation techniques with respect to how they might fit particular conceptualisation situations. We do this by referring to general properties of processes presented in Section 5.3.3: efficiency, usability, maintainability, reliability, and portability. Note that 'functionality' is not included explicitly, but is implied by the occurrence of a technique under some phase. More detailed aspects of the functionality of the techniques are discussed in Appendix B.
The process view on techniques as entails the use of 'process quality attributes' may seem a bit far-fetched. However, we strongly believe our approach is justifiable because forcing ourselves to think about conceptualisation techniques as parts of conceptualisation processes helped us greatly in developing our focus on the matter, which we believe is innovative. Admittedly, our estimations of quality properties as related to conceptualisation techniques are tentative and at times incomplete. They are based on our own insight and experience rather than empirical verification. We have found no references to process evaluation in relation to the techniques discussed, nor did we have we the time and means to gather and analyse data on the subject (but see Section 8.2).

5.4.2 Conceptualisation Techniques and Conceptualisation Phases

We finish with a discussion and some general observations concerning conceptualisation techniques per conceptualisation phase. We are particularly interested in

	Ph1	Ph2	Ph3	Ph4	Ph5	Auto	Eff	Usab	Main	Rel	Port	Strong point	Weak point
Selecting Artefacts	xx					x	med	high				comprehensive source	value for conc.may be unclear
Creating Texts	xx	x		xx	x		med	high				intuitive, basic skill	relatively unstructured
Creating Diagrams	x	x		xx	x		high	high				powerful and efficient	somewhat specialised
Describing Cases	xx	x					med	high	high	med	high	intuitive, basic skill	limited use in comm. study
Writing Scenarios	xx	xx					med	med	med	med	med	may be comm.-oriented	
Brainstorming	xx			xx			high	med				creative, free, easy, fast	individualist; possib. mediation
Keeping Diaries	x	x			x		low	med				covers incidents	patience needed
Conducting Interviews	xx	x		x	x		high	med				powerful and flexible	exec. harder than most expect
Sending Out Surveys	x			x	x	xx	med	med	high	high	high	large population coverage	heavily pre-structured; inflex.
Making Observations	xx	x			x	x	low	low	high	med	high	direct capturing of comm.	expensive; postprocessing
Introspection		xx		xx	x		high	med				direct and efficient	individualistic, unguided
Think Aloud Protocols		x			x		med	low				detailed, relevant results	hard to execute, limited use
Listing Exs./Reading Facts		xx					low	low				comprehensive, complete	requires much expertise
Telephone Metaphor	x	xx			x	x	high	high				easy and effective	
Constructing Dialogues	x	xx			x		med	high				captures communication	artificial setups may be req.
Seeking Conc. Differences	x	xx		xx	xx		high	med	dep	dep	dep	to-the-point comparison	few guidelines available
Grammatical Analysis			xx	x	x	xx	low	med	med	low	med	concrete analytical frame	unfit for bulk; specialist work
Identifying Semantic Types			xx	x	x	x	low	med	med	low	med	very direct link with meaning	many, many category systems
Procurement of Concepts			xx	x		x	med	low	dep	dep	dep	quick, standard-compliant	customisation may be a pain
Architecture Viewpoint	x	x	xx	x	x	x	low	low	low	low	low	less complexity, more struct.	difficult and expensive
Homonyms and Synonyms			x	xx	x	x	low	med	high	high	high	uniform and basic	complex and much work
Informal Definitions				xx	x	x	med	med	med	med	low	intuitive and flexible	vague, not machine-readable
Formal Definitions				xx	x	xx	low	low	med	high	med	pricese, complete, computational	expensive, cryptic, too detailed
Definition Diagrams				xx	x	xx	med	med	low	med	low	good layout, many relations shown	requires advanced representation
Mapping Linear/Diagram		x		xx	x	x	med	low	dep	dep	dep	improved meta-communication	exec. harder than most expect
Template Filling	x		x	xx	x	xx	high	high	med	med	med	efficient and easily auto-supported	requires fixed structure; inflexible
Adapting Existing Conc.			x	xx			med	med	dep	dep	dep	good fit with target domain	laborious and often difficult
Reaching Consensus	x	x	xx	xx	x	x	med	high	high	high	high	uniform and basic	may be expensive and polarising
Corpus Analysis	x	x		xx	xx	xx	high	med	high	high	med	covers broad usage	may be technically difficult

Figure 5.2: Techniques for Explicit Conceptualisation

considering possibilities for ex post application of techniques, since we generally observe a lack thereof in current practice. In addition, assuming that the existence of concrete, well-developed techniques can be taken as an indication of the maturity of practices concerning some conceptualisation phase, we can conclude that there appears to be a particular lack of maturity of Phase 3, and possibly also Phase 2. Consequences and remedies of this immaturity are also briefly discussed.

Phase 1

This phase includes a broad scala of techniques. It is interesting to see how many techniques for gathering raw material there are, and how they are alike across various fields of application. Phase 1 is mostly oriented on ex ante approaches. The most promising extension of techniques to an ex post approach is to (automatically) collect/record communication samples, possibly including meta-communication. This data may of course be subject to ex ante analysis, but it could also be referred to in ex post meta-communication: "looking at what others have done in similar circumstances". Such an approach seems related to the use of 'translation memory' (Section A.6); this notion might tentatively be extended to 'communication memory' and perhaps also 'conceptualisation memory'.

Phase 2

Phase 2 also covers diverse techniques, but they are fewer and more specialised – which is no surprise since the entire phase is more specialised: it is more explicitly communication-oriented than Phase 1. Phase 2 seems not so much 'underdeveloped' in terms of techniques as to suffer from the fact that it is very hard to *predict* what good concepts might be for a communication situation that yet has to come into existence.

Experimental approaches may be a way out here. Thus, as observation of 'real/current' operational communication is mostly rooted in Phase 1, observation of 'pre-operational', experimental concept use belongs to Phase 2. In essence, such experimentation will revolve around the ability to either *imagine* or *simulate* concept use. Again, the predominant approach here is ex ante. Yet ex post 'pre-operational' approaches seem possible in principle; they will probably depend on capability of a system to not only be customisable in terms of concepts, but also 'experimentally customisable'. In other words, it would be helpful if concept-related adaption of the *operational* medium system could be tried out before being 'fixed'. Such a trial structure might be an important part of an *evolutionary conceptualisation setup*. After all, evolution almost inherently includes experimentation, linked to 'survival of the object of experimentation'. In context of conceptualisation, it seems preferable to focus on enabling small-scale experimentation concerning only a particular concept use in a particular communication situation, since this reduces complexity, increases nuance, makes it possible to pinpoint specific problems (perhaps isolate excesses), and decreases the scope of intervening meta-communication (and thus, hopefully, of the implementation effort following from it).

Phase 3

Phase 3 techniques seem underdeveloped in particular with respect to directly communication-oriented techniques. With the exception of the interrelated techniques especially oriented on *identification* of candidate concepts (i.e. grammatical analysis, semantic typing, and to some degree coping with homonyms/synonyms), this phase involves attempts to acquire and apply explicit knowledge concerning the 'how' of concept selection. In this respect, experimentation as discussed above in relation with Phase 2 is the trial-and-error counterpart of the more predictive approach indicated here. Yet predicting which concepts will ultimately work best in some communication situation remains very hard, especially if various factors clash. It would therefore be interesting to try and have it both ways: enable Phase 2-like trials/experimentation but stimulate an explicit, differentiated focus in the analysis of (various aspects of) concept functionality. Possibly, known priorities in some communication situation or DE could be brought into the experimental equation, restricting the range of its outcome where possible.

Phase 4

The central observation concerning Phase 4 conceptualisation remains that it should be minimalised (and this holds especially for Phase 4.3). Meaning in particular can be described in so many ways and with emphasis on so many aspects that careful pre-selection of means (meta-languages but also procedures, techniques for editing and representation, and various other supporting mechanisms) is called for. Central here are still the questions *whether* and *how* concepts are to be described as part of meta-communication. The framework developed in this thesis may help here, but we are not able to devise standard answers: until certain generalised use patterns and heuristics emerge (and so far we are mostly in the dark in this respect; see Section 8.2), every situation will have to be analysed and evaluated individually.

Design of Phase 4 conceptualisation processes, if it occurs at all, is currently mostly performed within an ex-ante context. Ex-post extensions are virtually non-existent. Assuming that ex-post concept description will at times be useful or even necessary, and given the relatively low efficiency and usability of most concept description techniques, it seems worthwhile to develop appropriate meta-communicational description patterns and templates for operational, on-the-fly meta-communication. Importantly, not only language but also *meta-language* should somehow be tuned to language and communication practices that people are familiar with (note once more that meta-communication as such is not alien to common language use; see Sections 2.2 and 3.4). Conceptualisation may thus perhaps be 'brought to life' (within some evolutionary setup) by increasing efficiency and usability along the same lines as attempted with general information systems development. It has been suggested that intelligent computational approaches may help here, but the matters at hand will need to be well understood first. 'Manually' performed, dynamic, 'living' conceptualisation may have a role to play in the bootstrapping of a more evolutionary form of ICT.

Phase 5

Ex post concept checking is closely related to the notion of 'checking by using', and therefore mostly intertwined with regular activities belonging to phases 2-4. In contrast, systematic concept checking can more typically be seen as a specific ex ante activity apart from other phases, even though techniques used also occur across various such phases.
This concludes our discussion of processes and activities of explicit conceptualisation. We will return to our findings in Chapter 6, where they will be placed in a larger framework that also includes environmental factors as have been discussed in Chapter 4.

5.5 Summary

This chapter concerned processes for *explicit conceptualisation*: the specification of concepts. On the basis of a literature study of approaches and techniques for conceptualisation that occur across a number of fields (Appendix A), we concluded that though the importance of linguistic matters is generally acknowledged, this interest is mostly seen as related to the representations and artefacts of language description, not to the phased processes that underly the creation of these artefacts. In addition, the way various environmental factors (in relation to more general processes of meta-communication) influence the requirements for conceptualisation processes is not taken into account. The literature study and the conclusions drawn from it, complemented by Section 5.4 and Appendix B, constitute our main answer to Q1.5.
However, the most important contribution of Chapter 5 to the thesis is that it includes the chief answer to the central research question Q1.1. It provides a reference model and various other conceptual devices that together enable us to *identify, describe, and analyse conceptualisation processes*, in relation to observable activities and artefacts in organisations. Central to the analytic framework presented is a description of phases of conceptualisation processes that can be usefully distinguished:

1. acquiring raw material
2. capturing an envisaged UoD
3. selection of relevant concepts
4. naming and defining selected concepts
5. performing quality checks on previously conceptualised concepts

These phases and their sub-phases were elaborately discussed. They reflect significantly different activities in conceptualisation; we claim it is possible to categorise all main conceptualisation activities in organisations according to the model. The phased reference model was then used as a main ingredient of a *typology of conceptualisation processes*, complemented with a description of input and output of the main process phases in terms of language artefacts and knowledge types. Also, six main process quality attributes were discussed in relation to the description and evaluation of conceptualisation processes: functionality, efficiency, usability, maintainability, reliability, and portability. Finally, we listed 29 'conceptualisation techniques', indicating for each technique in which particular conceptualisation phases it can be usefully deployed (in relation to the process properties mentioned). The conceptualisation techniques distinguished for a large part emerged from our literature study of conceptualisation practices; individual techniques are described in Appendix B.

Chapter 6

An Architecture Viewpoint for Conceptualisation

In the previous chapters, we have discussed at considerable length conceptualisation processes and many related aspects of the environments they take place in. Theoretical considerations as well as observations concerning conceptualisation practices have been presented, but in a somewhat fragmented fashion. Yet we have also promised to present our main findings as a coherent analytical framework, and this is what we set out to do now. The chapter relates primarily to a research question (formulated as a design goal) that has so far received little explicit attention:

Q2.1 Develop a conceptual framework for description and analysis of situation-specific aspects of frozen language in the ICT domain.

However, the chapter also provides a necessary link between the theoretical part (Q1.1-5) and the more applied part (Q2.1-2). In addition, it provides most of the answer to a theoretical question that has not yet been answered in full: Q1.3 ("How do processes and environments for freezing language relate to and influence each other?").

The framework presented takes the shape of an *Architecture Viewpoint*. What is meant by this will be discussed in Section 6.1. Next, we line up our main findings, at times using alternative forms of presentation than in previous chapters (here mostly diagrams) that enable us to make visible a network of descriptive concepts: a language for speaking about conceptualisation, as it were. We also add some concepts needed to tie together and complete our descriptive framework. Thus, we answer Q2.1, and also Q1.3. This is Section 6.2.

We then address our final research question (again formulated as a design goal):

Q2.2 On the basis of the conceptual framework, develop an analytical method enabling the analysis and evaluation of concrete processes and environments of freezing language in real ICT situations

In Section 6.3, we operationalise the framework, complementing it with some *models* and *templates* that may be used as practical tools in analysis. We also provide some methodological guidelines. Thus we partly answer Q2.2, that is to say we provide chiefly *analytical* instruments. The *design* aspects for which the framework might also be used lie outside our immediate focus of the thesis. How to wield the framework successfully in context of real management will remain an open question; we will briefly return to it in Section 8.2.

6.1 Architecture Viewpoints

A particularly important notion in this chapter is that of a 'viewpoint' (see Figure 1.1 in Section 1.3.4). It is closely related to that of 'view'. In (Sherlund et al., 2000, pp.4-5) they are defined as follows:

> An architecture description is organised into one or more constituents called (architectural) *views*. Each view addresses one or more of the concerns of the system stakeholders. The term *view* is used to refer to the expression of a system's architecture with respect to a particular viewpoint.
>
> [...] A viewpoint establishes conventions by which a view is created, depicted and analysed. In this way, a view conforms to a viewpoint. The viewpoint determines languages (including notations, model, or product types) to be used to describe the view, and any associated modelling methods or analysis techniques to be applied to these representations of the view. These languages and techniques are used to yield results relevant to the concerns addressed by the viewpoint.
>
> An architectural description selects one or more viewpoints for use. The selection of viewpoints typically will be based on consideration of the stakeholders to whom the architecture description is addressed and their concerns. A viewpoint definition may originate with an architecture description, or it may have been defined elsewhere. A viewpoint that is defined elsewhere is referred to [...] as a *library viewpoint*.

So a viewpoint is a descriptive and analytical instrument that, if applied to a case, results in a view on that case that conforms to the viewpoint. We set out to create an *architecture viewpoint focusing on conceptualisation*. Since we do this independent of an actual architecture description, we develop a library viewpoint. In Chapter 7, we will present a conceptualisation view of a real organisational situation, conforming to our viewpoint.

As mentioned, the number of different views and viewpoints that might be created is endless in principle. A good example of a systematically derived collection of viewpoints for information system architectures is that of (Zachman, 1987). For more on architecture viewpoints, see (Hilliard, 2001; van den Heuvel and Proper, 2002).

Architecture viewpoints as described in IEEE recommended practice 1471 should included the following components (Sherlund et al., 2000, pp.9-10):

- Viewpoint name
- Stakeholders to be addressed by the viewpoint
- The concerns to be addressed by the viewpoint

- The language[1], modelling techniques, or analytical methods to be used in constructing a view based upon the viewpoint

- The source, for a library viewpoint (the source could include author, date, or reference to other documents, as determined by the using organisation)

We include all these components except for the last, which we cannot foresee (apart from a reference to this thesis). The viewpoint name is simply "Conceptualisation Viewpoint".

6.2 An Analytical Framework for Conceptualisation

In this section we provide a general overview of concepts that can be used to describe and analyse conceptualisation processes and their environment, largely based on previous chapters. We might call them the "pieces of the conceptualisation puzzle". We refrain from extensive justification of the inclusion of individual conceptual elements, assuming that their introduction and discussion in previous chapters validates them as relevant to the description of conceptualisation processes and their environments.

First, some concepts (*stakeholders* and *concerns*) are newly introduced in order to complete the descriptive language (in view of its use as part of an architecture viewpoint). These too are mostly derived from previous discussion, in which they have played a role but have not been made explicit as descriptive concepts.

The general rationale behind the setup of the framework is as follows. If something needs to be done (in our case, operational communication), tools (language, concepts) are needed to support the process. These tools need to match the requirements of the operational process, with respect to the primary functionality (specific concepts needed in the DE) as well as more general properties of functionality (types of language use) and the needs and restrictions of the conceptualisation environment (DE). In order to align the many factors involved, it is necessary to take into account the requirements as determined by the people (and roles) related with it (stakeholders). The activity of building or fine-tuning the tools (as well as related instructions for use) can be considered as a separate communication process, with its own sub-types and properties (meta-communication typology). On a more operational, practical level, this 'tool building process' (conceptualisation) has its own sub-processes and phases (conceptualisation process reference model), with their own input and output (artefacts, knowledge) and a number of techniques that may support it. In Figure 6.1, the relations between the 'pieces of the puzzle' are shown.

Stakeholders and Concerns

Stakeholders and related concerns have not been explicitly discussed so far, even though they have been implicitly mentioned throughout the text. Identifying them is crucial for determining what the needs, demands, and possibilities in a domain

[1]Note that architecture is also a field that heavily involves explicit conceptualisation.

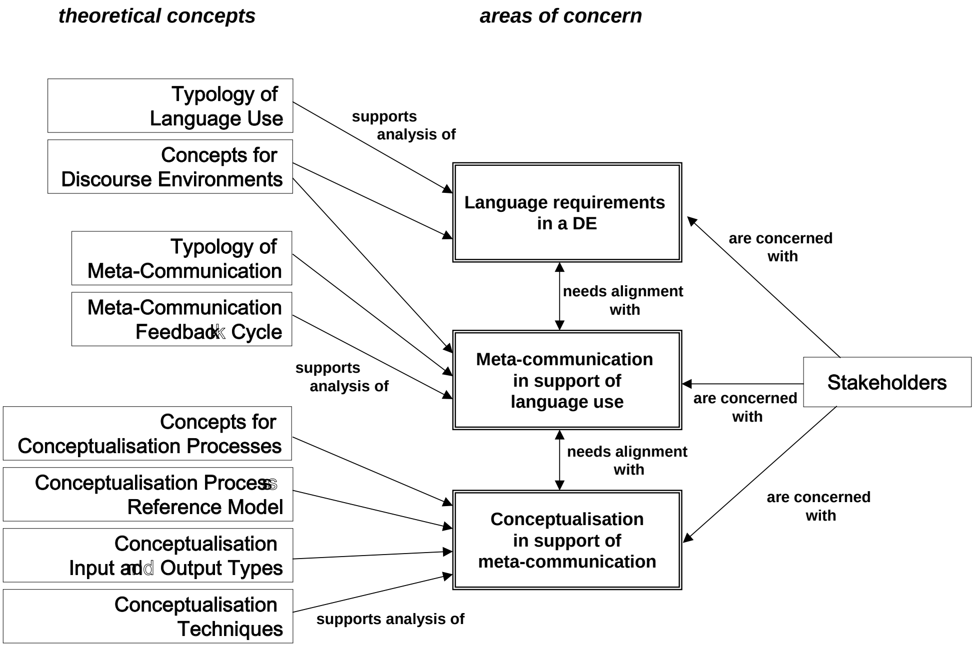

Figure 6.1: Relations between the main sets of theoretical concepts in the framework

(MSLC) are. We introduce five specialised stakeholders for the conceptualisation viewpoint:

- Concept informants: those who take part in discussions about language/terminology
- Concept authors: those who are authors of certain language specifying artefacts
- Concept authority: those who decide upon official terminology
- Concept managers: those who are responsible for concept management (including conceptualisation processes)
- Conceptualisation facilitators: those who are a facilitators for the conceptualisation process

In Figure 6.2, these conceptualisation stakeholders are related to the main 'general system stakeholders'. This gives a rough impression as to which conceptualisation roles may typically be mapped to which more general roles in the ICT game. The mapping is not mandatory, nor are alternative mappings excluded per definition.
The various stakeholders have different concerns related to conceptualisation. The seven main areas of concern listed below are mostly meant to make clear what distinguishes the conceptualisation viewpoint from other viewpoints, and make it possible to position it in relation to them. The stakeholders most directly related to them are added in italics.

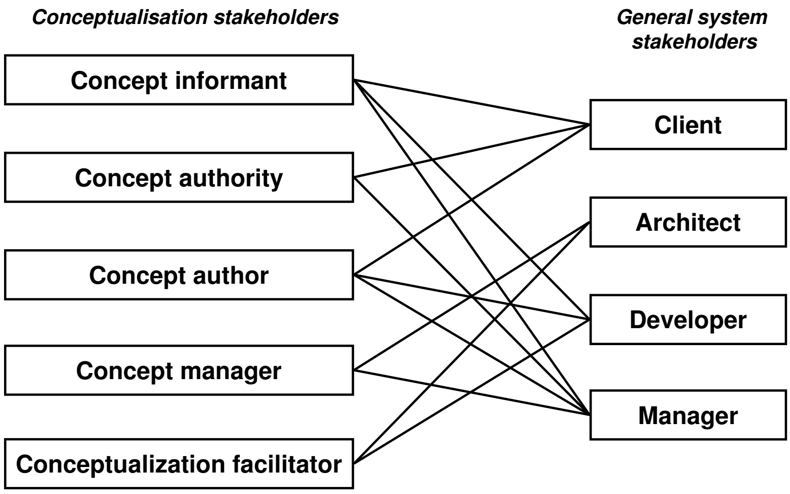

Figure 6.2: Conceptualisation Stakeholders and General System Stakeholders

- Organisation (*managers; concept managers*)
- Communication (*all stakeholders in the viewpoint*)
- Operational Communication (*concept informants; clients*[2]*, managers*)
- Linguistic Meta-Communication (*concept informants, concept authority, concept managers; developers*)
- Conceptualisation (*all stakeholders in the viewpoint*)
- Implementation of Concept Representations (*concept authors, concept managers; developers*)
- Information System Development (*concept managers; architects, developers, managers*)

Note that some areas of concern are enveloped by others. This is expressed in Figure 6.3. The figure mainly depicts areas of concern in a manner not unlike a Venn-diagram. Apart from overlap between areas of concern, the figure informally shows the present distance between main linguistic concerns (centred in communication) and ICT concerns (centred in information system development). Conceptualisation in MSLCs involves both aspects, and thus takes place in a field of tension between concerns of communication and of technology. This is represented by the two black dots and the arrows protruding from them.
Basic concerns in viewpoints can also be usefully expressed in the form of concrete questions that the resulting view should be able to answer (Hilliard, 2001). We formulated some central questions below.

[2]'Clients' here refers to 'clients of system development': simply put, 'information system users'.

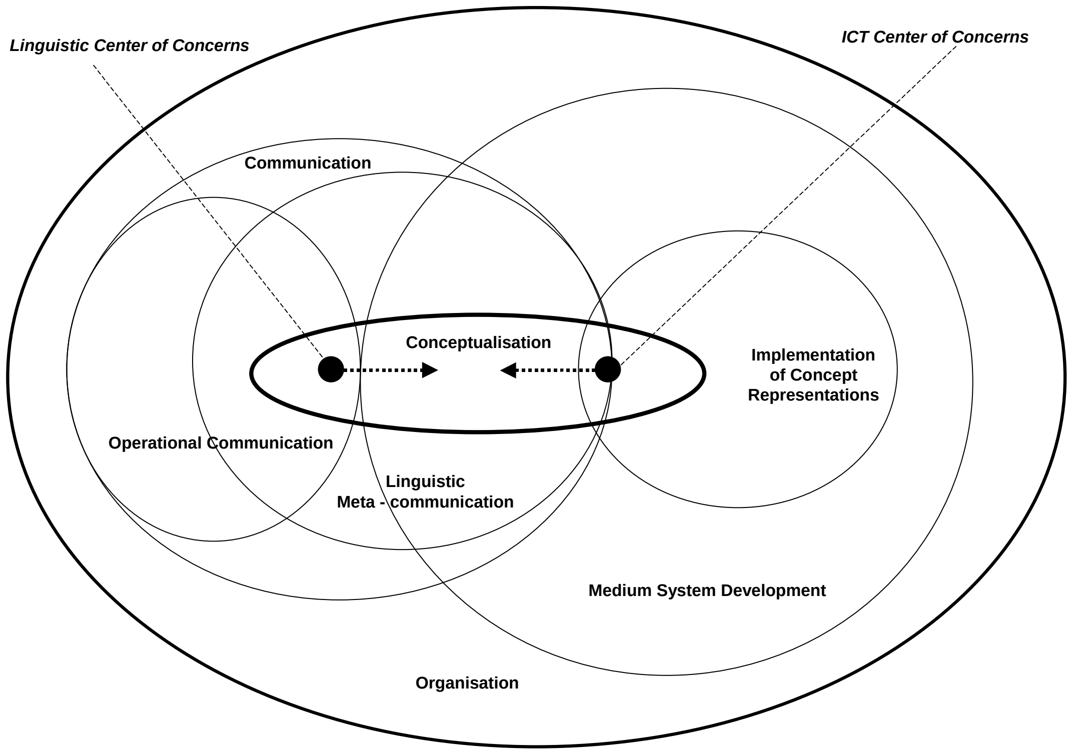

Figure 6.3: Overlap in Conceptualisation-Related Areas of Concern

- What forms of (meta-)communication play a role in operations in the MSLC?
- What forms of (meta-)communication play a role in conceptualisation in the MSLC?
- Which different DEs and languages need to be considered in relation to the use of the medium system?
- Which different DEs and languages need to be considered in relation to the design and documentation of the medium system?
- How and by who are concepts that are to be built into a mediums system to be selected?
- How and by who are concepts that are to be built into a mediums system to be described?
- How is conceptualisation to be organised?
- How is conceptualisation to be effectively supported?

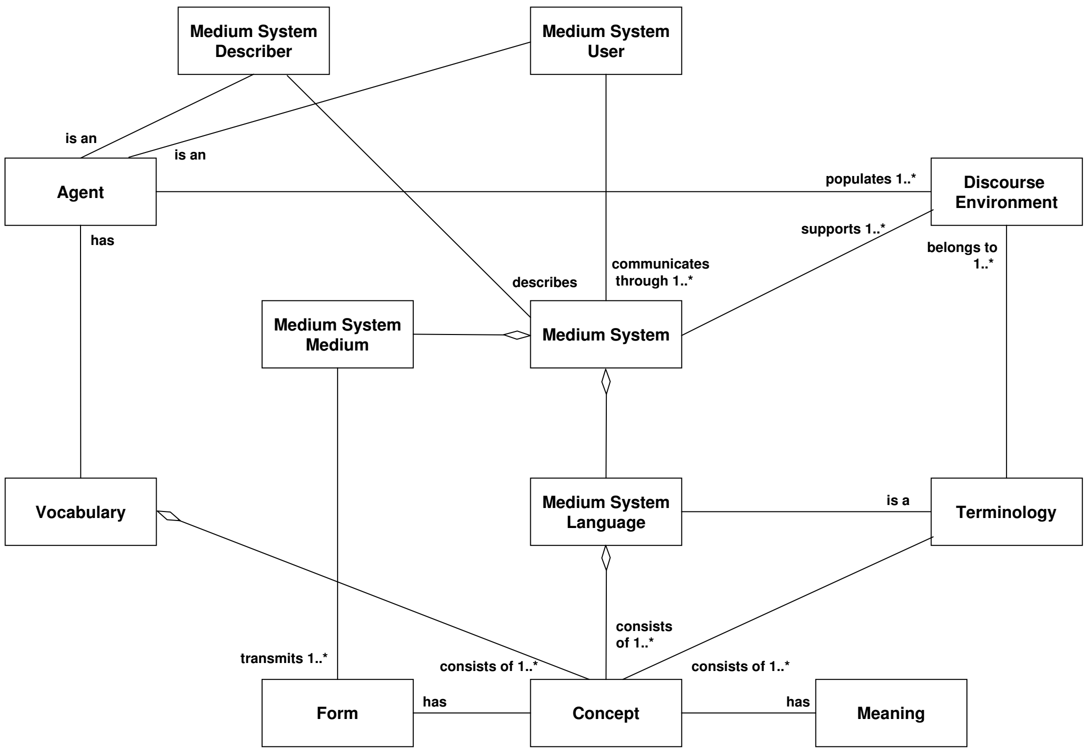

Figure 6.4: Basic Conceptual Framework for Discourse Environments

Conceptual Framework for Discourse Environments

In Figure 6.4, a simplified representation of the basic conceptual framework (Table 2.1, p21, and Table 4.2, p83) is presented. This framework includes the key concepts needed for the description of discourse environments. The simplification renders a set of concepts that is fit to describe the essential aspects of discourse environments in a reasonably transparant manner, leaving out some technicalities of theory and formal notation. Most notably, we have simplified the framework in terms of language concepts: in particular, we leave out the notions of “speech act” and “utterance”. These notions are vital for complete theoretical understanding, but are less relevant in a descriptive architectural effort.

Note that medium systems as presented in the framework can in principle be *any* medium system, including “the system carrying verbal language by means of sound waves”; only *some* medium systems are mechanised or computerised, and only *some* have an associated explicit language specification. However, we do, emphatically, assume a focus on computerised medium systems.

Importantly, “communication through a medium system” refers strictly to *operational communication in a particular discourse environment*. Thus, “description of a medium system” is *not* viewed as relevant communication in that DE as such. It is a pointer that –via an agent– leads to *another* DE in which operational communication concerns medium system development. Consequently, a medium system that is

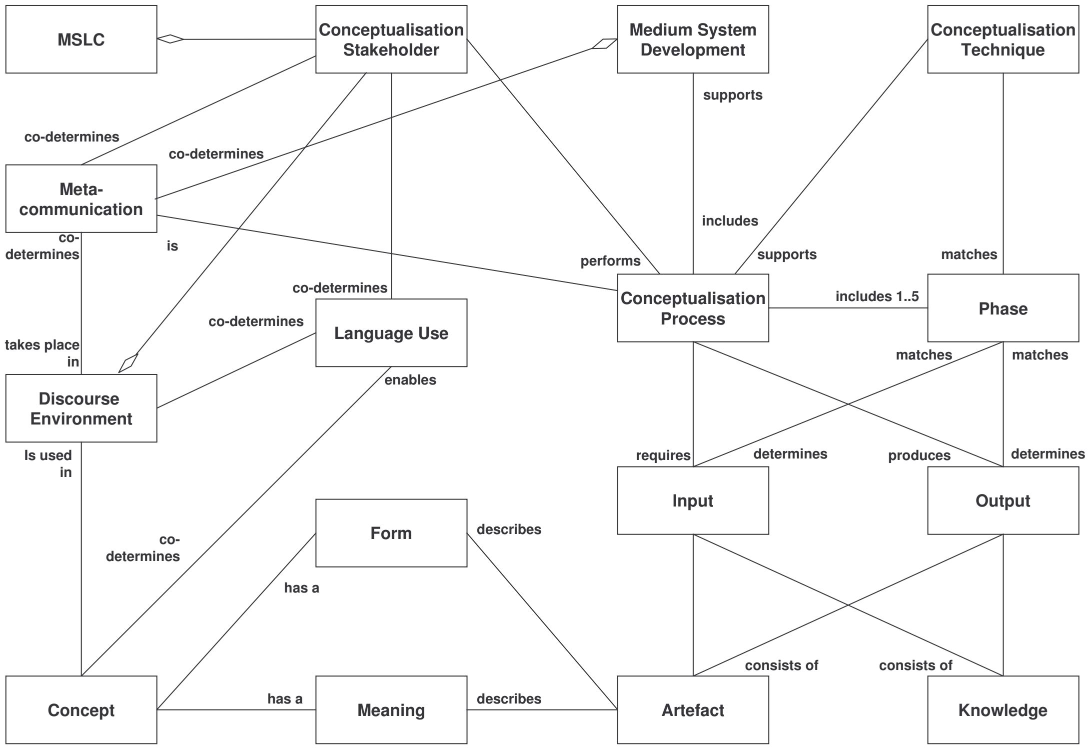

Figure 6.5: Conceptual Framework for Conceptualisation Processes

used in minimally one DE is almost inherently associated with another DE in which it is created.

Typology of Language Use

In Section 4.2, a number of different manners of distinguishing and typifying languages have been introduced. They were summarised in Figure 4.1 (p94). The typology is useful as an analytical and descriptive aid in determining what a concept will be used for, making it better possible to tune conceptualisation to conceptual requirements.

Conceptualisation Process Reference Model, I/O Types, and Techniques

In Section 5.3 (Figure 5.1, p129), a reference model was presented of the conceptualisation phases that can be distinguished. The model was complemented in Section 5.3.2 by two typical input-output descriptions linked to conceptualisation phases: one concerning artefacts (Table 5.1, p131), one concerning tacit knowledge (Table 5.2, p134). In Section 5.4, summarised in Figure 5.2 (p140), the phases were linked to conceptualisation techniques that might be used to implement and support them.

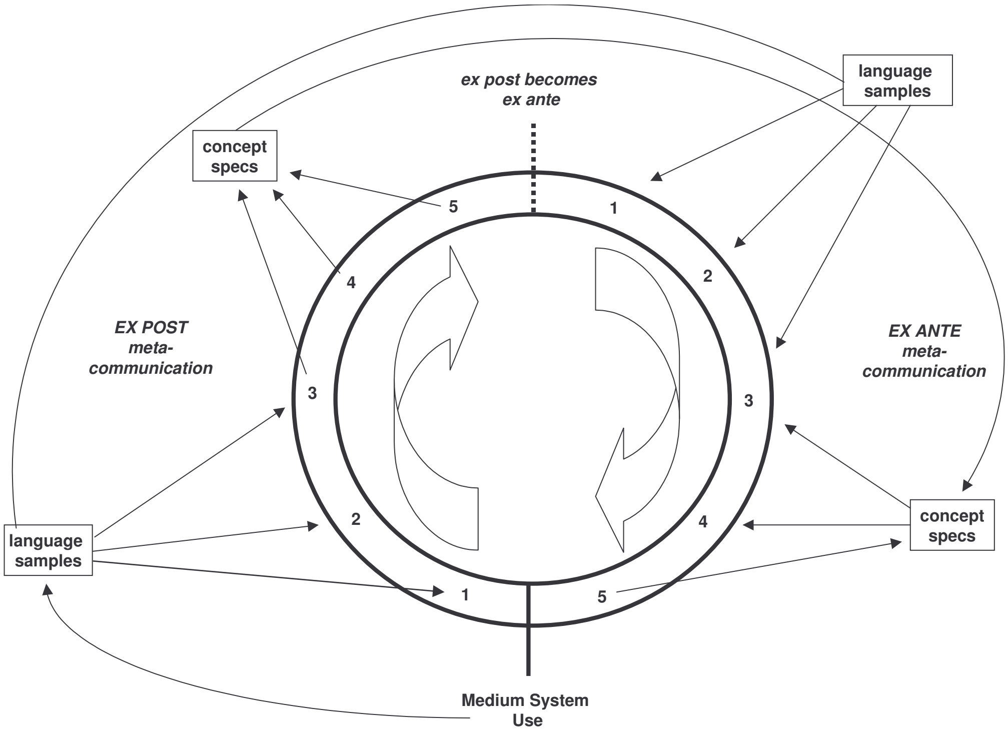

Figure 6.6: Feedback Cycle for Meta-communication

Conceptual Framework for Conceptualisation Processes

In Figure 6.5, a conceptual framework is presented that (if seen as the core terminology of a descriptive language) enables us to describe, in a static fashion, conceptualisation processes and their main features. Unlike the basic conceptual framework for the description of discourse environments, no formalisation underlies the concepts in Figure 6.5. Instead, the concepts it includes are derived from the various typologies, the reference model, and some other descriptive devices discussed in previous chapters. The framework is meant to put all these in one, coherent perspective. It also shows some overlap with the framework for discourse environments, thus relating conceptualisation processes to DEs.

Typology of Meta-Communication

Finally, we fit in the typology of meta-communication as previously presented in Section 4.3.3. Its relevance lies in the fact that which type of meta-communication is involved may have great impact on requirements and possibilities for its organisation and support, and also for any conceptualisation that might be related to it. Meta-communication, directly determining the requirements for conceptualisation, is generally driven by six factors (see Section 4.3.2). How the responsibilities related to these factors are distributed across an organisation determines what the priorities are in meta-communication. This in turn influences the requirements for each conceptualisation process. In a properly functioning conceptualisation setup, it should be decided on these grounds which concerns and stakeholders should have the most influence on the conceptualisation process.

Conceptualisation Feedback Cycle

In Figure 6.6, dubbed the *Conceptualisation Feedback Cycle*, it is shown how ex ante and ex post meta-communication (two halves of the cycle) relate to the five conceptualisation phases (numbers 1-5), and to operational communication ("medium system use", and a solid line at the cycle's 'six o'clock'). The transition point from ex post to ex ante meta-communication is indicated with a dotted line at the cycle's '12 o'clock'. The general temporal 'direction' is indicated by the arrows in the centre. Some references to crucial input/output artefacts for conceptualisation processes (language samples, concept specifications) are also included, in relation to particular conceptualisation phases and medium system use, globally representing how particular conceptualisation phases may provide input for other phases (arrows). Information derived from system use is similarly expressed. Importantly, if both ex post and ex ante meta-communication takes place, the processes involved may, as it were, 'feed off each other'. The cycle thus represents the ongoing creation and adaptation (and possibly cancellation) of concepts, which may find its ultimate realisation in an evolutionary conceptualisation environment, possibly integrated with an evolutionary information system.

6.3 A Method for Analysing Conceptualisation Processes in Context

We have finally reached the point at which we can present an applicable, methodic approach to analysis of conceptualisation processes and their environment (effectively answering research question 2.2). We integrate the presentation of various *diagrams* and *templates* with some *methodological guidelines* as to how they might be applied: in which order they may occur, and how intermediary results may be interrelated. The analytic instruments (diagrams, templates) suggested make use of and coherently relate the main theoretical concepts that have emerged from Chapters 2 to 5, and that have been summarised in Section 6.2. For effective use and full understanding of the conceptual instrumentarium, at the very least a reasonable grasp is required of these 'pieces of the conceptualisation puzzle'.

We organise our method along the lines of four of the seven basic concerns distinguished in Section 6.2: operational communication, linguistic meta-communication, conceptualisation, and implementation of concept representations. Organisation, general communication, and information system development we consider to be related concerns that are nevertheless outside our immediate scope and are covered extensively by other, existing architecture viewpoints and perspectives.

We present our method as existing of five consecutive stages, as depicted in Figure 6.7. Though our approach has no strict ordering, there are some expected dependencies between intermediary results. The most straightforward, concrete types of

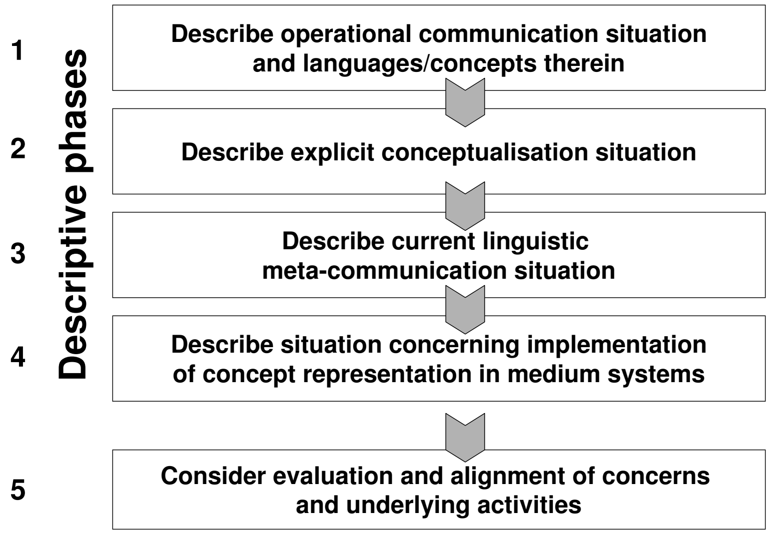

Figure 6.7: Stages in the Method for Conceptualisation Process Analysis

phenomenon (objects and activities) that may be observed in the domain analysed are the following:

- Agents and artefacts involved in *operational communication*
- Artefacts and techniques used in explicit *conceptualisation*

Since analysis of conceptualisation processes will mostly explore a new and relatively unknown perspective, it seems best to start out with what can be concretely observed in the domain onder consideration (bottom up approach). Hence, we recommend analysts to first focus on concerns of operational communication (stage 1) and explicit conceptualisation (stage 2). However, this should not be seen as a mandatory course of action; analysis may well develop incrementally and involve all concerns at an early point.
Linguistic meta-communication (stage 3) is a primarily analytical, abstract concern, that can only be usefully considered after more concrete aspects have been charted, and may then add greatly to the fundamental understanding of how languages and concepts are communicated about. Implementation of concept representations (stage 4) is part of information systems development. Our method touches upon this concern and includes some instruments concerning it, but these mostly play a role in evaluation and alignment of the concerns, not in the basic analysis.
Stage 4 involves a different sort of analysis than the previous four, not aiming at description and analysis of some existing situation ("Ist") but of evaluation thereof and possibly the creation of a future situation that might be worked towards ("Soll"). While it is not our ambition to make full-scale evaluation and alignment part of our current approach, we include some preliminary guidelines and heuristics that may help architects in evaluating and adjusting conceptualisation situations.
Below, we will present a textual description of the proposed method, complemented with diagrams and templates that should aid the obtainment and representation of

results. Some of the more detailed descriptive instruments may be relevant only to later stages in the approach, and may need input based on those later stages (implying iteration of analytical stages). These items are marked in the templates by being shown in italics. The diagrams are loosely based on UML Class Diagrams (also see footnote 4 in Section 1.3.4). We present the concept types included in each diagram instrument as well as examples of instantiation. The diagrams have no underlying formal syntax or semantics. Templates are not presented here as forms to be filled in, but as checklists, leaving no space for answers but indicating the main issues involved. How exactly templated results are represented and handled is therefore left to be decided by the analyst.

6.3.1 Stage 1: Describing Languages for Operational Communication

The first stage of our method concerns the identification of the various languages that may occur in a Medium System Language Community. Note that therefore, our approach explicitly centres on the medium system. However, this is only a starting point; we emphasise that all languages that members of the MSLC use may be relevant in principle. Depending on the situation, the MSLC may be related to one or more Discourse Environments.

The first step taken is the identification of the information system to which the MSLC belongs. Recall that an MSLC basically consist of medium system users and medium system describers (and thus that not all conceptualisation stakeholders are necessarily part of the MSLC). It may be unclear whether some information system has sub-systems that should be considered as having their own MSLCs, or maybe that various technically independent systems are still together seen as “The System” in an organisational context. Where to draw the lines is a matter of general architecture analysis. However, the link between a system and a coherent medium system language may actually serve as an indication of the demarkation of functional components of the system. Which DEs are related to the MSLC is shown in the MSLC *diagram*.

For each DE deemed relevant, it is helpful to chart the main tasks performed by means of the medium system, as well as agent roles involved in carrying out those tasks. This analysis may well be based on findings related to some other architecture viewpoint, most typically a business process analysis. However, medium system, tasks, and roles here serve the main purpose of identifying *languages* (most typically, terminologies) that are used by agents in the DE. In addition, it is useful to identify any external *language authorities* that may influence language-related choices made within the MSLC, for example organisation-wide authorities that impose legal or professional standards. The results can be described by means of the *language identification diagram.*

Once the different languages used in the domain (or distinguished sub-domains) have been distinguished, we can further analyse and describe them. This may be done at two levels: concerning languages in general, and (if languages include explicitly described *terminologies*) concerning the individual concepts that constitute a terminology. Though the latter level is not central to our approach (which focuses on conceptualisation processes, not concept descriptions as such), it may be necessary

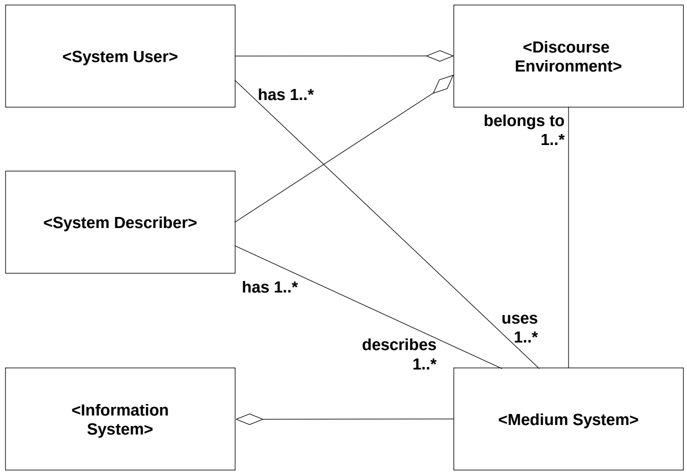

Figure 6.8: 1a: MSLC Diagram, Concept Types

to descend to this level in order to develop an understanding of the concept use situation. Full-scale, comprehensive analysis of individual concepts in the domain, however, seems an exaggeration within our approach, and would fall under a different perspective (possibly, an *information analysis*, *data analysis*, or *domain analysis* effort). Language descriptions can be reported with aid of the *language description template*, concept descriptions are supported by the *concept description template*.
Having charted languages and language use within the DE or DEs distinguished, we can attempt to fill in a symptom template, pinpointing perceived flaws in the current situation: the *concept use symptom template*. Experience has shown that symptom-oriented language use analysis works best if related to individual concepts, but sometimes results can be generalised to 'language' or 'terminology'.

Instrument 1a: MSLC Diagram

The MSLC diagram[3] serves to help explore the various discourse environments that may be distinguished in an MSLC. An optional division of an *information system* (or a *portfolio* of information systems) in component *medium systems* is part of the analysis. The notions of 'medium system user' and 'medium system describer' can be included as heuristics; it is recommended that they are not instantiated at this level of analysis in order to keep diagrams transparent. In Figure 6.8, concept types for the diagram are presented; this is indicated by putting the type labels between <brackets>.
Figure 6.9 gives an example of an instantiated MSLC diagram. It assumes a main 'Auditing System' that supports the operational processes in one DE. The Auditing System in fact consists of two separate medium systems: 'WorkflowAppl' and 'AccountDatabase'. Both medium systems are related to the 'Auditing DE', but both

[3]The notation of the diagrams used in our method is mostly based on the UML specification already mentioned in footnote 4 of Section 1.3.4. Some extensions of this notation will be introduced where relevant.

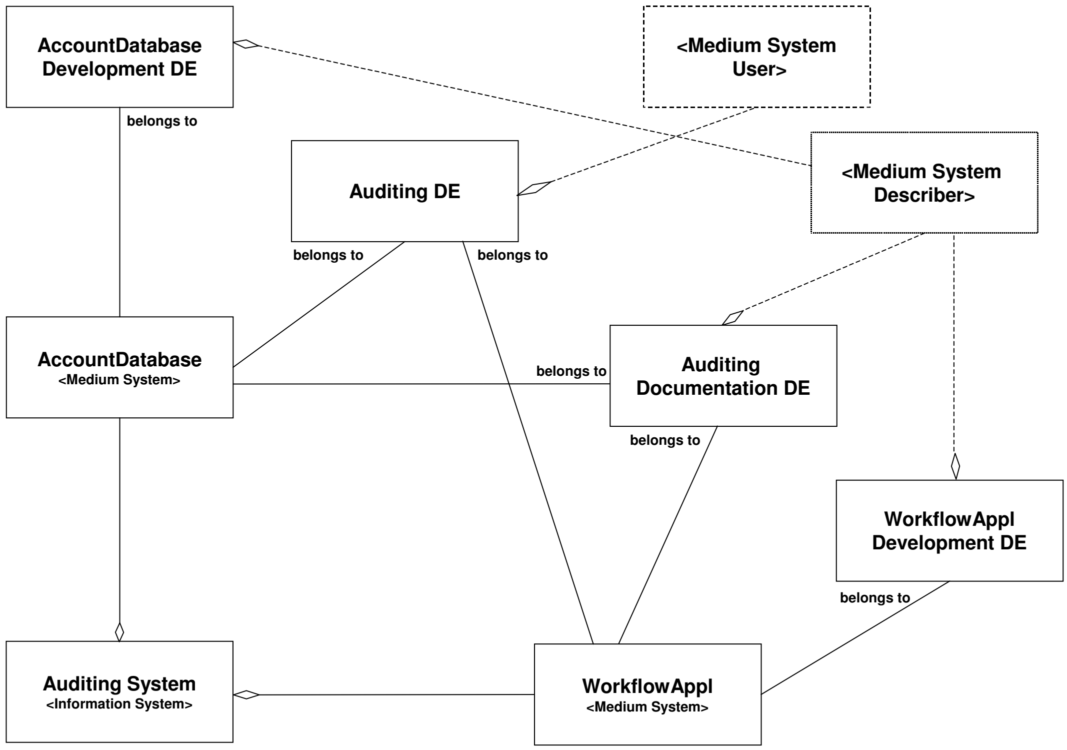

Figure 6.9: 1a: MSLC Diagram, Instantiation for Auditing System

have a separate development DE. There is a specialised DE for the creation of documentation. The diagram shows which DEs involve medium system describers and medium system users, but not as instances. For clarity's sake, these concepts are represented using dotted lines, indicating their heuristic status. Also, the types that the instances belong to are added where clarity demands this. We will continue to do this, so the reader will occasionally encounter added type references in diagrams at instance level.

Instrument 1b: Language Identification Diagram

The language identification diagram is intended to aid the distinction of relevant languages within a DE by means of related medium systems, tasks, agent roles, and language authorities. The concepts at type level and a way of representing them are depicted in Figure 6.10. The 'Language' concept is depicted as a circle because at instance level, languages (being sets of concepts) are represented in a manner reminiscent of Venn-diagrams, allowing for a fairly intuitive presentation of overlap between languages (or the lack thereof). We emphasise, however, that this feature is not intended as a formal descriptive device, but strictly as a heuristic. Some creativity in its application is both allowed and required.
Examples of instantiated diagrams are given in figures 6.11 and 6.12. In the first, the somewhat complex terminological situation in the 'Auditing DE' is depicted.

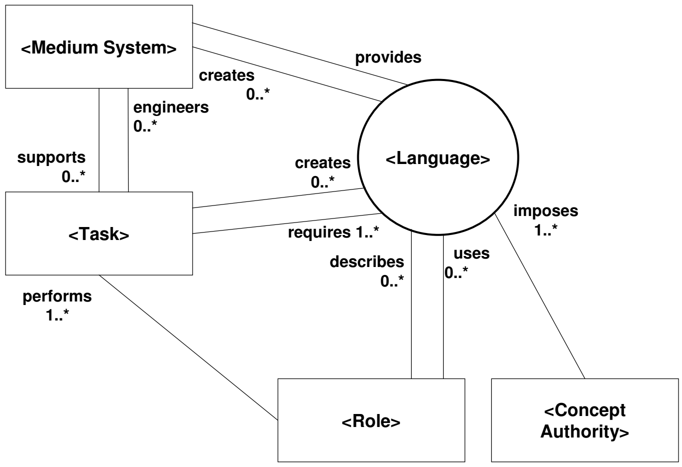

Figure 6.10: 1b: Language Identification Diagram, Concept Types

Note that the "Process Terminology" is both provided by the mediums systems and imposed by the management, implying a deliberate, close link between the operational process language and the frozen language built into the information systems. The relationship between tasks and agent roles are simplified by representing them 'back-to-back'. Also note that "Process Terminology" is identical with the language provided by the two medium systems involved (as indicated by the dotted line), and is divided in two parts, one belonging to WorkflowAppl and one to AccountDatabase. In Figure 6.12, the simpler language situation in the 'AccountDatabase Development DE' is presented. Note 'AccountDatabase Process Terminology' is the *product* of the DE here, and the only langauge directly relevant to the Auditing DE.

Instrument 1c: Language Description Template

The language description template presented in Figure 6.13 is a checklist-like list of questions and items that represent the core issues at this level of analysis. It is basically a reformulation of the Language Use Typology (Figure 4.1), with some limited meta-data added. We refrain from further discussion, presuming all terminology involved to be sufficiently covered by previous sections and chapters.

Instrument 1d: Concept Description Template

The language description template is complemented by the concept description template, which is more detailed but also less central in context of the conceptualisation viewpoint. The concept description template helps describe concepts. It is by no means meant to be used in comprehensive concept analysis, but merely as a detailed aid for finding out more about the languages and terminologies the individual concepts may be part of. On the whole, analysis of individual concepts is optional within the conceptualisation viewpoint. The concept template is presented in Figure 6.14.

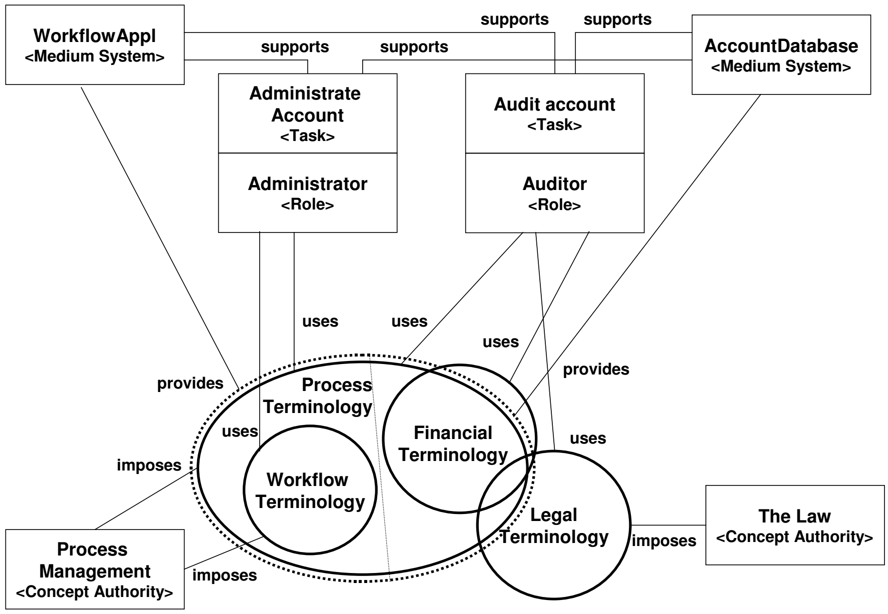

Figure 6.11: 1b: Language Identification Diagram, Instantiation for Auditing DE

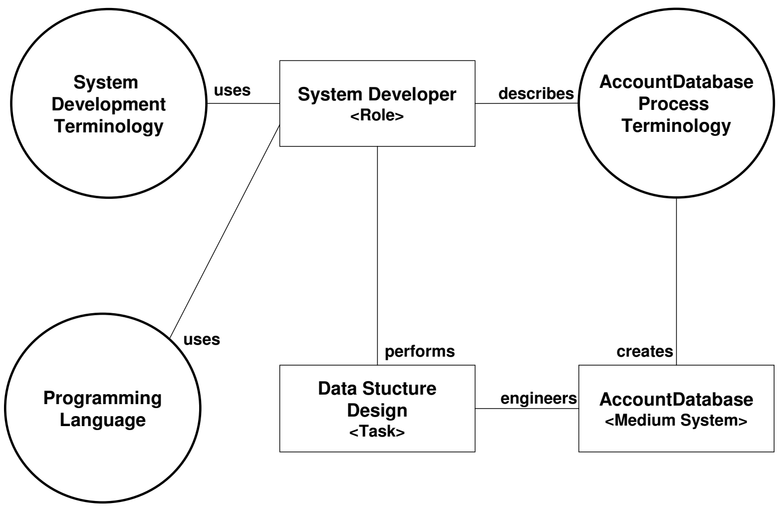

Figure 6.12: 1b: Language Identification Diagram, Instantiation for AccountDatabase Development DE

Instrument 1e: Concept Use Symptom Template

The symptom templates that are included in our method support the analytic effort but border on evaluation. They include questions and items that help ascertain what deficiencies may exist in the current conceptualisation situation, without asking questions as to the 'why' of these deficiencies. The concept use symptom

1c: Language Description Template

Language Description

- *Which specific artefacts are involved in its description?*
- Which agent roles are involved in its description?

General Language Properties

- Common classification: who use it?
    - Topic: is it related to some topic or field?
    - Professional group: is it related to some professional group?
    - Organisational unit: is it related to some organisational unit?
    - Involved in activity/task: is it related to some task or activity?
    - Using medium system: is it related to some medium system?
    - Any other means of classification?
- Global or local use: how many different roles use it (perhaps outside the MSLC)?
- Social properties: how much do people in the DE feel attached to it?
- For intra or inter-DE communication: is it also related to or used in another DE in the MSLC?
- Official status of language: does some authority impose it?
- Computational or communicational use: is it intended or 'talking' or for 'building'?
- Persistency of utterance in use: how long do utterances made with it last?
- Productive or receptive use: do people usually write with it or only read it?
- *Practical properties: does it generally manage to do its communicative job?*
- Operational or auxiliary use: how important is it within the MSLC at large?
- General versus specialised use: does using it mark users as specialists?
- *Contextualised or decontextualised use: is fixed pragmatic information part of it?*
- *Flexibility of language: how fast does it (need to) change?*

Figure 6.13: 1c: The Language Description Template

1d: Concept Description Template

Concept Properties

- The name (form) of the concept/term
- The meaning of the concept/term (described in a format of choice)
- The terminologie(s)/language(s) the concept belongs to
- The task(s) the concept is used for
- The role(s) who use the concept
- Pragmatic aspects 'built into' the concept use (e.g. fixed instance-reference, tightly related operational aspects of use; contextual knowledge assumed)
- Standardisation status of the concept/term: ad hoc – local standard – global standard

Concept Description Meta-attributes

- Author of the concept/term: person or role, or group
- Contact person for concept/term: person or organisational unit to which inquiries concerning form and meaning can be addressed
- Artefact(s) in which an explicit description of the concept form and/or meaning can be found

Concept's Relation to General Language Properties

- Global or local use: how many different roles use it (perhaps outside the MSLC)?
- Social properties: do people in the DE feel attached to it?
- Official status of language: does some authority impose it?
- Computational or communicational use: is it intended for 'talking' or for 'building'?
- Persistency of utterance in use: how long do utterances made with it last?
- Productive or receptive use: do people usually write with it or only read it?
- Practical properties: does it generally manage to do its communicative job?
- General versus specialised use: does using it mark users as specialists?

Figure 6.14: 1d: The Concept Description Template

template (Figure 6.15) focuses on perceived or observed deficiencies concerning the use of individual concepts. The findings may be cautiously generalised and applied to languages/terminologies as a whole. The use of this template is optional; it complements both the language description template and the concept description template.

6.3.2 Stage 2: Describing Conceptualisation Processes

In this stage the focus lies on existing conceptualisation processes in the DEs concerned, i.e. those activities in which people engage in the explicit description of language, as elaborately discussed previously. Such activity may be clear enough without being labelled as "conceptualisation" in the organisation's terms (which indeed will hardly ever be the case). In order to identify conceptualisation processes, we start with looking for concrete exponents, primarily *artefacts* and *techniques* that are typically related to explicit conceptualisation. We can then add *roles*, which should (at least for some part) be related to some language identification diagram rendered in stage 1. All this is captured by means of the *conceptualisation situation diagram*.
Next, artefacts, techniques, and roles can be described in more detail (captured in a *conceptualisation exponent template*). After this, it is a small step to description of the process itself: its phases and sub-phases according to the 'conceptualisation process reference model', their input and output (relating it to identified artefacts), and their detailed relation to one or more conceptualisation techniques. This constitutes the *detailed conceptualisation process diagram*. In addition, it may be possible to say something about the process in terms of the 'general process properties' (functionality, efficiency, usability, maintainability, reliability, portability), and about resources and attitudes concerning the process; this can be done with aid of the *conceptualisation process property template*.
As in stage 1, we can attempt to find and describe various symptoms of trouble concerning conceptualisation. These symptoms can be gathered by means of the *conceptualisation symptom template*.

Instrument 2a: Conceptualisation Situation Diagram

Conceptualisation situation diagrams help to explore the conceptualisation situation and to chart conceptualisation processes that occur in some DE. Such processes may occur regularly and may be part of an organisation's described processes, but it may also be that conceptualisation is an activity engaged in only as an exception, or at a purely ad hoc, implicit basis. We recommend an approach in which *all* conceptualisation processes related to the medium system of focus are charted in the exploratory stage; further, detailed analysis of processes identified is optional and depends on the nature and goals of each particular analysis effort. The concepts of the diagram at type level are presented in Figure 6.16. Two types of roles are included: 'Conceptualisation Roles', which is the main type and is closely connected with conceptualisation stakeholders (Section 6.2), and the more general 'Role', which is optional. Roles may be mapped to conceptualisation roles in order to make clear which conceptualisation roles are played by which general roles (for example, system

1e: Concept Use Symptom Template

The items below may be viewed as concerning either particular concepts/terms or the use of some language/terminology in general

Questions related to Symptoms of Language Dysfunctionality

- Has it occurred that terms used have led to misunderstandings, confusion, or other problems effecting the operational process in the DE?
  Specific examples?
- Has it ever occurred that terms available have been insufficient, or insufficiently precise, to effectively perform operational process in the DE?
  Specific examples?
- Has it ever occurred that people or authorities within or related to the DE objected to the use (mandatory or not) of some particular term or concept, or advocated alternatives? Specific examples?
- Has it ever occurred that the use within the DE of a particular term or concept has been refused, or that the concept proved “useless”, resulting in *disuse*?
  Specific examples?
- Has it ever occurred that a particular term or concept has been *misused*, i.e. used wrongly (deliberately or otherwise) in context of the DE? Specific examples?
- Has it ever occurred that someone has expressed dissatisfaction with the level of standardisation of terminology in the DE (either *too much* standardisation or *not enough*? Specific examples?

Figure 6.15: 1e: The Concept Use Symptom Template

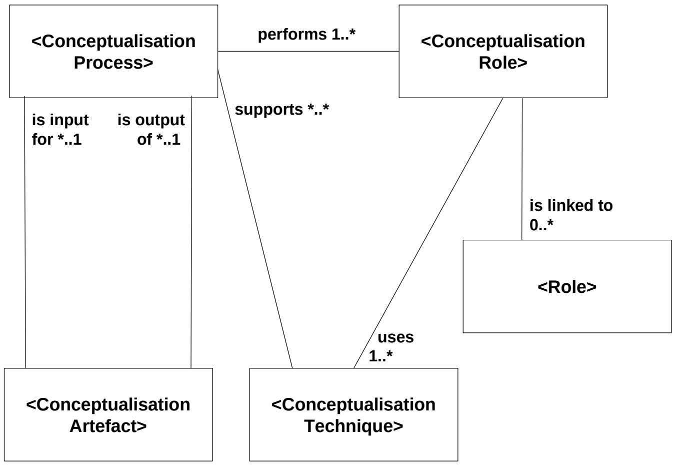

Figure 6.16: 2a: Conceptualisation Situation Diagram, Concept Types

users or system developers).
In Figure 6.17, an example of an instance of the conceptualisation situation diagram is provided for the DE in which the (conceptual load for the) medium system 'WorkflowAppl' is put together. This involves the concept author (a conceptualisation role performed by the system developer), concept informants (some system users plus the auditing process manager), and a concept authority (also the auditing process manager). Six conceptualisation techniques have been identified, listed in one box. Also, three types of artefact have been identified as input and three as output of conceptualisation. All the items identified point to one central conceptualisation process: 'WorkflowAppl Process Terminology Specification'.

Instrument 2b: Conceptualisation Exponent Template

The conceptualisation exponent template (Figure 6.18) helps describe the artefacts, techniques, and roles included in the Conceptualisation Situation Diagram in more detail. Note that quite a few of the items listed are optional. The template should be used as a supporting device; in order to be included, an item should be considered 'clearly useful' within a particular analytic effort.

Instrument 2c: Detailed Conceptualisation Process Diagram

Detailed analysis of a particular conceptualisation process is supported by the 'detailed conceptualisation process diagram'. It involves explicitly distinguished subphases of conceptualisation. These phases *must* be linked to explicitly identified artefacts or techniques. In other words, if no artefacts or techniques are involved, a conceptualisation phase or sub-phase is deemed 'implicit' and is not included in the diagram. It is possible that this renders the distinction of sub-phases superfluous. Knowledge as input and output is represented only optionally, since it is less tangible. Note that 'knowledge input' is related to 'expertise required for participation in conceptualisation' as included under 'roles' in the conceptualisation exponent

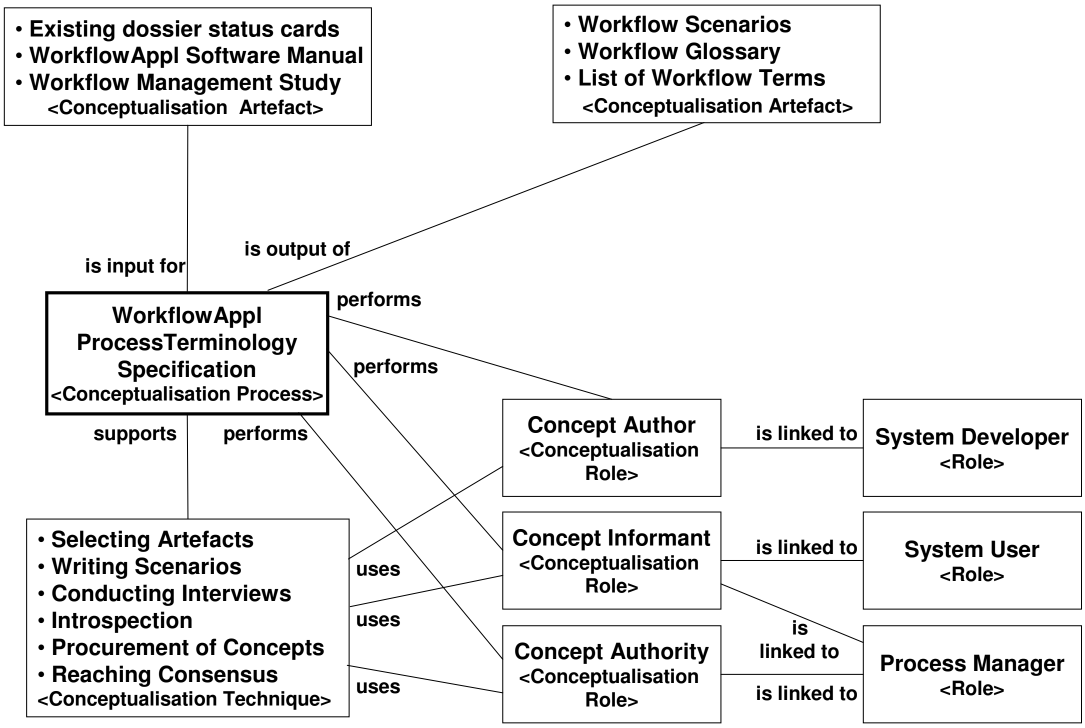

Figure 6.17: 2a: Conceptualisation Situation Diagram, Instantiation for WorkflowAppl Development DE

template. Concept types of the diagram are presented in Figure 6.19.

An example of the use of the detailed conceptualisation process diagram is provided in Figure 6.20. The example concerns further analysis of central conceptualisation process in the previously introduced WorkflowAppl Development DE. Shown are various encountered techniques related to explicitly identified phases of the process, as well as how the known artefacts in the DE are related to those phases as input or output. Knowledge related input and output is left out in the example.

Instrument 2d: Conceptualisation Process Property Template

The Conceptualisation process property template (Figure 6.21) serves to help analysts to describe the properties of a conceptualisation process apart from those already captured in the 'detailed conceptualisation process diagram'. It concerns required resources and expertise, general process properties (functionality, efficiency, usability, maintainability, reliability, portability), the process's relation to the general process and system development process and the Conceptualisation Feedback Cycle (Figure 6.6), and attitudes to the process as exist in the DE. Some aspects can be inferred from findings as described through the conceptualisation exponent template (Figure 6.18).

2b: Conceptualisation Exponent Template

Artefacts
- The name of each artefact
- The origin/author(s) of each artefact
- The person or organisational unit currently responsible for each artefact
- The content type of each artefact (what it describes)
- The format of each artefact (what form it describes it in)
- The goal of each artefact (what it was authored for)
- The authoritative status of each artefact (from informal sketch to standard)
- Optionally, the use an artefact has in the conceptualisation process

Techniques
- The name of each technique (possibly just its type as named in our overview of conceptualisation techniques)
- An estimation (however rough) of the resources required for execution of the technique (either for one use event or continuous use, as found appropriate)
  - Time spent by people (possibly subdivided in roles, e.g.:
    - Informant
    - Author
    - Facilitator
    - Over-all manager)
  - Possibly, cost of training of people
  - Possibly, cost of procuring conceptual material or documentation
  - Possibly, additional support costs (authoring tools, technical support)
- Sketch of properties of the particular technique, possibly in terms of:
  - Functionality
  - Efficiency
  - Usability
  - (Maintainability –in case the technique is applied continuously)
  - (Reliability –in case the technique is applied continuously)
  - (Portability –in case the technique is applied continuously)
- Possibly, the person responsible for the selection of the technique
- Possibly, the main person responsible for the execution of the technique (process leader)
- Possibly, a log (however imprecise) of execution of techniques over time (iteration, time between execution)

Roles
- Names of conceptualisation roles involved
- Their relation to organisational roles
- For each role, expertise (input knowledge) required for participation in conceptualisation (if relevant)

Figure 6.18: 2b: The Conceptualisation Exponent Template

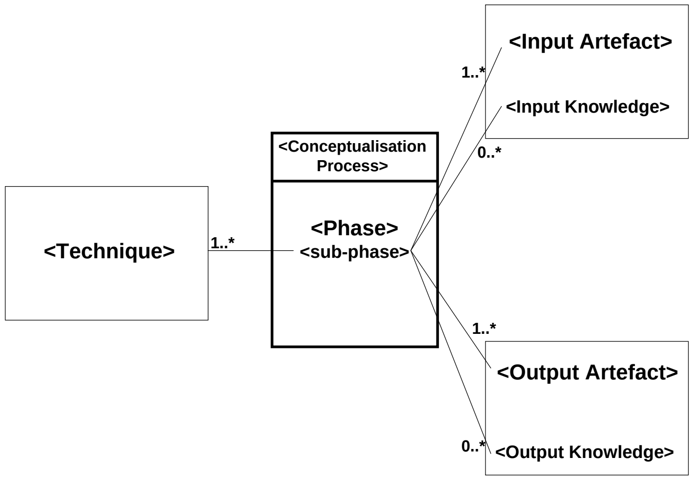

Figure 6.19: 2c: Detailed Conceptualisation Process Diagram, Concept Types

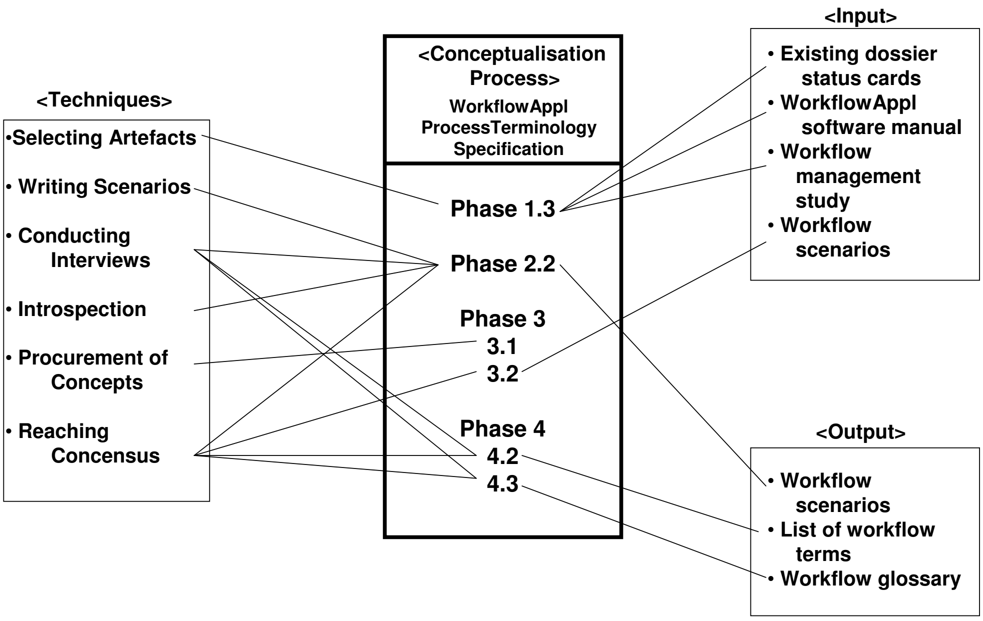

Figure 6.20: 2c: Detailed Conceptualisation Process Diagram, Instantiation for WorkflowAppl Development DE

Instrument 2e: Conceptualisation Symptom Template

The conceptualisation symptom template focuses on observed or reported trouble specifically related to some conceptualisation process, and aspects thereof. It is shown in Figure 6.22.

2d: Conceptualisation Process Property Template

- Who is main person or organisational unit responsible for setting up and initiating the process?
- Who is the main person or organisational unit responsible for correct execution of the process?
- An estimation (however rough) of the resources required for execution of the process (may be compiled from the analysis of Techniques in 2b)
  - Time spent by people (possibly subdivided in roles, e.g.:
    - Informant
    - Author
    - Facilitator
    - Over-all manager)
  - Possibly, cost of training of people
  - Possibly, cost of buying or lending conceptual material or documentation
  - Possibly, additional support costs (authoring tools, technical support)
- An estimation (however rough) of the expertise required for execution of the process (may be compiled from analysis of roles in 2b), possibly divided in expertise related to the conceptualisation process as such and domain expertise needed as *input* to the process
- General process properties in terms of
  - Functionality
  - Efficiency
  - Usability
  - (Maintainability –in case the process runs continuously)
  - (Reliability –in case the process runs continuously)
  - (Portability –in case the process runs continuously)
- The relation of the conceptualisation process to the operational and system development process, possibly with aid of the Conceptualisation Feedback Cycle
  - Does the process take place occasionally, is it structurally iterated, or does it run continuously?
  - *Is it triggered by encountered trouble (ex post) or initiated as part of a (re)design activity (ex ante)*
  - How strongly are the conceptualisation process and the information system development process integrated?
- What are the attitudes concerning conceptualisation in the DE?
  - How important is it considered to be?
  - How interesting or challenging is it considered to be?
  - How tedious is it considered to be?
  - Is conceptualisation (as an explicitly language-oriented activity) perhaps considered a `waste of resources', or rather an `unavoidable burden'?
  - Is conceptualisation considered an activity that is beyond what can be demanded of the people involved?

Figure 6.21: 2d: The Conceptualisation Process Property Template

2e: Conceptualisation Symptom Template

Artefacts

- Are there any known cases of conceptualisation artefacts being
  - Unclear/unreadable
  - Useless
  - Not elaborate or detailed enough
  - Too elaborate or too detailed
  - Cumbersome given the rules for language use they impose
  - Of poor quality (content)
  - Unavailable (i.e. they exist but cannot be found or accessed)
  - Missing completely
- Are there any known cases of conceptualisation artefacts being too costly in creation/maintenance (time, other resources) in relation to what they are used for?

Roles/resources/expertise/permission

- Are there any cases in which people (playing conceptualisation roles) have complained that they lack time for conceptualisation they are supposed to perform?
- Are there any cases in which people (playing conceptualisation roles) have complained that they lack expertise for conceptualisation they are supposed to perform?
- Are there any cases in which people (playing conceptualisation roles) have complained that they lack support for conceptualisation they are supposed to perform?
- Has anyone (playing a conceptualisation role) ever indicated that for some reason she is unable or not allowed to engage in some conceptualisation activity they want/need to perform?

Techniques

- Are there any known cases of conceptualisation techniques applied in the DE that are too difficult to perform?
- Are there any known cases of conceptualisation techniques applied in the DE that are too expensive (time, other resources) to perform?
- Are there any known cases of conceptualisation techniques that should be applied in the DE but for some reason are not?
- Are there any known cases of conceptualisation techniques applied in the DE that are considered boring, confusing, or otherwise troublesome (and perhaps are being avoided)?

Process

- Is there any known case of a conceptualisation process failing for any of the abovementioned reasons?
- Is there any known case of a conceptualisation process lacking where it would be useful?

Figure 6.22: 2e: The Conceptualisation Symptom Template

6.3.3 Stage 3: Describing Linguistic Meta-communication

Linguistic meta-communication usually concerns conceptualisation in order to support operational communication. However, there may also be forms of meta-communication across the MSLC which do not directly involve actual conceptualisation (for example, a helpdesk providing information on the meaning and use of particular database fields). The number of possibilities for meta-communication may on the whole be quite substantial, and a detailed analysis of all occurrences of meta-communication across an MSLC may turn out expensive, while partly irrelevant to our current concerns. We therefore discard meta-communication at discussion level here here (see Section 3.3); it is crucial for communication at large and as a theoretical consideration, yet it is tightly related to particular instances of operational communication and does not directly concern system design. Hence, we focus on meta-communication at discourse level. This enables us to base analyses of (linguistic) meta-communication on the languages distinguished previously for particular DEs, as captured in 'language identification diagrams'.
Considering the languages identified, we ask ourselves what options for linguistic meta-communication exist for each language. We chart the possibilities in the *linguistic meta-communication process identification diagram*, optionally relating processes with roles, dedicated medium systems, and dedicated tasks. We then classify and describe the meta-communication processes thus distinguished using the *linguistic meta-communication process description template*. Finally, we may use a *linguistic meta-communication symptom template* to indicate miscommunication and related trouble.

Instrument 3a: Linguistic Meta-Communication Process Identification Diagram

The linguistic meta-communication process identification diagram supports the rather complex task of identifying the various possibilities for discourse-level meta-communication on the basis of the languages used in some DE. However, a resulting diagram may reach beyond a single DE and is best seen as an extension of the MSLC diagram (Figure 6.9). It effectively links the artefacts and roles belonging to the various DEs distinguished in the MSLC diagram with meta-communication processes. Note that findings in stage 3 may therefore lead to an iteration of stage 1 analysis. However, not every form of linguistic meta-communication deserves to be explicitly modelled as part of some DE. The 'meta-communication process identification diagram' can therefore be used for carrying out a 'completeness check' on roles, artefacts etc. relevant to meta-communication (and possibly conceptualisation).
Two new, rather specific concepts are introduced for the diagram: *information source* and *change option*. They both are not unlike *services*, providing information about a language or enabling a change in some language specification, respectively. The service may be restricted, however, to the availability of some artefact (for example, a dictionary). The concept types included in the diagram are presented in Figure 6.23. Note that 'meta-communication processes' are *not* an explicit part of the diagram. They can be identified by derivation, as follows:

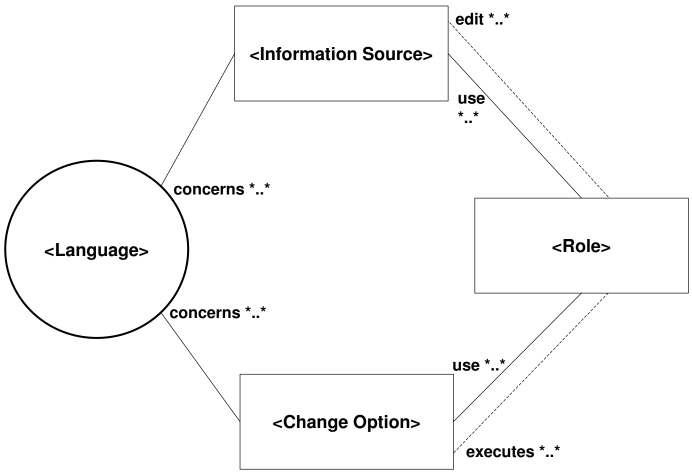

Figure 6.23: 3a: Linguistic Meta-Communication Process Identification Diagram, Concept Types

- each combination of '<role> uses <information source> (edited by <role>) concerning <language>' represents a linguistic meta-communication process

- each combination of '<role> uses <change option> (executed by <role>) concerning <language>' represents a linguistic meta-communication process

Strictly for reasons of representation, the 'edits' and 'executes' relations are marked with a dotted line. In instances of the diagram, no explicit relation names are included.

Instantiation of the meta-communication process identification diagram is exemplified in Figure 6.24. The absence of explicit relation names should not be a problem given the limited possibilities in this respect; the dotted lines serve to distinguish purely informative relations from those potentially involving active changing of artefacts. Admittedly, the diagram is rather complex. Given the substantial number of linguistic meta-communication processes that can be distinguished even in relatively straightforward organisations, this cannot be helped. The best way to read the diagram is by using the two derivation rules mentioned above.

Instrument 3b: Linguistic Meta-Communication Process Description Template

Once the linguistic meta-communication processes in the MSLC have been identified, they (or most likely some of them) can be described in more detail by use of the linguistic meta-communication process description template. Each instantiation of the template concerns one linguistic meta-communication process. The template is presented in Figure 6.25. It is partly based on the 'meta communication typology' (see Section 6.2).

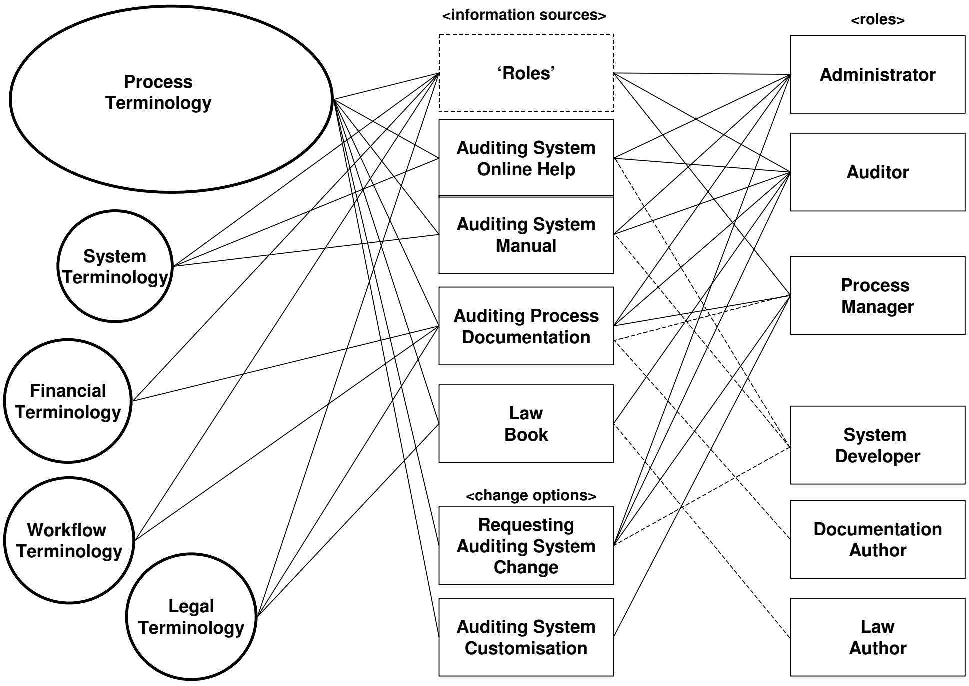

Figure 6.24: 3a: Linguistic Meta-Communication Process Identification Diagram, Instantiation for MSLC, based on Auditing DE

Instrument 3c: Linguistic Meta-Communication Symptom Template

The last instrument for stage 3 analysis and description of conceptualisation processes and their environment is the 'linguistic meta-communication symptom template', which serves to identify any trouble specifically related to linguistic meta-communication. It is presented in Figure 6.26.

6.3.4 Stage 4: Describing Implementation of Concept Representations

The final descriptive stage of our method differs from the previous stages in that it is less elaborate and shows more overlap with traditional ICT architecture domains. It focuses on the technology-oriented aspects of medium system development. The stage does not involve a succession of diagrams and templates, but just a descriptive template (instrument 4a: medium system development template) and a closely related symptom template (instrument 4b: medium system development symptom template). We introduce these without further ado in figures 6.27 and 6.28.

6.3.5 Stage 5: Architectural Evaluation of Conceptualisation Processes

After concluding the descriptive stages of our method, and given that the required information has indeed surfaced, a comprehensive and coherent picture of concep-

3b: Linguistic Meta-Communication Process Description Template

First describe the linguistic meta-communication process in terms of one of the two derivation rules:

<role> uses <information source> (edited by <role>) concerning <language>
<role> uses <change option> (executed by <role>) concerning <language>

- Information Source User/Change Option User
  - Which roles are involved? (general and conceptualisation roles?)
  - What expertise is the typical source/change option user expected to have?
  - How much time/effort is the typical source/change option user expected to be able/willing to spend on accessing and using the source/exploring and using the change option?
- Information Source/Change Option
  - Which artefacts are involved (if any)?
  - Are there any services involved (e.g. a helpdesk)?
  - Is the information source/change option integrated with some medium system?
  - If the source involves a role, answer the questions on “information source editor/change option executor” for the role
- Information Source Editor/Change Option Executor
  - Which roles are involved? (general and conceptualisation roles?)
  - What expertise is each role typically expected to have?
  - How much time/effort is a role typically expected to be able/willing to spend on keeping the source in order/executing the required changes?
- Which languages/terminologies are involved?
- Is there a relation with some conceptualisation process?

- Can the process be classified as ex ante or as ex post meta-communication?

- Can the process be classified as intervening or as non-intervening meta-communication? (Note that this is related to whether the first or second derivation rule is used, respectively)

- Can the process be classified as non-mediated or as mediated meta-communication?

- Can the process be classified as language-related or as implementation-related meta-communication (or both)?

Figure 6.25: 3b: The Linguistic Meta-Communication Process Description Template

3c: Linguistic Meta-Communication Symptom Template

- Has any Information Source User expressed dissatisfaction with the meta-linguistic information available?
- Has any Information Source User expressed dissatisfaction with the way meta-linguistic information can be obtained?
- Has any Change Option User expressed dissatisfaction with the change options available?
- Has any Change Option User expressed dissatisfaction with the way change options can be used?
- Has any Information Source Editor expressed dissatisfaction with the meta-linguistic information available?
- Has any Information Source Editor expressed dissatisfaction with the way meta-linguistic information can be edited?
- Has any Change Option Executor expressed dissatisfaction with the change options available?
- Has any Change Option Executor expressed dissatisfaction with the way change options can be executed?

- For processes classified as ex post, has any agent involved expressed dissatisfaction with the ex post character of the process (i.e. calling for an ex ante process)
- For processes classified as ex ante, has any agent involved expressed dissatisfaction with the ex ante character of the process (i.e. calling for ex post process)

And so on for:

- Intervening and non-intervening meta-communication
- Mediated and non-mediated meta-communication
- Language-related and implementation-related meta-communication

Figure 6.26: 3c: The Linguistic Meta-Communication Symptom Template

4a: Medium System Development Template

Data Structure Design (mostly related to stage 1)

- Are there technical restrictions related to data structure design that influence the options available for 'Medium System Language Design?'
- What are the resources generally available for data structure design?

Human-Machine Interface Design (mostly related to stage 1)

- Are there technical restrictions related to human-machine interface design that influence the options available for 'Medium System Language Design?'
- What are the resources generally available for human-machine interface design?

Machine-Machine Interface Design (mostly related to stage 1)

- Are there technical restrictions related to machine-machine interface design that influence the options available for 'Medium System Language Design?'

Technical Infrastructure

- Are there any further technical restrictions or properties of the technical infrastructure that may restrict possibilities for 'Medium System Language Design'?

Methods and Techniques in Medium System Development (mostly related to stage 2)

- Are there any aspects of software/system development *methods and techniques* involved that relate to conceptualisation/medium system language?
- Do these methods/techniques impose restrictions on conceptualisation or do they explicitly guide conceptualisation towards a technology-dependent standard solution?

System-Based Support Mechanisms for 'Change Options' and Linguistic Meta Communication (mostly related to stage 3)

- Are there any mechanisms or options integrated in the technical system that are related to linguistic meta-communication (i.e. changes in data structure, thesauri, ontologies, or any other form of language change intended to improve operational human-human communication)?

Figure 6.27: 4a: Medium System Development Template

4b: Medium System Development Symptom Template

- Are there any cases in which language-related trouble has been blamed on lack of necessary *expertise* of technology-oriented system developers?

- Are there any cases in which language-related trouble has been blamed on lack of *resources* assigned to the technical (implementation) aspect of language-related system design?

- Are there any cases in which language-related trouble has been blamed on lack of *cooperation* of the technical stakeholders involved in language-related aspects of system design?

- Are there any cases in which language-related trouble has been blamed on fixed properties (language-related or otherwise) of the technology (software, hardware) underlying the implemented medium system?

- Are there any cases in which language-related trouble has been blamed on methods or techniques applied in medium system development?

- Are there any cases in which language-related trouble has been blamed on (lack of) technical support for communication about medium system language?

- Are there any cases in which language-related trouble has been blamed on differences in languages between the 'technical community' and other sub-communities within the MSLC?

Figure 6.28: 4b: Medium System Development Symptom Template

tualisation processes and their environments should have been created. However, we have not yet addressed the question whether, in terms of conceptualisation and meta-communication, the situation is anything from ideal to disastrous. We did include the description of explicit symptoms of trouble occurring (based on reported situations and opinions of stakeholders) but have not provided any means for active, architecture-level evaluation by analysts.

Within context of Stage 5 we provide some guidelines for evaluation of conceptualisation processes in relation to their language and communication environments. This should be seen as no more than a first step towards a complete, rational evaluation system. It is not our intention (nor within our current reach) to provide any sort of quantified system for evaluation of conceptualisation processes in ICT supported organisations. However, we can indicate certain points of consideration that (especially if complemented by reported symptoms) can help greatly in a focused assessment of how well-organised and successful linguistic meta-communication processes are in some organisation –in view of their conceptualisation-related goals, requirements, and restrictions. These 'points of consideration' typically involve comparison of some particular findings as described in the four descriptive stages. 'Mismatches' between requirements and means deployed indicate (potential) trouble. Identification of mismatches points towards possible directions for improvement, and may thus be used as a basis for decision making.

An important goal of 'architecting' involves finding a balance between various contradicting goals and priorities throughout an ICT-supported organisation. Achieving optimal *alignment* of goals and means in an organisation, for as many aspects as possible, is, as it were, the ultimate prize for an architect. Stage 5 of our method aims at helping architects get nearer to winning that prize by providing insight into conceptualisation situations. In Figure 6.29, we sketch the relation between stage 5 and stages 1-4 with respect to the alignment process, which will almost unavoidably involve iterative analysis and change concerning items and activities focused on in all 5 stages. Note, however, that our method only involves analysis and evaluation of (present or envisaged) situations; it does not involve 'change management' as such, i.e. the means to create a different conceptualisation situation.

For each matching point, particular items are matched that belong to the its specific underlying concerns. Items may be combined if they are sufficiently similar. We suggest a rough standard structure to be used in describing the matchings:

- Means: the item that is analysed, or concrete exponents thereof
- Purpose: the basic functional requirements of the item
- Match: whether or not the means fit the purpose
- Situational factors: any present and unchangeable factors that may play a role (optional)
- Improvements: what might be realistically altered to improve the situation (optional)

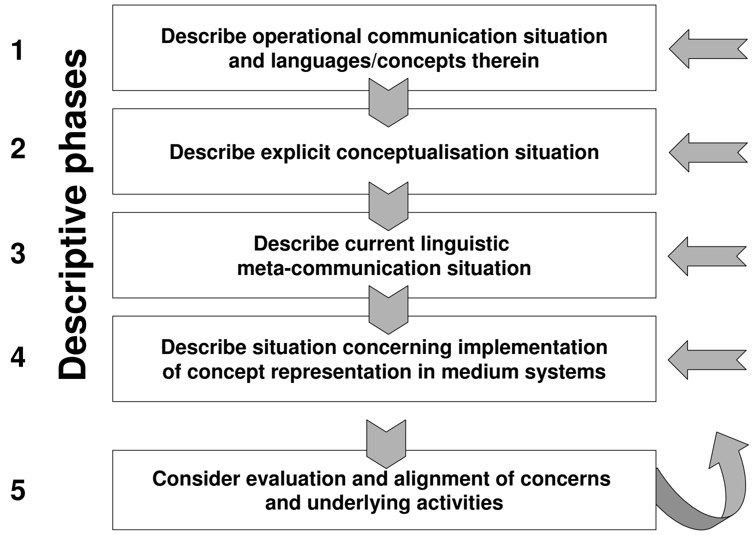

Figure 6.29: Alignment in Relation to Stages in the Method for Conceptualisation Process Analysis

Matching Point 1: do languages/concepts match their operational communicational requirements?

The first matching point concerns the functionality of languages and concepts in operational communication (in some DE). In order to identify relevant DEs, the MSLC diagram (instrument 1a) is helpful, but it plays no crucial role in matching linguistic goals and means of communication. More central are the language identification diagram (1b), language description template (1c), and the concept description template (1d), though the latter is optional.

A match between 'means' and 'ends' here implies that a language's functionality is reasonably in line with the requirements posed by its environment of use. Thus, for each language as used in a particular DE, we should inquire *what purposes it serves*, and *whether it suffices in view of these purposes*. We may relate our findings with *situational factors* that influence the functioning of the language.

Problematic situations that may be explained by mismatches can be explored and made concrete by looking at some individual concepts and the use they are put to. General properties of the language use situation in the DE can be derived from information gathered through the second section of (1c), but this analysis may benefit from findings related to (1d) and (1e). Importantly, however, *matching Point 1 does not include individual concept analysis*; comprehensive conceptual analysis would go way beyond the goal of an architecture viewpoint on conceptualisation processes. Detailed concept analysis belongs to fields like language engineering, information structure analysis, or knowledge engineering. Evaluation of functionality in terms of the actual meaning borne by a particular concept (and whether it fits the situational requirements) falls outside the immediate scope of our method. Thus, the question whether or not the results are satisfactory is important, as are some general considerations, but not the question why *exactly* some particular concept does or does not fit its use situation –unless as an indication of more generic language (dys)functionality. In other words, matching point 1 serves to indicate how well language and conceptualisation work, but in order to find concrete, detailed examples of deficiencies, information will need to be drawn from experience and knowledge concerning bordering viewpoints. Such dependency lies in the nature of any architectural approach.

Matching Point 2: do available linguistic meta-communication processes match the requirements posed by potential lack of concept functionality?

The second matching point concerns the question what linguistic meta-communication processes may be called in to complement available concepts. This includes clarification, selection, creation, and adaptation of concepts. The 'concept use symptom template' (1e) may be of use in understanding the nature and scale of conceptual dysfunctionality, i.e. the need for meta-communication. Another helpful instrument here is the 'linguistic meta-communication process identification diagram' (3a). In combination with the 'meta-communication process description template' (3b), it quite comprehensively brings to the fore all the intervening and non-intervening meta-communication processes available in an MSLC, i.e. the means available for linguistic meta-communication. The requirements to be fulfilled by those means concern the countering of structurally occurring forms of conceptual dysfunctionality as identified through matching point 1. Actual occurrences of dissatisfaction with options for meta-communication can be explored through a symptom template (3c).

A particularly interesting aspect of matching meta-communicational goals and means involves the balance between ex post and ex ante meta-communication. As discussed in Section 4.4, in situations where ex post meta-communication is difficult or downright impossible, demands for high quality ex ante meta-communication increase substantially[4]. Conversely, abundant options for ex post meta-communication may compensate for inescapable restrictions imposed on frozen language functionality. Note that ex ante meta-communication can be intervening (creating proper frozen language) but also non-intervening (for example, people may be extensively trained to cope with the use of deficient frozen language).

Even if meta-communication processes seem to be in place, it is advisable to extend the analysis and also look at meta-communication from a more practical angle. Thus, we should also look at the practical means of communication (mediums, people, resources) available for support of meta-communication. These matters are covered to some extent by the 'meta-communication process description template' (3b).

Matching Point 3: do conceptualisation processes match the requirements posed by (relevant) change options related to intervening meta-communication processes?

Once it is clear what the situational requirements of meta-communication are, including requirements of intervening meta-communication (both ex ante and ex post),

[4]In addition, the situational capacity for engaging in 'discussion level linguistic meta-communication' is relevant here.

we can consider the match between these and conceptualisation processes that may be engaged in. The requirements ('ends') can mostly be derived from stage 3 findings, in particular those related to instruments (3a) and (3b). The conceptualisation processes as such (the 'means') can be described at a general level using the 'conceptualisation process property template' (2d). On this basis, one can start exploring how well particular phases, techniques, and artefacts fit the situation at hand; here, the 'detailed conceptualisation process diagram' (2c), the 'conceptualisation exponent template' (2b), and the 'conceptualisation situation diagram' (2a) are helpful.

Matching Point 4: does the technical development infrastructure underlying medium system implementation match the requirements posed by the the relevant conceptualisation processes?

Lastly, some of the more technology-oriented aspects of system development and design need to be (brought) in line with conceptualisation processes. We look at the *technical means* (software, hardware) available for implementation and deployment of proposed changes in medium system language, as well as the *system development procedures and mechanisms* that envelop the application of technological means. The matching point is only partly covered by our method, since it requires insight into the system development situation that may go beyond our language-oriented scope. Various related aspects are covered by the 'medium system development template' (4a); some typical questions involving problems with those aspects are explicitly raised in the 'medium system development symptom template'.

More Matching Points, and Environments for their Application

The four main matching points described so far do not comprehensively cover the possibilities for matching 'goals and means' across the rather broad spectrum of matters related to conceptualisation processes in ICT-supported organisations. More detailed, possibly partly formalised techniques might be introduced to refine the requirements, means, and matching between them; in principle, it seems even possible to design a decision support system on this basis. Perhaps more realistically, the applied description and matching of the issues discussed throughout this thesis may become part of principles, patterns, viewpoints, and heuristics underlying an advanced ICT-architecture description and management environment. This is not the place to carry this line of thought further, but see Section 8.2.

6.4 Summary

In this chapter, we have taken the main findings of earlier chapters (i.e. various conceptual devices for analysis and description) and on this basis have put together a method for describing conceptualisation processes in their environment. The method and its descriptive/analytical instruments have been shaped as an Architecture Viewpoint: an analytical and communicative instrument that is dedicated to a particular aspect of systems architecture. We first listed the existing concepts relevant to our enterprise (Section 6.2), and added some others (in particular, some stakeholders and concerns appropriate for the viewpoint). We also presented some perspectives on combined aspects of conceptualisation.

We then presented a five stage method for the analysis of conceptualisation processes in an organisational context (Section 6.3): four descriptive stages and one evaluation stage. The stages are derived from concerns distinguished within the viewpoint. They should not necessarily be executed in any fixed order, but we recommend starting out with the analysis of concrete, easily observable phenomena in an organisation (stages 1 and 2), work towards deeper and more theoretical analysis (stage 3), and finally describe links with system development practices of the organisation (stage 4). The descriptive instruments used are specialised UML-based *diagrams*, and *templates* (focused lists of questions and 'points to check'). The evaluation stage (stage 5) is based on the deployment of four matching points that indicate what aspects (described in the previous stages) need to 'fit' in terms of requirements and means available. The matching process is not deterministic or based on quantified metrics; this is in line with the general nature of 'architecting'. However, it provides concrete clues as to the factors than need balancing in order to make the conceptualisation aspects of an architecture 'work'.

By presenting the method, we have rounded off our answer to Q2.1, and have come a long way in answering Q2.2 (that is, as far as can be achieved in view of our current thesis). What is still needed is *validation* of the method: though its concepts and the coherence between them are the result of careful consideration and are for a considerable part backed up by theory, more is needed to provide at least a 'proof of concept' of the method. In the next chapter, we present a detailed case study in which the method is applied. In the conclusion to this thesis (Chapter 8), we will further discuss matters of validation and suggest further steps in order to reach a higher degree of validation than is possible within our current scope.

Chapter 7

Case Study at the Joint Administration Office (GAK)

In this chapter, we present a 'real world' study of conceptualisation processes and related matters in a large ICT-supported organisation. The study took place between March and December 2001, at the *Gemeenschappelijk Administratie Kantoor* (Joint Administration Office). The organisation preferred[1] to use what was originally the acronym GAK without explicit reference to its origin, i.e. as 'Gak', and as a whole was called *Gak Nederland* ("Dutch Gak").

The case study mostly serves to demonstrate that the descriptive and analytical method presented in the previous chapter can be applied in a real situation, thus providing a 'proof of concept' of the method and the conceptual framework underlying it. It should convincingly show that the main concepts in our framework indeed apply: that instances of the concepts introduced can be meaningfully pointed out in the domain, and in particular that they help make better sense of a complex language and meta-communication situation than was previously possible. It should thus help chart and understand problems in the domain that are within scope of the method, and possibly assist in finding directions for solutions of those problems. The case study mainly contributes to the answering of Q2.2: "On the basis of the conceptual framework, develop an analytical method enabling the analysis and evaluation of concrete processes and environments of freezing language in real ICT situations"; more in particular, it should show this goal has been achieved (validation).

We first briefly describe the rationale behind the case study and the approach taken, and report our experiences with the realisation of the study (Section 7.1). We then briefly describe relevant aspects of the RESA/FASA information system, which lies at the centre of our case study (Section 7.2). Next we apply our architecture viewpoint for conceptualisation processes to the Gak situation, more in particular the RESA/FASA Medium System Language Community. This is Section 7.3, which is the core section of the chapter. It is followed by a brief discussion of results and a conclusion (Section 7.4). Importantly, the conclusion includes an elaborate discussion of how much each of the chief concepts and analytic instruments of the method have individually contributed to our analytic effort (i.e. 'how well they worked'). A general description of the Gak organisation and the RESA/FASA information system

[1] In 2002 (after our case study was finished) the organisation was merged with its four Dutch sister organisations into the 'UWV'; see below.

can be found in Appendix C.

7.1 Case Study Setup and Realisation

The main goal of the case study was to provide a 'proof of concept' of the method (architecture viewpoint) for conceptualisation processes and their environments as presented in Chapter 6. The case provides ample example material for demonstrating the workings of our architecture viewpoint for conceptualisation, and the method it includes. However, when the case study was carried out, the architecture viewpoint had not been developed yet, though most of the underlying conceptual instruments had already taken their basic shape. The case study has therefore also been crucial in developing a feeling for how matters covered by our theory present themselves in real practice. Also, the final shaping of the method and in particular its instruments (diagrams, question templates) was influenced by it. We acknowledge that the strength of what is arguably a self-referring validation is somewhat limited. Nevertheless, we believe the case study contributes much to the thesis. For further discussion of the validation issue, see Section 8.1.4. In the conclusion to this chapter, we will systematically discuss how the various concepts, diagrams, templates, and methodical stages contribute to analysis of the case. This is how we achieve explicit proof of concept of our method.

7.1.1 The Case Organisation

Gak Nederland (which as an independent organisation has ceased to exist in 2002) had two core tasks: to implement social insurance laws (roughly, those laws safeguarding a form of income for the unemployed and the disabled; a combination of six different laws) and also to make sure that a maximum number of people moved out of the benefit system back into the active labour force. The organisation resembled a large insurance company, requests for benefits being seen as the claiming of "social insurances" and, contracts as "insurance policies". Our research took place at *Gak Gelderland-Zuid* ("Gak South Gelderland", which is a district covering two regional offices: Nijmegen and Arnhem), and at the Amsterdam headquarters.
In 2000, more than 200.000 employer organisations were associated with Gak, together employing almost 3.9 million people (who of course were not all enjoying benefit through Gak, but were potentially *entitled* to it). The number of people enjoying unemployment benefit nationwide was almost 190.000 in 1999, representing a value of 3 billion DFL (more than 1.4 billion euro). An average of 400.000 people received disability compensation in 2000, 10 billion DFL (almost 4.8 billion Euro). Gak employed around 12.000 people in 28 offices distributed over 17 *regio's* (regions or districts) and the Amsterdam headquarters.
Gak was one of five executive organisations that implemented social benefit laws in the Netherlands, though by far the biggest. All five organisations were contractors of the LISV, the *Landelijk Instituut Sociale Verzekeringen* (National Institute Social Insurances). As of January 1, 2002, (after our case study was finished) all implementation of social insurance laws came under one organisation, merging Gak with its four colleague organisations. However, the size of Gak guaranteed its major role

in the set-up of this new organisation, called UWV (*Uitvoeringsinstituut Werknemersverzekeringen* or “Institute for the Implementation of Employee Insurances”). The LISV was also incorporated in the UWV.
Further description of the Gak organisation can be found in Appendix C. From this point on, we discuss Gak as if it still exists as it did before January 2002, for the sake of more agreeable presentation: persistent use of past tense felt cumbersome and unnatural when we tried it, even though the Gak has by now been reorganised and renamed. For the case study as currently presented, the situation at the moment the study took place is all that matters. Focus on a specific part of the organisation was added to the case as it progressed. We ended up looking at one particular, centralised information system that is used in one of the two main branches of the organisation.
The Gak organisation was selected as the subject of our case study because it was a large organisation with strong dependence on ICT, revolving around an information-intensive production process that over the years has been subject to many concept-oriented changes. This made it an interesting research subject. Also, Gak showed itself cooperative, allowing us to interview a number of their employees without demanding results immediately useful for the organisation. However –and understandably– Gak officials made it clear at an early point that time allowed would be limited. Finally, in a previous life, the author of this thesis has worked in one of the sister organisations of Gak, which provided him with something of a head start in terms of understanding the organisation, its main work processes, and its culture.

7.1.2 Research Set-up

The case study, which was from the beginning subject to restrictions of time and resources, was to be qualitative rather than quantitative, given that it was an important part of an effort to the develop and validate a descriptive and analytical *conceptual* framework. Describing, understanding, and structuring were the focus, not measuring or calculating. The data gathering consisted of a series of carefully prepared interviews, and some study of internal documentation and other artefacts. Initially, we had hoped to be able to include some observations of actual communication, meta-communication, and conceptualisation processes in the organisation, but this turned out impossible given limited time and resources, both of Gak (especially in view of the upcoming UWV reorganisation) and of the author of this thesis.
Roughly, this strategy was followed:

- Find a key official and secure cooperation
- Ask the key official to help out with contacting other key officials
- Find key informants within department of the key official
- Prepare questions, on the basis of previously gained information if relevant
- Carry out and record interviews; transcribe/process the recordings and notes taken

- In the mean time, move on to a new department (helped by a key official).

As a result of our strategy, which allowed for step-by-step exploration of various branches and levels in the Gak organisation, 'question templates' could be developed incrementally as the study advanced. From the outset, we opted for a combination of a focused search for symptoms of conceptualisation problems and the charting of the multi-facetted organisational situation. This combination was an explicit and deliberate part of the prepared question lists. The templates used in the study were somewhat different from those eventually included in the architecture viewpoint and its method, though they covered essentially the same ground. The original templates were much more elaborate, and less clearly structured and focused, than their better developed counterparts in the method presented in Chapter 6. The diagram-based instruments presented were developed only after the data gathering phase of the case study was completed. Much information that was later captured in diagrams was initially elicited through questions. As explained above, the distilling of means to present a structured overview of the conceptualisation situation in an organisation was a secondary goal of the case study.

Even though prepared lists of questions were used extensively in the interviews, we did not apply a strictly 'closed' interview technique, as we realised that the ideas underlying our questions were still under development; learning about new issues and questions was necessarily part of the process. Thus, while attempting to somehow get at least our key questions answered, we allowed interviewees to freely talk about examples, situations, and problems they saw as relevant in context of the interview. This was also a bare necessity because of the underlying topic of our research, which has little to do with regular Gak operations. One of our biggest challenges was the translation of our initial questions to questions that made sense to Gak employees. For example, in order to enable interviewees to understand the topic and 'think along', we referred to "terminology" and "language", not "concepts". We avoided theoretical terms like "meta-communication", "conceptualisation", etc., and attempted to ask questions using either "common language" or "Gak language". The burden of translating the Gak situation, phrased in Gak terminology, to the conceptual framework of the architecture viewpoint thus firmly lay with the interviewer, not the interviewee.

7.1.3 How it all Went

We came to a total of 12 interviews, between one and two hours in length. The first took place on March 12, 2001, the last on December 4, 2001. Between interviews, results were processed and analysed and question templates developed. We interviewed people with the following roles in the Gak organisation[2]:

1. 3-12-2001 Acting Director of GZ and Head of Quality Control GZ
2. 3-28-2001 ICT Manager GZ
3. 3-29-2001 Head of Quality Control GZ

[2]For a description of Gak organisation, organisational roles, and locations like GZ and DZ, see Appendix C.

4. 5-4-2001 Head of Quality Control (DZ, Headquarters)
5. 5-4-2001 ICT Manager DZ Staff-WW (IT Dept., Headquarters)
6. 7-11-2001 Process Office Coordinator GZ and Process Support Consultant GZ
7. 7-17-2001 Quality Controller GZ
8. 7-2-23-2001 Claim Assessor GZ
9. 7-23-2001 Staff Claim Assessor GZ
10. 10-8-2001 Head of Quality Control GZ
11. 12-4-2001 IT Architect (ASZ, Headquarters)
12. 12-4-2001 Process and System Designer (DZ, Headquarters)

Without exception, the interviewees showed themselves cooperative. As informants, their expertise and general ability to function as informants turned out to vary from almost ideal (in four cases) to somewhat disappointing (one case). Generally, the results of the interviews were quite satisfactory in view of the limited setup. Twice, an interviewee took the initiative to bring along a second informant, which turned out very well. Thus, two interviews were simultaneously held with two interviewees. This posed no practical problems to speak of. With only one exception, the interviewees considered themselves good informants and showed and expressed considerable interest in the topic of discussion.

Because of the open interview technique applied, it regularly occurred that some prepared questions were not explicitly answered, or even not answered at all. However, details missed because of this were generally deemed less important than the unexpected yet relevant information that came up spontaneously. It is unfortunate that, with a few exceptions, we were unable to return to interviewees with additional questions after the interview had been processed. The premature character of the prepared questions has turned out not very problematic: 'translation' was needed anyway (see above). However, the approach followed has not been optimally systematic and comprehensive in terms of information gathering. This is a price we have paid for the explorative nature of our study.

Willingness to spend some time on interviews was notably greater at Gak headquarters and in the higher ranks of the local office than in its lower ranks. To gain cooperation of people directly involved in the main operational processes, some diplomacy was required to gain permission from the local executive to "draw time from the production process". Here, the daily pressure of core operations was clearly felt. This prevented us from adding an observational component to our case study, which we originally intended.

Various interviewees were willing to make copies or print-outs of informative documents (even some 'screen shots'). In general, however, the number of documents gathered was limited. In accordance with previously made agreements, no details of Gak system documentation can be disclosed in their original form. This entire chapter has been screened and approved by an UWV official.

7.2 The RESA/FASA System

Early on in our case study, we decided that to analyse a conceptualisation situation at various levels, including a quite detailed level (requiring the analysis of some individual concepts), we needed to focus on a particular application, not Gak-AG[3] as a whole. Once we had chosen to work with the AG product branch, and had indicated that 'legacy systems' were interesting to our goal, the RESA/FASA system was among those suggested as a potential point of focus. Its functionality puts it central in the AG claim assessment process, which has been subject to many changes over the years. It is essentially a database system, a typical 'information and communication system': a medium machine that is chiefly intended to enable specialised communication between people, apart from feeding and triggering computational actions. Also, it involves over a thousand 'frozen concepts'. All in all, it is a system that seemed well suited as the core item of our case study: the 'MS' in our MSLC.

In the name RESA/FASA, "RESA" stands for REgistration System AAW[4], "FA" stands for "Financial Processing[5] System AAW". Tellingly, most interviewees could not provide us with a definite and official explanation of what the full acronym stands for. The system was first deployed nationwide in 1992, and was based on an existing system that was developed and successfully used at a local AG office. It has since been subject to many expansions and changes, mostly in response to law changes.

In its current role, the RESA/FASA system serves to register the detailed state of all client dossiers in relation to the general work process of the AG production process. Most prominently, it shows what actions within the process have been initiated in view of a particular dossier, and what decisions have been taken. The system thus serves as a workflow registration and management system, but also gives access to most if not all available digitised information concerning the actual content of a dossier (for example, what kind of benefit is enjoyed a client, or according to what schema she is being paid). The system is therefore central to the whole claim assessment process and also plays an important role in the administrative processes (mostly payment of benefits) that take place iteratively after a claim has been assessed. The system can be accessed (at various levels of authorisation) by quite a few employees, with a number of organisational roles.

Considerable time after its initial introduction, RESA/FASA functionality has been expanded to the registration and management of the *products* that are created during the claim assessment process. Those products are mostly related to the various kinds of *decisions* that are the main result of the claim assessment process. For example, if a claim has been assessed and is rejected, this results in a completed product which can be billed to the LISV[6]. Management of the automated execution of payments for a client is also a product. Hundreds of different products (differently priced) have

[3] AG stands for "ArbeidsGeschiktheid", literally *working-ability*, but in fact it euphemistically refers to the department taking care of disability benefits, though Gak also actively tries to help people overcome disabilities and return to work.

[4] AAW is the "Algemene Arbeidsongeschiktheids Wet", *General Disability Law*. The law has in fact been differently named since RESA/FASA was introduced.

[5] "Financiele Afhandeling" in Dutch.

[6] LISV was the governmental organisation dictating how social security laws were to be implemented; it defined 'products' and paid Gak and its sister organisations for services rendered. For more on this, see Appendix C.

```
AAWFRM1201A-11                                  23-Jul-2001 12:36:53
             MUTATION OF INFO- AND SIGNAL ACTIONS

 capdate   : 03-08-01     wag person MR. J. DOE
 fisc nr   : 9646.23.598             12 CHURCH ST.
 reg number: 556.765432.0            34F WK3 WILLOWHOW
 ____________________________________________________________________

    sel action description      startdate dk fun pr rappel   nr proc

    1   537   Appl WAO           24-09-99           10-01-00    1 24-09-99
    2    265   Decision WAO      03-01-00     52  3 17-01-00      11-02-00
    3   141   Assig.Emplmntinv.  24-09-99           10-01-00      11-02-00
    4   677   Decis. Review      28-07-00     52 16 13-10-00      23-08-00
    5   141   Missing incomeinq. 02-03-00           11-07-01 10

                       next transaction: 1201 / 7635.56.443 / 53432.0
```

Figure 7.1: An impression of RESA/FASA "screen 1201"

been defined by the LISV; 36 main categories are distinguished, and many nuances add to the general complexity. The RESA/FASA system thus has become burdened with the added functionality of a fairly advanced accounting and billing system. If products are not properly registered through the system, the line organisation executing the work process does not receive any money for the service provided, since formally a delivered product only exists if registered in the system.

RESA/FASA is a rather dated transaction-based database system. Information entry and representation for a large part concerns codes (without exception 1-3 digit numbers), in some cases complemented with an entry description (words or short phrases) that is generated by the system in response to the entered code. This results in rather poor interface understandability for laymen. The user-friendliness of the system may leave much to be desired, but because all users are (necessarily) specialists, they cope. However, only very few people understand the ins and outs of the coding system, and even fewer know what exact functionality underlies the codings: what computational processes are triggered or influenced by them.

The system's elaborate interfacing functions through over 50 different 'transaction screens' (views on the database), some of them intended for the entering of data mutations, some 'read only'. Possibly the most important screen, and the one central in our investigations at concept level, is 'screen 1201', which shows an overview of relevant mutations concerning 'progress registration and control' of a dossier. Figure 7.1 gives an impression of what screen 1201 looks likes[7]. Screen 1201 is in fact one

[7] We present a translated approximation. Confidentiality prevents us from including a literal representation.

of the more intuitively readable, user friendly screens of RESA/FASA.
Spread out over all screens, over 1000 different codes are used, many of them 'homonyms': identical numbers with a different meaning in relation to different screens. For example, code "01" has twelve different meanings, and in addition there is also code "1", which has 15 different meanings. In documentation, the codes are grouped according to functionality; there are more than 40 main code categories, varying from, for example, "mutation type codes" (48 of them) to "country codes" (signifying 18 different countries by means of 3-digit codes). A limited 'help function' allows users to view simple tables explaining the codes (i.e. providing a short description). This function is not available for all codes in all situations.

7.3 Conceptualisation View of the RESA/FASA MSLC

We now present an analysis of the Gak conceptualisation situation in relation to the RESA/FASA system, in line with the architecture viewpoint for conceptualisation as introduced in Chapter 6. We apply our method for analysis and evaluation of conceptualisation processes, as described in Section 6.3.

7.3.1 DEs, Languages, and Concepts in the RESA/FASA MSLC

Stage 1 of our method concerns the charting of the domain under consideration in terms of discourse environments, and identification of the various relevant languages used in them. Indicated dysfunctionality of language is discussed at the end of this section. We performed an analysis of some particular concepts that are part of domain languages, but for reasons of both relevance and confidentiality those findings are described only briefly and for illustrative purposes, in the dysfunctionality section.

MSLC

The Information System Language Community we focus on centers round the RESA/FASA information system. Though various functionalities of the system might be distinguished (most prominently 'Claim Assessment Progress', 'Client Information', and 'Product Accounting and Billing'), we decided not to distinguish three medium systems explicitly, since for users and describers alike, RESA/FASA is still perceived as essentially one system. Thus, we only differentiate between the RESA/FASA Information System (the whole technical system) and a more abstract component thereof: the RESA/FASA Medium System. The latter includes a specified 'language' (specified set of concepts), which as a sub-language 'belongs to' (i.e. is shared by) three separate main Discourse Environments (see figure 7.2). These DEs will be further described below. As indicated in the diagram, the DEs are divided into two categories, populated by either 'medium system users' or by 'medium system describers'. In DEs in which RESA/FASA *use* is central, the medium system language is indeed mainly involved in operational communication, though it may

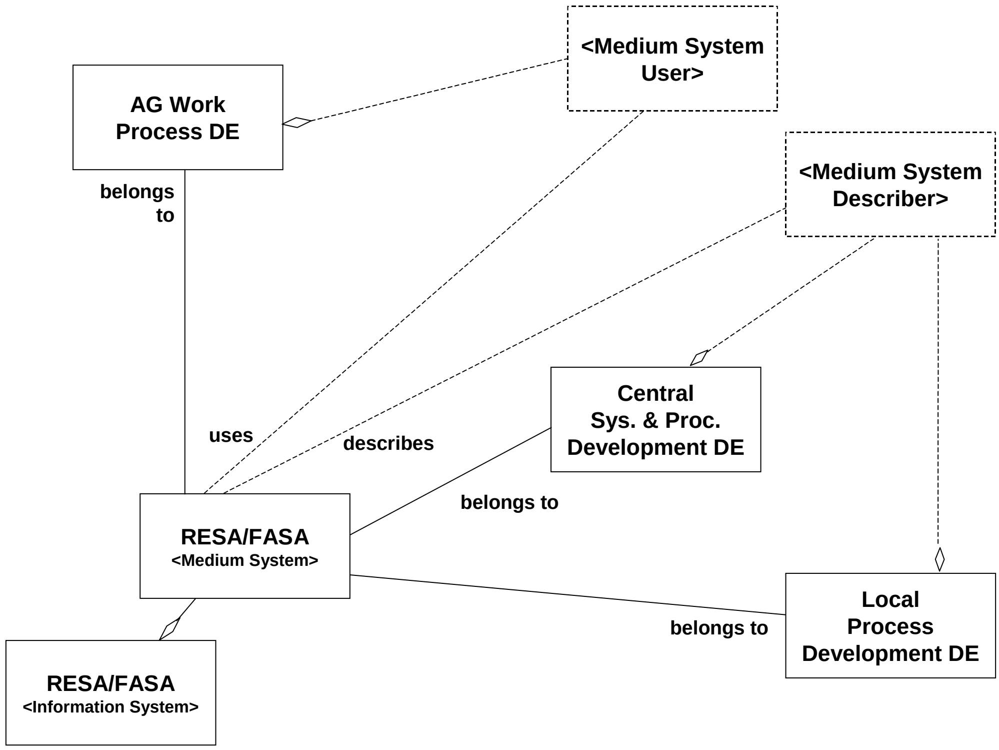

Figure 7.2: 1a: MSLC Diagram for RESA/FASA

also be subject to non-intervening linguistic meta-communication. In the *description* DEs, the medium system language is primarily subject of intervening linguistic meta-communication.

Quite a few distantly related roles and activities are excluded from the MSLC as defined in the diagram. They may well be referred to in more specialised diagrams and analyses, but are considered to be only indirectly related to the RESA/FASA Medium System. For example, we have excluded the LISV product specification DE (related only through 'translation' steps), the System Implementation DE (does not directly involve RESA/FASA concept use or functional specification), and the Central Process Management DE (only uses and describes RESA/FASA in terms of statistics). These considerations are related to *identification and restriction of the MSLC*, and reflect architecture-level choices by the analyst.

In addition, and more importantly, further differentiation of DEs would of course have been possible. For example, we might have distinguished between a 'Claim Assessment DE' and a 'Service Desk DE', on the grounds that they involve different tasks and different roles performing them. However, the Gak-AG/RESA/FASA situation is characterised by the use of one information system by many different people *who work closely together* and who use various terminologies in parallel. This linguistic complexity is better represented by charting a few complicated DEs than by distinguishing many simple DEs, which would suggest that the central medium system can separately handle multiple user languages as used in various separate and unrelated domains and can be changed through various independent and unrelated task domains. At Gak, the contrary is the case. These considerations are related to *recognition of the of the nature and structure of the linguistic situation* which could be called the Gak 'linguistic architecture'.

Language Identification and Description

Five relevant 'languages' (seen as sets of concepts within our general framework) are distinguished that are *described* and/or *used* within the various DEs of the RESA/FASA MSLC. They vary in character, and only two of them are 'closed' in the strict, medium system related sense. They are listed below:

- Dutch legal language, and the Gak extension thereof
- LISV product specification language
- Central process and product language, including central RESA/FASA codes
- Local process language, including local RESA/FASA codes
- Functional specification language

The relation of the languages with the RESA/FASA system and various Gak working environments are depicted in figures 7.3-7.5.
The five languages are described below, in line with the Language Description Template (instrument 1c).
Dutch legal language and the *Gak extension* thereof is the language in which the abstract goal of the Gak organisation is described: the law it is to implement, i.e. rights of Dutch employees to certain social benefits, and explicit restrictions imposed on those rights. Legal terminology is pervasive in Gak systems and processes, but most people working with the Gak systems are not legal experts. The artefacts in which national legal terminology is specified are law books or other national legal documentation; in fact, only a subset of legal terminology is relevant here, namely the subset needed to describe social security law. Though experts at Gak necessarily adopt legal language in line with national legal language, they provide their own detailed interpretation of the language, oriented on implementation of the law. This should be seen as a specialisation and extension of Dutch legal language rather than a variety.
Gak legal language is mostly laid down in the Gak Handbook of Laws and Rules and in the Guidelines (see Section C.3), the latter of which are most directly used to provide a link between the 'production process work floor' and the legal frame to which the process must conform. On the work floor and in the 'staf' departments (Section C.3) that support and supervise it (both locally and centrally), the roles most directly involved with legal language are the legal experts (i.e. people that are explicitly educated for legal functions), the claim assessors, and quality control employees. Other employees use legal terms within their own context, but they are generally unaware of technical matters of 'legal meaning' and 'legal use'. Even the claim assessors have a limited and mostly practical grasp of legal language.

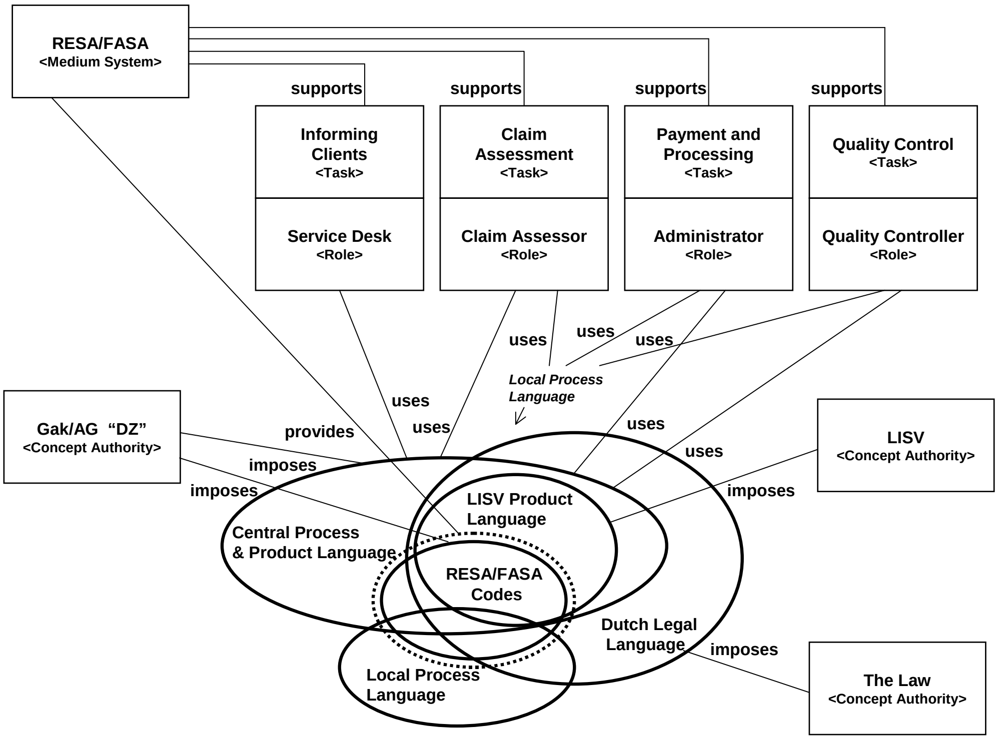

Figure 7.3: 1b: Language Identification Diagram of the AG Work Process DE

National legal language clearly is intended for fairly global use, and yet truly understanding it requires considerable expertise. Gak legal language is of course aimed at a more local use, but even so it is used in 28 local offices plus Gak headquarters and is expected to be used uniformly across all of them[8]. In terms of 'communal linguistic conventions', legal language is most strongly felt to be 'important' and 'ours' among legal experts and people involved in quality control. Other roles acknowledge the importance of legal language, but often consider it to be 'difficult', even 'foreign'. They are not very interested in the ins and outs of legal terminology, only in what they need to know about it in order to do their jobs and stay out trouble with their superiors.

In line with its global character, legal language is often used in communication between DEs, including DEs outside the RESA/FASA MSLC. For example, the LISV is an important link in the 'translation' of law texts to Gak intepretations thereof, so legal language is crucial in negotiations and information exchange between LISV and Gak. It will come as no surprise that legal language is strongly authoritative, at national level but also within the Gak organisation. Everyone is expected to be willing and capable of using the language correctly, at least correct enough to avoid

[8]The creation of the UWV organisation is expected to lead to an even closer relation between national law and the activities implementing it, since they are now all gathered under one centrally controlled umbrella.

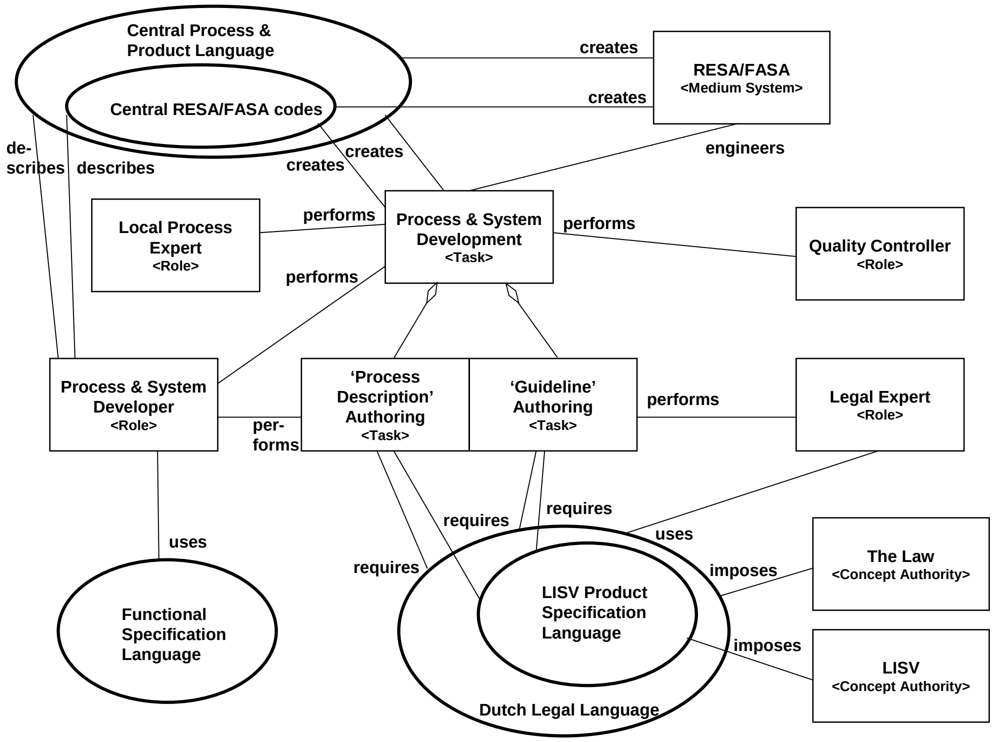

Figure 7.4: 1b: Language Identification Diagram of the Central System and Process Development DE

the occurrence of illegal or damaging utterances/interpretations.

Legal language is typically intended for communicational use. Though quite a few legal terms turn up in the RESA/FASA system, their meaning is only indirectly translated to operational semantics. The persistency of utterances containing legal language is generally quite high: they are intended to be recorded and kept for considerable time (often many years). Within an explicitly legal context, legal language is of course used productively by experts, but apart from that is more often subject to receptive use: people reading it rather than writing it. However, outside a strictly legal context, quite a few cases occur of productive use by people who are not primarily legally interested: claim assessors or even administrators. This is a known source of conceptual problems. In addition, legal terms used in Gak production processes will often acquire a pragmatic load that is detached from its original meaning. For example, if a certain type of benefit is awarded, this has immediate consequences for the work process. For example, certain administrative actions may need to be performed, certain options blocked, etc.; such actions may well take place outside a clear legal context. In other words, non-legal meaning will be attached to legal terms. This adds to interpretational complexity and diversity.

Legal language typically aims for communicational certainty rather than efficiency. As such, it is not unlike formal languages, but it takes a different approach, striving

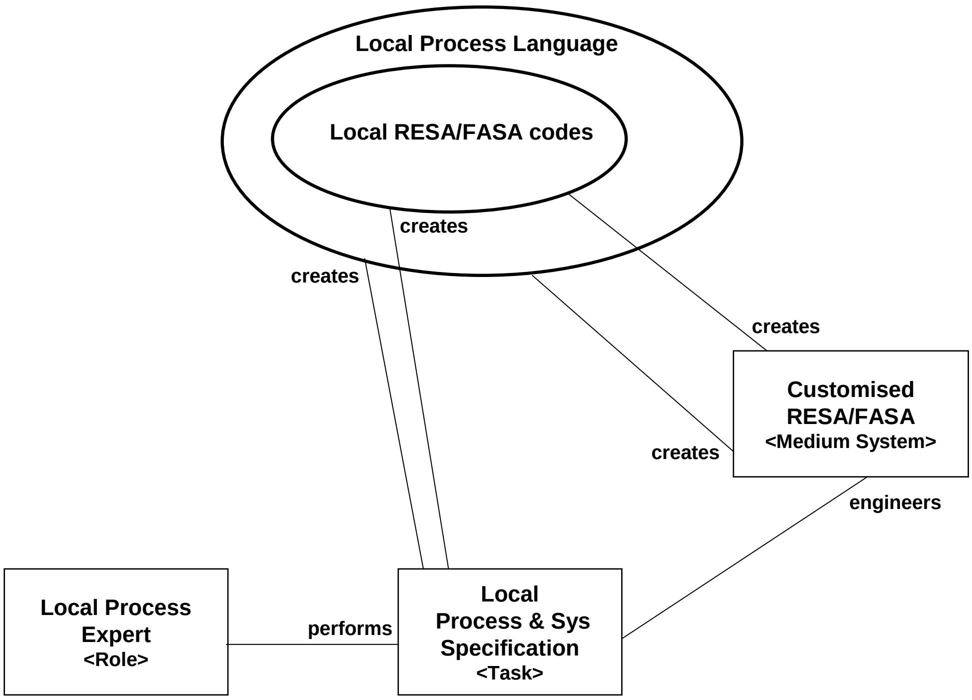

Figure 7.5: 1b: Language Identification Diagram of the Local Process Development DE

to avoid ambiguity by means of descriptive comprehensiveness and detail, rather than a tightly defined syntax and a high level of abstraction. The use of legal terms in an information system like RESA/FASA is unavoidable, but adds greatly to the level of expertise required of users, and increases the risk of misuse and misinterpretation if a user is not legally trained. At Gak, a small subset of legal language is used operationally, but in many cases legal language is translated to operational structures that no longer involve explicit legal terminology: in order to execute a law, it is not always needed to use legal language. Therefore, legal language is also often put to auxiliary use (i.e. in communication somehow supporting the primary communication in a DE).
Finally, legal language is subject to steady change. Laws are occasionally altered, and legal language changes with it. More importantly, crucial descriptive terms from legal language (for example, those specifying a certain type of benefit) are usually the ones that filter through to the Gak claim assessment and payment level. Though changes in legal language are not massive, they occur fairly regularly and always have immediate implications for terminology as built into the Gak systems and used in system-related DEs.
LISV *Product specification language* is closely related to legal language, but is used to phrase decisions and management of law implementation in terms of well-defined products. Thus, LISV product specification language has the function of bridging the

gap between the legal domain and the work process domain (though at a very high level), and dictates part of the Gak process terminology. We consider it apart from legal language because it is much more directly aimed at the production process.
To our knowledge, LISV product specification language is part of legal-style documentation, but admittedly our information on details of LISV product specification is limited. In many respects, its nature and use are linked to and are similar to that of legal language. It is, however, more specialised and less global than general legal language, and as such can be positioned in between Dutch legal language and Gak process language. Most importantly, it is imposed by the LISV as a language authority: the LISV has a contract with Gak, and this contract is largely drawn up in terms of LISV product specification language. The remunerations Gak receives for its services to LISV are therefore completely dependent on descriptions in LISV product specification language. Because by nature it is more closely tied up with work processes than legal language, it is less prone to misinterpretation and misuse in work process context than general legal language. Its flexibility will have to be in line with the changes in legal language, since product specifications necessarily reflect law changes.
Central process and product language (the language used in carrying out the Gak production processes) is of course the chief language involved in RESA/FASA functionality. Product language concepts (either conforming to LISV specifications or Gak-specific) play an important role in it, as do legal terms, but they are incorporated in a language that reflects the detailed coordinated activities that the production part of Gak organisation performs. Gak process and product language may include terms from other languages as discussed, but the interpretation of those terms is often somewhat different. A fair part of process terminology, and virtually all of product terminology, is centralised: it is dictated by authority at Gak Headquarters. However, because the Gak organisation incorporates 28 regional offices that are allowed to organise their own production process, central process and product language is complemented by a considerable body of 'local process concepts'. We will deal with these further on.
RESA/FASA *terminology* is an integrated part of the work process language. Some concepts in the language seem to have originated in RESA/FASA design, and also the system necessarily reflects product names and process terminology. In many cases, the work process was designed in parallel with the system supporting it. The RESA/FASA system makes extensive use of short *codes*, which represent more elaborate terms or phrases. From a decontextualised point of view, these codes are *synonyms* of the terms they stand for: they are different in form, but equal in meaning. When the codes are used in operational communication, they often represent highly condensed, contextualised *utterances*. Codes and terms as built into the RESA/FASA system also influence the work process language, even that used in informal, verbal communication between colleagues. For example, a client may be referred to as a "typical code 253"; this is a completely understandable utterance for someone working with the RESA/FASA system on a daily basis.
Gak product and process terminology is laid down in a fair number of different artefacts. To mention the most prominent ones:

1. LISV Product Specification

2. Gak Handbook of Laws and Rules
3. Product Registration Specification
4. Guidelines
5. Process descriptions
6. Local Code List
7. Instruction material (courses)
8. Personal notes
9. Listed legal, system, and process mutations
10. System description (functional)
11. System user manuals
12. Help screen specifications
13. System Code and tables

These artefacts will be more elaborately discussed further on. Importantly, there exists no concrete, comprehensive repository of Gak process and product concepts. The agents using them are all those working with RESA/FASA; the degree to which some agent (in some role) actually uses an artefact varies considerably.
Central process and product language is intended for global use within the Gak organisation. Because of the number of people involved in Gak/AG processes, its use is reasonably widespread considering its highly specialised nature. It belongs quite firmly to Gak culture, and cannot be easily acquired by an outsider. Consequently, it is subject to fairly well established communal linguistic conventions, which are not easily changed. The language is hardly used in communication between DEs outside the MSLC. It is for a considerable part dictated by official decrees and decisions, but apart from the terms and codes built into information systems like RESA/FASA, the (correct) use of centrally determined process and product language can not easily be enforced, even though disciplinary measures after incorrect use of terminology are not unheard of, and correct term use is an acknowledged and integrated subject of quality control and instruction. The general attitude to central process and product language (across all layers of the Gak organisation) is that it should indeed be a standard, and therefore that employees can be expected to conform to that standard. How this can be achieved is another matter; people acknowledge that it poses a problem. In extreme cases, legal terminological disputes are refereed by a 'Lawfulness Committee', which is not installed especially for terminological matters but includes them in its verdicts.
Just like legal language and LISV product specification language, central Gak process and product language is used for human-human communication rather than for system construction. It is of course an all-important part of medium system language, but plays hardly any role in communication about matters of system implementation. In RESA/FASA, it is part of functional specifications but programmers

are generally unaware of the underlying meaning of process terms. It clearly plays a crucial role in the creation of persistent utterances, being the main language in which Gak client files, reports, and other documentation is formulated. It is also important in informal, non-persistent use. Though receptive use is of course widespread, few terms in Gak process and product language are not regularly used productively, even if only through the entering of some code. However, in some cases it is doubtful whether the entering of a code can be seen as full-fledged term use: in reported cases, codes are sometimes entered without their deeper underlying meaning being clear. The language is very much an operational language.

The balance between efficiency and certainty in central process and product language is a precarious one. Though certainty is something that is, as a matter of course, deemed important in Gak communication, in practice efficiency often prevails, at considerable cost for certainty. The use of codes is a particularly significant example of this. Process and product language is tightly connected to the process, and thus both implicitly and in explicit description carries a relatively heavy pragmatic load: its description is heavily contextualised, both in terms of context-related references and implications and in terms of the communicative and coordinative actions (speech acts) that are connected with the use of certain terms. This makes the language extremely hard to acquire outside the Gak/AG production process domain. Though it is linked to law and product specifications and changes along with them, Gak product specification language is a large and difficult-to-control body of concepts, which tends to be less flexible than Gak practice generally requires.

Local process language is much less well documented than its centralised counterpart, and obviously much more local. We can safely say that all 28 regional offices of Gak have their own distinct 'process dialect'. It became quite clear in our case study that processes and languages differ considerably between offices, and that employees cannot be relocated at another office without re-training. Local process language is outside the reach of centralised control, and though it is formed locally, such formation is hardly the result of clearly identifiable conceptualisation and specification processes. Put bluntly, it 'just happens'. It is heavily embedded in the local Gak communities, and subject to strong communal linguistic conventions. It is known to and touched by local agents only, creating a barrier for communication about local process details with Headquarters. Not surprisingly, it is hardly subject to language authority.

We were somewhat surprised to find that even in a centrally designed and controlled system like RESA/FASA, a deliberately created 'conceptual area' exists for local terms and codes. A range of codes known as the '400 series' is designated for local use: local AG employees may use these codes as they wish, conform their own, local agreements and conventions. The code space is indeed used as such, giving rise to many code systems differing between offices and even between different local teams. These codes are therefore unintelligible for outsiders, which occasionally poses problems, but in general the conceptual freedom it allows is welcomed. The flexibility of the concept range is considerable, given the small number of people that needs to be consulted and/or informed in case of changes. This creates a small conceptual 'escape hatch' that may be used to compensate lack of expressiveness and flexibility of the medium system language at large. However, the locality of code agreements obviously also restricts the general, standard usability of particular codes.

Finally, we briefly consider *functional specification language*, used by process and system developers to communicate functional specifications of (changes in) the RESA/FASA system to ASZ programmers. Functional specification language is positioned in the margin of our case study, because it is not used for operational communication in the work process DE. Nevertheless, functional specification language is used (among other things) to describe the medium system language in an implementable fashion, an activity which is an important part of operational communication in the central system and process development DE.

Though we have not studied functional specification language in any depth, it has become clear that plain Dutch is used as a basis, and occasionally 'pseudo Dutch' (resembling the rigid syntactic structure of programming code) for strongly algorithmic descriptions and prescribed calculations. Functional specification texts are uniformly structured; a Gak standard requires they include an introduction, a problem statement, a description of the current situation, a description of the required situation, and a suggestion as to what technical solution may be chosen. ASZ (see Section C.2) keep a central database of all functional specifications, which can be accessed and mutated by Gak process and system developers.

The authoring of functional specifications in 'ASZ-accepted format' as such does not appear to play an important role in initial medium system language specification; it merely translates agreements reached under supervision of an RMT[9] to a coherent format that is ready for a final transformation to COBOL code and tables. We refrain from further discussion of this language here.

Dysfunctionality of Language

In our case study, we found a fair amount of evidence of dysfunctionality of language, both in the form of explicitly described but general dissatisfaction of language users and describers, and in the form of described language-related trouble with particular concepts and their functioning within the Gak/AG work process. We are primarily concerned with the description of general language dysfunctionality here, but provide some brief examples of concept-specific trouble. We describe language dysfunctionality in the RESA/FASA MSLC at the hand of instrument 1e: the Concept Use Symptom Template.

When asked whether operational trouble concerning terminology ever occurred, interviewees at all levels answered affirmatively, though giving concrete examples was not so easy. For example, consider the following quotation:

> "Yes, terms occur that nobody really knows the meaning of. Term meanings also tend to go and lead a life of their own. [...] In many cases, people think 'yeah, whatever'. [...] They say: 'I work in IMF'. But if you ask what that stands for, they answer something like 'don't know, but that's how I keep track of the result dates'. That's their meaning of IMF". (local Head of Quality Control)

Not unexpectedly, abbreviations, which Gak language is riddled with, are an infamous source of confusion. The most obvious but also most problematic aspect

[9] Release Management Team; see Section C.2.

of abbreviation use is that people use terms without knowing their exact meaning. This often gives rise to *meaning shift*: people making up their own meaning, which may be quite different from the one originally intended.
Asked how problematic failing conceptual knowledge really can be, the same senior quality control interviewee quoted above answered as follows:

> "I think it's quite problematic! For example, we have come across a manager who didn't know the meaning of a central term [an abbreviation]. This led us to the conclusion that he simply wasn't interested. We abandoned the audit; 'you're not on top of the matter, it has a low priority for you'. The plain disinterest bothers me mightily. People turn away from fundamental matters. [Interviewer: "They only get interested when things go wrong?"] Yes, but that's reactive, I prefer pro-reactive. I think that, as a manager, you cannot afford to be only concerned with the everyday bulk of work, you have to understand what it is all about. But those people are being terrorised with information and law changes, they loose the general overview." [...]
>
> [No longer talking about management but about the production process:]
>
> "There are situations where people ignore signals because they just don't know what to do with them. Insufficient knowledge, high pressure, information overload, expectations set too high."

In view of the latter observation, we would like to emphasise once more that working with an information system is very much an activity involving 'languaging'. Ignoring a system signal is ignoring someone speaking to you; not understanding a signal is not understanding what is said; not knowing how to respond is not knowing what to say back.
In terms of explicit concepts, we have found various illustrative cases. For example:

- We encountered a local, free form concept that is generally used in a district office, and which stems from a concept used in a different system a decade ago. It is used frequently, also in database entries, but is not described or defined anywhere, and perhaps because of this it is used in quite different ways throughout the office, giving occasional rise to misunderstandings. It can be 'read' (accessed) nationally, but makes no sense to non-locals –or makes the 'wrong' sense. Interestingly, there is a perfectly good alternative available for the concept in the regular RESA/FASA system, but this is mostly ignored by the system users, who keep using the 'ghost of an old and familiar concept'. Quality control has explicitly reported on this situation, but this has not led to further action.

- Product registration, which is vital for safeguarding correct remuneration for Gak products, can only be performed through the RESA/FASA system. The AG 'professionals' (medical experts, labour ability experts; see Section C.1) do not use this system, but communicate the products they deliver to the claim assessor in their team informally. The claim assessors then have to translate this informal report into the right codes and register the proper products (that, incidentally, may be quite expensive). However, one reported action by a professional can cover a number of related but different products. It is very hard for a claim assessor to translate the informal description (often "one complex sentence") into clear individual product codes as required by RESA/FASA product registration. This occasionally leads to misregistration. Officially, forms exist that should guide the communication between professionals and claim assessors, but these forms are generally ignored.

The examples above concern misunderstandings caused by conceptual unclarity or problematic translation. In other cases encountered, concepts available are well understood but their meaning is too 'rough', leading to insufficient capacity to communicate some necessary detail. For example:

- The Gak system does not enable registration and processing of some types of benefit the client may receive from other organisations than Gak. This registration is formally required in relation to a particular rule (official decree and protocol), but is not supported by the information system. Hence, the decree cannot be followed.

- A particular action code lacks nuance: it covers two officially distinct actions that therefore cannot be distinguished in communication through the medium system. This is mainly the result of the policy to restrict the number of codes, for reasons of conceptual complexity but chiefly for technical reasons (running out of 'code space').

- In some cases, it is possible that a client's dossier is generally managed by one office but a particular product concerning this client is delivered by another office. The 'foreign' office, however, cannot directly register its service (the necessary concept for this is not made available to it). It can only code its involvement in a field disconnected from the registering system. Thus, it only receives remuneration for the product delivered if a human agent at the managing office correctly reads the office code and takes the initiative to register the product accordingly.

It occurs that objections against the availability and/or use of a certain concept in the medium system language are raised, or alternatives suggested. Centrally, such requests are always heard, but they rarely result in medium system language adaptation (for reasons discussed elsewhere). However, if concepts used have legal consequences, matters are serious enough to require timely and decisive action. In case of unclarity or dispute, the 'Lawfulness Committee' decides. We have encountered one concept that did not directly involve the RESA/FASA medium system language, but general use of an official Gak term. Its use was officially objected to by an important professional group (the medical specialists), and the suggested change was carried through.

In extreme cases, concepts may be so dysfunctional that people refuse or fail to use them, and even find 'workarounds' that permit them to communicate what they want without using the concept officially intended for it. We already presented an example above (first example listed). More seriously, the whole RESA/FASA work progress registration system is relatively unpopular. A process and system developer at Headquarters reports that "we suspect that many claim assessors use their own spreadsheets for this [i.e. those RESA/FASA functionalities that can be avoided], which blurs our picture of the situation". One concrete case of widespread and deliberate conceptual disuse has been observed:

- If some action is cancelled, this cancellation is explicitly coded. People generally fail to see the distinction between two of the three codes designated for this. As a result, they tend to simply ignore one and only use the other.

In one case, an old and abandoned concept code has been assigned a new meaning. This did not lead to conceptual problems as such, but initially the code still triggered computational effects in the system related to the old concept, which caused a serious clash with the intended function of the new code meaning. The defect was quickly repaired.
On a more general level, interviewees have expressed dissatisfaction with the level of conceptual standardisation. This is usually linked to the Gak practice that allows local officies to design and implement their own work process. It obviously tends to create some tension between local practice and central product, process, and system design. Everyone acknowledges that this problem cannot easily be resolved, and is an almost inherent problem in every large, centralised organisation, but the problems are indicated nevertheless.
Opposite signals are also heard, sometimes from the same interviewees. Lack of expressive power is reported in relation to a clear symptom: "it can be simply derived from the fact that locally, quite a few independent mini-systems are constructed and used. [...] When it comes to their personal work situation, everyone wants more space for change, their own specific systems, and their own specific processes" (process and system designer at headquarters). The following quite nicely sums up the general attitude towards Gak terminology at production process level: "If people come up left and right with different terms, well, you just cope with it. And also, we're so busy nowadays, there is no time left for discussion. You're happy with what good documentation you have".
Finally, we observe that at Gak as in most organisations with large ICT departments or other large groups of technically oriented people, people at the very least have the impression that "ICT people do not speak our language". Indeed, in the RESA/FASA MSLC, no strictly ICT-oriented agents and languages seem involved, the linking pin between the process world and technical world being 'functional specification language'. However, in Section 7.3.3 we will come across some cases in which technically oriented agents are after all involved in RESA/FASA-related communication –with interesting consequences.

7.3.2 RESA/FASA-Related Conceptualisation Processes

Now we have acquired an overview of the linguistic situation in our case domain, we turn to the second stage in our analytical method: the identification and charting of explicit conceptualisation processes. In the case study, this has turned out as the least fruitful area, since Gak practices include hardly any deliberately selected or

installed processes of the kind we are interested in. Conceptualisation activity is scattered in a rather ad hoc fashion over the various process and system development activities and documentation authoring activities. In addition, it is very much *deliverable oriented*: management of conceptualisation, if we can speak of this at all, focuses on describing the definitive representations that are to be delivered (their global content and form), not the techniques or intermediate steps/representations this may involve. Also, in the core process and system development process, hardly any artefacts are primarily dedicated to the description of medium system language or other terminology. For example, explicit definitions do occur, but they are always part of texts focusing on process description, legal guidelines, or system description. As a result, an interested agent can only hope for some explicit information on a concept to exist, and even then may not be able to track it down since no index or structured retrieval mechanism exists for terminological information. At local level, we found no explicit language specification worth speaking of, hence no conceptualisation. In other words, local conceptualisation is implicit, ad hoc, and language 'just happens'.
Nonetheless, some form of explicit conceptualisation must take place (centrally, that is), given the fact that concepts do end up being built into the RESA/FASA system. We will discuss the few concrete exponents of conceptualisation we have found evidence of, and use them to chart the limited explicit conceptualisation processes in the MSLC.

Conceptualisation Situation

A general overview of the conceptualisation situation in the RESA/FASA MSLC is given in Figure 7.6. Note that the only conceptualisation processes deemed important occur roughly in the Central System and Process Development DE (though documentation authoring for a considerable part occurs in a number of sub-DEs, which are not distinguished in the diagram for reasons of transparency).

Conceptualisation Exponents in the RESA/FASA MSLC

We proceed with a description of the main artefacts identified as playing a role in conceptualisation (either as input or as output). We only include aspects relevant for later evaluation. The descriptions are based on the Conceptualisation Exponent Template (instrument 2b).

LISV Product Specification official specification of products that Gak must deliver. Crucial for production process. Highly authoritative, and the responsibility of LISV.

Legal Documents various legal documents, varying from actual law texts to more specialised Gak documents like the Gak Handbook of Laws and Rules, and the Legal Guidelines. Only the latter might possibly be changed within the RESA/FASA MSLC. Responsibility varies, but always lies with legal specialists and ultimately the general authorities. Legal language prevails. Highly authoritative.

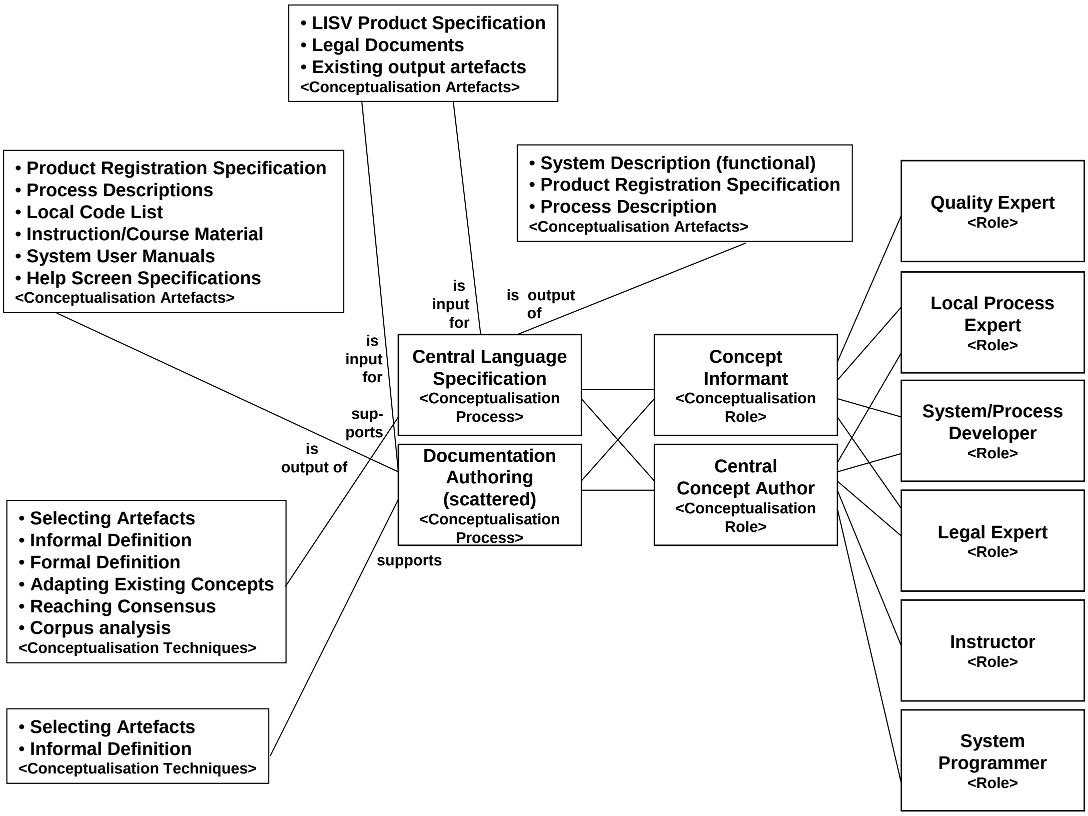

Figure 7.6: 2a: Conceptualisation Situation Diagram of the Central System and Process Development DE

Product Registration Specification a system-related, detailed description of how delivered products are to be registered in the RESA/FASA system. Authoritative. Responsibility of central Gak product authority (DZ), but co-authored with process and system experts.

Process Description central specification artefact for process and system development; serves both as instructive documentation and prescriptive documentation. Authored by process and system development department; responsibility of DZ. Authoritative as a central repository of process information, but should be in line with legal authority.

Local Code List a practical RESA/FASA code list exists: list of system codes, divided in categories, with the meaning of each code provided (a description of 1-10 words). Essentially provides a list of most RESA/FASA concepts. To our knowledge, the compilation of the code list that we encountered did not occur centrally, but locally. It thus is 'a' RESA/FASA code list rather than 'the' RESA/FASA code list. Responsibility and authoritativeness: unclear, low.

Instruction/Course Material derived from more authoritative documentation, yet detailed and written to be optimally readable. Many agents keep using this material once they have it –even if it becomes outdated. Instructors are very well informed, but also tend to develop their own interpretation of terms. Authoritative in the eyes of people following courses, but in fact not very authoritative if compared with more central documentation. Still, good source for linguistic information.

System User Manuals this technically oriented documentation is a side deliverable of ASZ system realisation, and is its responsibility. Some ICT-oriented interviewees claim that system manuals play a role in work process support, but this is contradicted by a number of remarks by non-technical Gak employees. We expect system manuals to be rarely used, and to be only marginally useful for linguistic meta-communication, except perhaps for translation of system codes into 'Gak process and product language'.

Help Screen Specifications the 'helps screens' that are part of the RESA/FASA system are tables that help translate system codes to process and product terminology. We have not found out whether they are 100% generated from the system program and parametrisation tables, but expect this is not the case. Therefore, ASZ system programmers are assumed to play a role in the authoring of this limited but much used form of documentation, which officially has a low authority and for which ASZ is responsible.

Functional System Description this type of artefact supports the main means of communication between Gak system and process developers, and ASZ programmers. It is highly authoritative for implementation of the system (and dictates the actual medium system language as it is realised in the technical RESA/FASA system), but should be in line with less technically oriented artefacts. In this respect, it has low general authority.

We continue with a very brief description of Gak conceptualisation techniques, which as discussed are few, hardly explicit at all, and about which consequently little is known.

Selecting Artefacts this technique is necessarily involved as part of Phase 1-3 activity, but is not expected to be important except for establishing which documents (LISV documents, decrees, specifications, reports) are to be leading in process and system development (including medium system language development). We expect it to hardly take time, and to be mostly guided by implicit yet well established Gak conventions. It can safely be assumed marginal in the conceptualisation process, since it has little to do with any particular *approach* to freezing language.

Informal Definition this is by far the most prominent conceptualisation technique used in the case domain. Note that 'informal' does not mean 'careless': elaborate and precise legal definitions are still 'informal'; 'formality' involves mathematical precision and notation. Informal definitions take considerable time to create, especially if discussion or research is needed. In addition, they require time to be disseminated and to be included in instruction material.

However, the centralised nature of the RESA/FASA system and its legal underpinnings suggest that definitions, once authored to satisfaction, will not frequently require change. Considering the scale the which the technique is currently used, the resources required raise no problems.

Formal Definition strictly spoken, this form of definition is only marginally used in the RESA/FASA MSLC, namely for the description of exact calculations and computational manipulations as part of functional system description. However, in the ASZ technical domain bordering on the MSLC, formal information analysis is at times performed (in context of the old but respectable SDM method: Turner et al., 1985). Though the primary aim of this activity is to make realisation of the technical system possible, it still involves 'user language'. Formal system analysis at ASZ may –intentionally or not– add detail to medium system language specification that *influences* user language, or even *deviates* from language as intended by, for example, legal experts. This is a known side effect of the coupling of 'cognitive' and 'operational' semantics, which can not easily be remedied. Formal definition is quite expensive, yet required for solid system engineering. It can only be performed by specialists.

Adapting Existing Concepts this conceptualisation technique, which should not be confused with the necessary *incorporation* of dictated concepts (related to procurement), is applied wherever extra detail is added to known concepts ('refinement'), or when known concepts are somehow altered. In Gak context, *refinement* of concepts occurs mostly in view of the many authoritative definitions 'from above' that are to be respected yet also interpreted. The most influential occurrences of concept *adaptation* are related to the translation of terminology from laws to products and processes. This in fact happens on many levels, from LISV to local production processes. However, adaptation as part of *explicit* conceptualisation is typically a centralised activity. Refinement and adaptation are quite expensive, but so much part of the Gak development process that they are a crucial and accepted part of process and system development.

Reaching Consensus though it is not part of an explicitly language-oriented development process, nor a well described conceptualisation practice, quite an extensive mechanism is in place at Gak that is meant to safeguard 'stakeholder agreement' with respect to system and process development. It requires much time and concerns many different people, goals, and ideas. When working towards standards, compromises have to be reached. Amidst the complex negotiations and consideration taking place, terminology and choice of words frequently gives rise to heated debate. However, controlled, deliberate action concerning the reaching of agreement about actual language specifications is quite marginal.

Corpus Analysis finally, we have come across an interesting case of concept-oriented management bordering on corpus analysis. Centralised statistical analysis (counting) of the use of codes in the RESA/FASA system is at times performed, and cases are known in which this has led to explicit instruction concerning the use of concepts, mild disciplinary action, and changes in the system. Given that the mechanisms needed for statistical analysis of code use are already in place, it is quite inexpensive and yet may provide great insight into current use of system concepts, potentially resulting in improvements of various kinds. However, much of course depends on the detail of analysis possible. A mere count of codes used may certainly have its merit, but effectiveness would be greatly improved if details were known as to who uses which concepts, and what for. Currently, the counting of code uses cannot be refined beyond distinction between district offices.

In Figure 7.7, the main organisational roles are listed in combination with the conceptualisation roles, language artefacts, and conceptualisation techniques they are typically related with. We refrain from a description of related expertise, since within the Gak case, role-related knowledge could not be analysed in enough detail to be meaningfully distinguished at this level of description. The matrix includes conceptualisation roles as "A" (author) and "I" (informant) in combination with artefacts. The representation chosen emphasises the direct link between organisational roles and conceptualisation at Gak, showing a lack of centralisation and organisation in the assignment of language-oriented roles. Techniques used by organisational roles are simply marked with "X". Note that 'LISV Product Specification' is not linked to any role since it is only used as input for conceptualisation, and is not subject to conceptualisation as such within the RESA/FASA MSLC.

Conceptualisation Processes

Having explored the concrete exponents of explicit conceptualisation, we can turn to a more in-depth description of the conceptualisation processes as they exist in the RESA/FASA MSLC (already introduced in Figure 7.6). We do this roughly at the hand of instrument 2d: the Conceptualisation Process Property Template.
It has become apparent from the analysis so far that explicit conceptualisation at Gak is not a neatly organised, centralised, transparent process. Activities and overlapping deliverables are scattered throughout various organisational areas in a rather ad hoc fashion, while their result is expected to show a high level of standardisation in line with the centralised information system that is at the centre of the MSLC. Instead of trying to chart the somewhat blurred and complex maze of conceptualisation activity in too much detail, we have distinguished two rather abstract main processes; it is hard to clearly and decisively describe neatly separate conceptualisation processes here. The two processes distinguished are presented in figures 7.8 and 7.9.
The conceptualisation process dubbed 'Central Language Specification' is primarily related to the RESA/FASA medium system, but should be in line with Gak process design. With respect to medium system language, it chiefly involves selection of concepts and the forms (words, codes) representing them (phases 4.1 and 4.2), yet mostly *without explicit meaning description* (Phase 4.3). Little or nothing is known about explicit activities belonging to phases 1-3 at Gak; as observed before, the nature of conceptualisation is mostly *deliverable oriented*. The resulting deliverables are the central, most authoritative AG documents plus the functional description of the system. The development process in general is quite centralised (DZ-AG) and well

	Quality Expert	Local Process Expert	Sys./Proc. Developer	Instructor	Legal expert	Sys. Programmer
LISV Product Spec						
Legal Documents					A / I	
Prod Reg. Spec.					A / I	
Process Descriptions	I	I	A / I	I	I	
Local Code List		A / I				
Instruction Material	I	I	I	A / I	I	
Sys. User Manual						A / I
Help Screen Spec.			I			A / I
Functional Sys Descr.			A / I			I
Selecting Artefacts			X	X	X	
Informal Definition		X	X	X	X	
Formal Definition			X			X
Adapting Concepts			X		X	
Reaching Concensus	X	X	X	X	X	X
Corpus Analysis	X		X			X

Figure 7.7: Chief Roles involved in Conceptualisation in the RESA/FASA MSLC

managed; the conceptualisation effort that is part of it is implicit but substantial. Expertise required is channelled reasonably effectively, and prescribed authoritative documents are used as input.

Central language specification is closely linked with system releases as authorised by the Release Management Teams and, ultimately, by the head of DZ-AG. Mechanisms for reaching agreement and testing are in place, though they are not always optimally functional. In particular, though in principle users can exert influence on medium system language based on their normal working experience, in practice this is hardly ever the case except in large ex ante (re)design efforts. Flexibility of medium system language is undeniably quite low. Conceptualisation is not very efficient, and the process requires considerable expertise and resources. In view of the grave legal consequences of incorrect language use, the freezing of 'the right' language forms is deemed important, but unavoidably there is a terminological legacy due to many law changes and related language changes, which seriously complicates matters and hampers changes.

As mentioned, concept *selection* and *labeling* is mostly covered by the central lan-

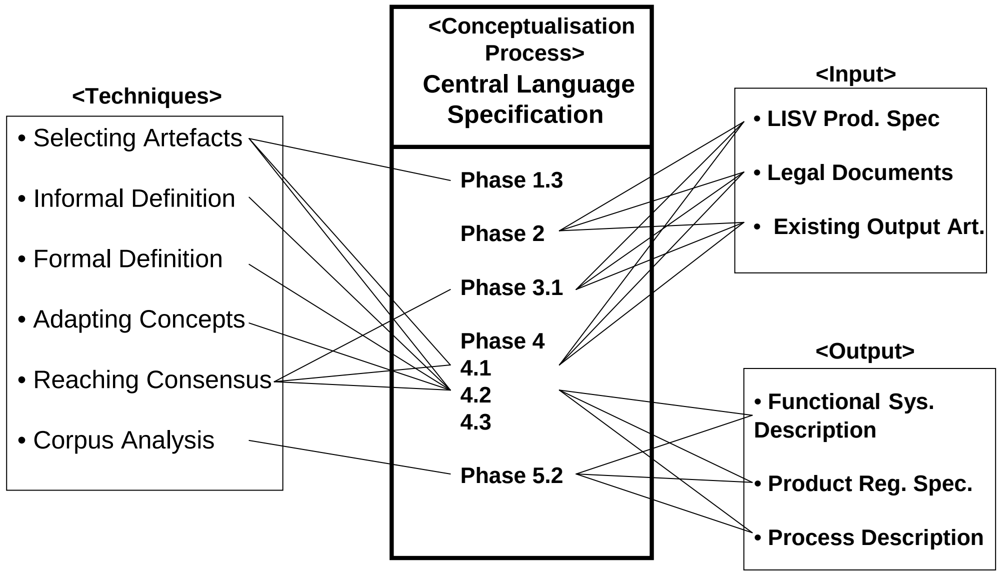

Figure 7.8: 2c: Detailed Conceptualisation Process Diagram of RESA/FASA Central Language Specification

guage specification process, which is indeed a fairly well-demarcated conceptualisation process. Importantly, matters are different concerning the 'Documentation Authoring Process', which mainly concerns *meaning description* of RESA/FASA medium system language (hinging round Phase 4.3). It is merely a bundle of indistinct activities, carried out by various roles, under various authorities, rendering meaning descriptions that are scattered across many different sorts of documentation, which together show considerable redundancy and the coherence of which is not systematically guarded. Though considerable effort is put into documentation authoring, the language component of it is hardly ever addressed directly.

As a result of insufficient availability of centrally distributed information concerning medium system language, alternative 're-conceptualisation' initiatives are taken locally: concepts are de-centrally 'charted' in order to keep on top of them (the Local RESA/FASA Code List is a fine example). This increases heterogeneity of language documentation and the chance that meaning description goes awry somewhere. Finally, documentation is mostly seen as a derivative of the central language specification output (process description, product registration description, functional system specification). In terms of required correctness and usability, it is considered secondary to the actual, implemented medium system language. This is unfortunate in view of the fact that documentation is the only source for non-intervening linguistic

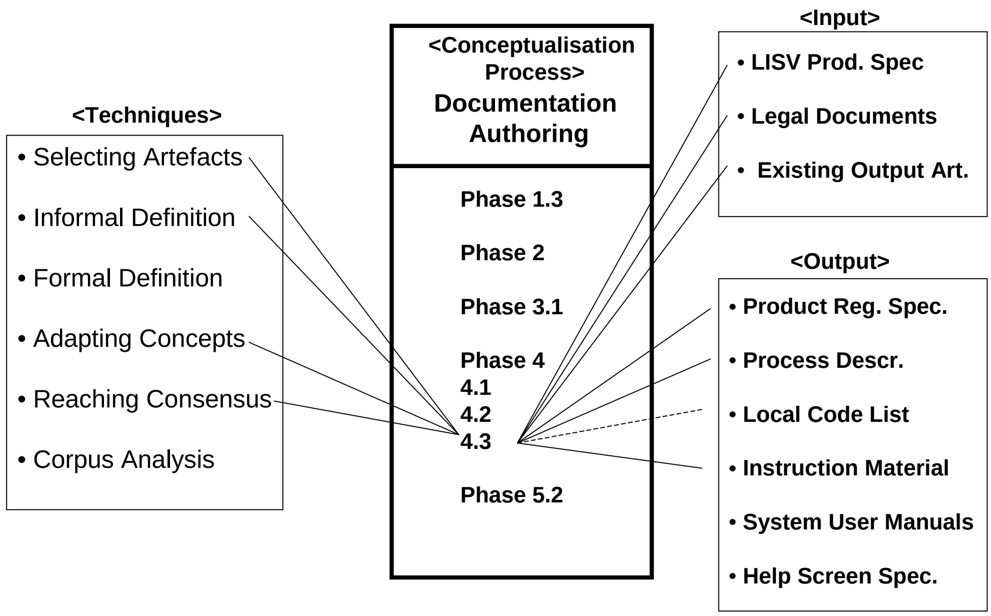

Figure 7.9: 2c: Detailed Conceptualisation Process Diagram of RESA/FASA Documentation Authoring

meta-communication that is potentially centrally controllable (because it is mediated). In other words: documentation helps give controllable meaning to the often enigmatic terms and codes that are part of the medium system. We will elaborate on this in Section 7.3.3.

Trouble with Conceptualisation

Based on the Conceptualisation Symptom Template (instrument 2e) we briefly consider what generally ails Gak-AG conceptualisation processes. Not many explicitly voiced complaints about conceptualisation activities have been encountered in our cases study. We expect that this is largely due to the plain lack of conceptualisation being explicitly performed or distinguished (awareness, recognition). We observe that the often expressed wish for more elaborate and systematic terminology description (e.g. the construction of a Gak glossary or a central Gak data dictionary) remains unfulfilled mainly because there is no time and money for it. Flawed conceptualisation in the domain studied is mostly apparent through flawed *results* of conceptualisation.

Summary

To sum up the main findings concerning explicit conceptualisation that have emerged:

- Conceptualisation is deliverable-oriented and takes place mostly on an ad-hoc basis. It is simply not something that is part of Gak culture.
- Very little explicit conceptualisation support (methods, techniques, tools, approaches) is prescribed for Gak system and process development practice, and even less is actually applied. Various conceptualisation phases remain implicit, i.e. are skipped or 'intuitively' performed. It is possible that we have missed out on some isolated conceptualisation activity that does take place, given the limitations of our case study, but methodically speaking, conceptualisation techniques are not on the Gak map.
- Selection of concept forms, even though implicit, is reasonably well-structured yet mainly takes place centrally and as part of large, ex ante, project-like system changes (usually triggered by law changes), mostly excluding incidental, locally oriented concept selection. Centralisation and standardisation works fundamentally against locally initiated conceptualisation and central ex post conceptualisation.
- Meaning description and concept selection are largely separated as processes. This is damaging to the general coherence and quality of conceptualisation.
- Meaning description as an activity is scattered across a fair number of different roles, sub-DEs, authorities, and –most importantly– deliverables.

7.3.3 Linguistic Meta-Communication in the RESA/FASA DEs

Languages and conceptualisation related to the RESA/FASA MSLC are elements in the larger context of *linguistic meta-communication*: communication about language. Insight in the role that conceptualisation plays in meta-communication is of fundamental importance to our insight into requirements of conceptualisation (i.e. for Freezing Language). Building on our findings thus far, we proceed by charting meta-communication processes in line with Stage 3 of our method. We will try to avoid repetitive discussion, and therefore will emphasise additional findings instead of comprehensively reviewing previous findings in a meta-communication context. However, some overlap with previous descriptions is unavoidable, in particular in diagrams.

General RESA/FASA Meta-Communication Processes

It comes as no surprise that the linguistic meta communication situation in the case domain is highly complex, involving many roles using various information sources and change options, and in addition some roles that edit sources or execute change options. The graphical depiction of the situation in Figure 7.10 is not so much

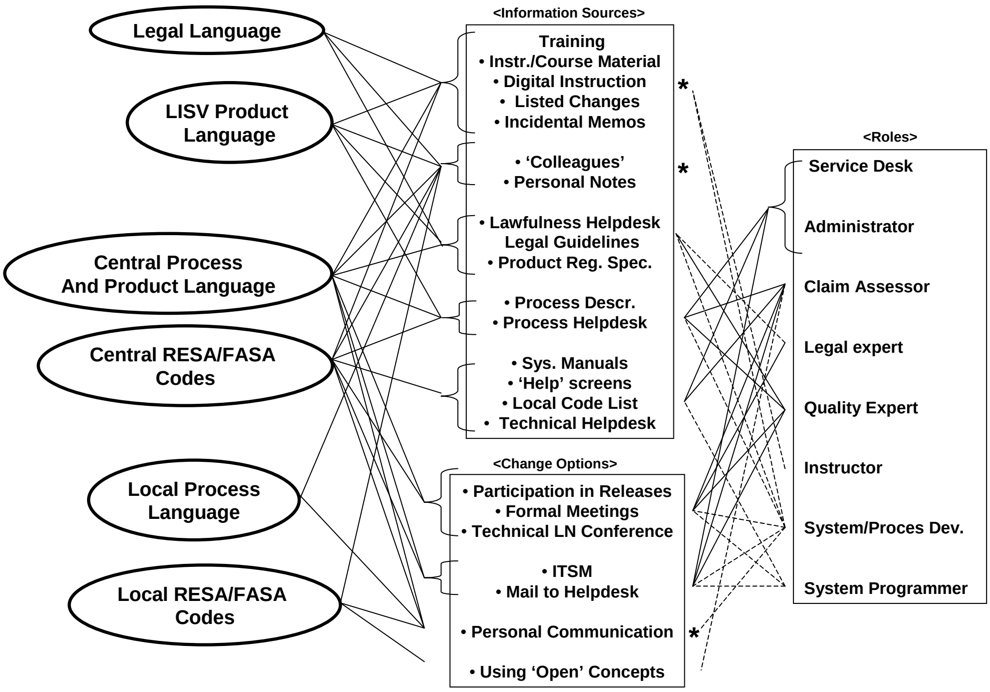

Figure 7.10: 3a: Linguistic Meta-Communication Process Identification Diagram of the RESA/FASA MSLC

intended to present details as to give an impression of the complexity[10]. Even in our somewhat simplified diagram, no less than 33 meta-communication processes are listed for use of information sources and change options, and that is discarding the fact that most sources provide information on various different language categories; 11 meta-communication processes are distinguished concerning the editing of information sources or execution of change options.

Recall that by nature, the use of information sources involves non-intervening meta-communication, and change options involve intervening meta-communication. Ex post meta-communication is problematic in our case domain, given the centralised and inflexible nature of the RESA/FASA system (i.e., the mainstay of linguistic meta-communication is ex ante). The presence of various concept authorities (i.e. authoritative terminological standards) strongly encourages mediated linguistic meta-communication; explicit non-mediated meta-communication appears to be very rare at Gak (as opposed to implicit non-mediated meta-communication). We focus mainly on language-related meta-communication, but will repeatedly touch upon its implementation-related counterpart.

[10]It was necessary to introduce some simplifying notations: some information sources, change options, and roles are collected by means of a bracket, and sources and change option that are used by all roles are marked with a "*".

Instead of exhaustively describing all these meta-communication processes, we observe that plainly, there is an awful lot of them, which suggests a considerable variety of meta-communicational requirements in the MSLC and possibly a need for better meta-communication channelling. We select some interesting and significant cases for a brief discussion.

Some Relevant Meta-Communication Processes

The discussion of significant meta-communication processes below is guided by the Linguistic Meta-Communication Process Description Template (instrument 3b). We centre our discussion on the *use* of information sources and change options. We discard editing and execution, which has generally been covered in Section 7.3.2, and will receive some additional attention in Section 7.3.4. In view of our evaluation interest, we pay special attention to aspects that are related to flawed meta-communication. Where possible, we generalise over similar or collected processes.

Use of Derived or Incidental Documentation (includes training and course material, digital instruction[11], listed changes, and incidental memos). This very general type of linguistic meta-communication process is both quite powerful and, potentially, linguistically quite out of control. It is a typical ex ante, non-intervening, mediated, 'mass consumer' form of meta-communication, that can make or break linguistic knowledge concerning particular concepts. It may be called on by a public that may have quite varied and even changing needs. Its sources tend to combine scattered conceptual information with respect to various MSLC languages, and generally do not support structured querying on specific concepts. With digital instruction as a possible exception, source artefacts involved may easily become outdated but nevertheless tend to stay around and available for consultation. In addition (and this includes digital instruction), they are *derived* from primary sources, which introduces the risk of misinterpretation or misrepresentation by the documentation authors. To keep derived linguistic information sources conceptually coherent and up-to-date, quite an effort is required, including appropriate conceptualisation. However, many problems can be avoided by including standard references to a centralised, well-maintained source for meta-communication.

Personal Communication (including informal informative questions to colleagues and personal notes). This is a highly flexible and efficient form of meta-communication as long as it does not concern precise and authoritative information. The advantage over 'derived documentation' is that personal meta-communication is practical, but also markedly informal and not conforming to standards. In other words, it has no pretense of central authority. It can of course render false information, but if it does this will not

[11]Digital Instruction is a relatively new medium at Gak-AG, combining process descriptions and legal guidelines in a product-oriented instruction that is available on the Gak intranet. It is a marked improvement on previously available documentation, but it has not yet reached the status of 'authoritative artefact' (as held by guidelines and process descriptions); it is still very much a derived product. It is, however, a fair step in the right direction concerning centralised, up-to-date, and comprehensive instruction of general AG work practices.

come as a complete shock to any agent involved. In addition, verbal communication (face-to-face, telephone, informal mail) enjoys the full benefit of natural *flexibility of language*: communication is optimally tuned to the pragmatic situation. It encourages the fruitful combination of discussion-level and discourse-level meta-communication (see Section 3.4.1). It is typically ex post, but can be both non-intervening (information source) and intervening (change option; see below). It may be both mediated and (as an important exception to most other meta-communication processes discussed) non-mediated. At Gak, we have received clear confirmation that informal, verbal communication about concepts is an important communicational lubricant. In combination with well-controlled, authoritative, and accessible concept documentation, it is unbeatable.

User Documentation from the Technical Domain (includes system manuals, help screens, and the technical help desk). Engaging in meta-communication concerning work process language using sources with a technical background is generally thought to be a very bad idea indeed; in relation to this observation, some very strong statements have been recorded. The viewpoint and perception of technically oriented people is found to be so different that it is bound to cause conceptual problems. Generally, technical people admit this, though not enthusiastically so. However, various communication processes and channels do invite this sort of meta-communication. This is partly due to the difficulty that arises in separating 'language functionality problems' of the medium system from technical problems, but also to the cultural remnants of a past situation in which all electronic distribution of information at Gak heavily involved ICT specialists. Before the widespread use of the internet, Gak used its own network not only for purely technical matters but also for *any* centrally distributed documentation. The ICT department thus served as the general 'mail box' for much of the communication with Headquarters. Though linguistic meta-communication using technically oriented sources is now popular nor frequent, it is still a possibility. This represents a modest liability.

Project-Like System Changes (includes formal requests and meetings, RMT participation, and related activities; also see Section C.2). Achieving change of centralised systems at Gak is no mean feat. Because of high demands on quality and standardisation, and because it involves many different stakeholders, changes are implemented carefully and slowly, at considerable expense. In addition, there is a shortage of resources (time, money, people), and law changes draw away most resources and keep many other desired changes off the priority shortlist. We will refrain from further discussion of this dinosaur of a meta-communication process here, but observe it is quite a long way from the intuitive, 'run-time' language changes that characterise 'open' language. All but one of the change options in fact relate to attempts to stir the dinosaur –with little hope of achieving actual adaptation of the medium system language, or even medium system language documentation. Its ex ante status is almost proverbial; its intervening nature clear, but note that to try to intervene is not the same as achieving change.

Official Digital Messages (includes mails to various helpdesks and service centres, ITSM[12] calls, and groupware use in the higher-level technical domains). Some dedicated digitised communication channels are available at Gak-AG to communicate remarks and requests about system and process functionality and technical matters to central points, that gather, sort, and filter messages and either respond to them or send them on to the appropriate experts. In addition, some technically-oriented communication is supported by and structured through standard 'groupware' (based on Lotus Notes or similar applications). The ability to structure messages according to topic, and thereby also to get an impression of how many people are concerned with a particular topic, is considered valuable. A dedicated mailing system that allowed for such structuring was decommissioned because of high cost, and was replaced by regular (unstructured) e-mail; helpful functionality was thus lost, which is deplored. With the exception of groupware used in well-organised release-oriented projects, the digital message services are employed for ex post, intervening, mediated meta-communication. However, messages are mostly channelled straight into the release processes, and consequently many perish due to strict prioritising –which is of course not a matter of communication but of policy and capacity. Since technical and functional matters are kept apart, there appears to be some distinction between language-related and implementation-related meta-communication, but the distinction is not always clear.

Personal Communication (includes verbal communication, telephone communication, personally addressed e-mail, etc. outside regular and official group meetings) we have already mentioned individual colleagues as an important source of conceptual information. However, also as a change option, personal communication is frequently used. In many cases, it will be followed up by more official intervening meta-communication (e.g. some matter will be put on the agenda of a meeting), but there are also cases in which through personal communication, shortcuts can be taken in getting a change effectuated on the actual technical level. The interviews have shown that experienced employees "know their people", i.e. maintain warm relationships with key players at various levels and in various Gak departments. This 'informal circuit' of course occurs in many if not all organisations. However, at Gak it is used extensively simply because it is one of the few means available for *effective* ex post meta-communication.

Using Open Concepts (involves the few cases in which RESA/FASA codes and related concepts can be freely chosen and used). A limited but interesting meta-communication process within the RESA/FASA MSLC is that concerning the locally determined use of a specifically designated range of codes known as the '400 series'. These action and signalling codes may be assigned meanings at will. Consequently, they are meaningless at a central level. Use may be completely individual, but usually agreements about this code use are made

[12]The IT Service Management system or ITSM is a dedicated mail system that stems from pre-internet days. It falls under responsibility of ASZ (a strictly technical party), and is used for messages concerning both functional and technical problems, which are distinguished and channelled differently. However, ASZ is the primary party at the immediate receiving end of ITSM.

within local teams and departments. The availability of open codes provides a limited but much appreciated patch of open conceptualisation space, that is gratefully used by local Gak employees. Conceptualisation concerning assignment of local codes is implicit and ad hoc, and is driven by local work processes and culture. We have encountered similar semi-organised open concept use in various MSLCs bordering on that of RESA/FASA. A process and system developer at Headquarters and some other 'staf' officials repeatedly emphasised that many local initiatives complement and, at times, circumvent centrally 'frozen language' medium systems.

Trouble with Meta-Communication

In the above descriptions of linguistic meta-communication, a number of observations and statements were included concerning meta-communicational dysfunctionality of one kind or another, reflecting the questions included on the Linguistic Meta-Communication Symptom Template (instrument 3c). We will now add a few general remarks, summarise, and generalise about flaws and risks of linguistic meta-communication in the RESA/FASA MSLC. We have boiled down our general observations to 4 main points.

Lack of Efficient and Coherent Non-Intervening Meta-Communication
Information sources fail to serve all but the most basic needs for ex post, non-intervening linguistic meta-communication, i.e. they fail to provide correct, clear, to-the-point, and well retrievable information on system concepts and related terminology. Documentation includes scattered fragments of language description, the coherence, actuality, situational applicability, and general quality of which are doubtful in a fair number of observed cases. Also, those concepts used or described can hardly ever be traced to their origin or to some concept authority, which makes further inquiry difficult. All this entails risks in view of the general (and sensible) Gak policy of centralisation of system and process language (often linked to national legal language). In addition, it seriously hampers knowledge sharing with new employees. Efficient access to good conceptual documentation can greatly improve the ease with which they can acquire Gak language (in particular, by the checking of specific definitions and uses of terms as they encounter them –ex post), which is a key factor in learning. Various senior Gak officials named this specifically as an urgent problem.

Meta-Communication through 'Technical Agents' should be Avoided
With respect to both information sources and change options, there is a policy of rather strict separation between communication about functional and about technical matters. Even so, various communication channels (also used for meta-communication) are used that violate this separation. Most importantly, communication about strictly functional matters at times still actively involves human agents with a strongly technical background (system helpdesk workers, mail and message assessors, system documentation authors). This is observed to cause incidental conceptual problems. It is partly due to a technological and cultural heritage (existing medium systems,

habit, politics): in the past, said separation was less strict, and in some cases non existent.

Ex Post Meta-Communication has Little or No Effect It is almost impossible to perform effective ex post intervening meta-communication (i.e. resulting in actual medium system language change). This is a direct result of the large, centralised approach to ICT fundamental to the Gak organisation, and also of plain lack of resources (priority of ex post changes is low). Clearly, these causes are general and not communicational in nature, but they hamper ex-post intervening linguistic meta-communication nevertheless. Many calls for change are in fact heard and even found reasonable, but yet cannot be responded to. This frustrates locally initiated initiative to improve the support of complex local work processes.

Local Language is Impenetrable Local language is not documented and is largely impenetrable for outsiders (other local offices, headquarters). While this is a common and predictable phenomenon, we observe that it is indeed evidently reflected in (lack of) possibilities for linguistic meta-communication, and that parties advocating centralisation see it as problematic if unavoidable. In the mean time, local system and process languages flourish, at times penetrating 'open' conceptual areas of centralised systems, but much more frequently as part of local 'rogue systems'. Both the languages and the systems they are part of are outside the view and reach of centralised authorities, who worry about this but cannot prevent it.

7.3.4 Implementation Situation of the RESA/FASA Medium System Language

Stage 4 of the method is limited, and requires no more than a general consideration of relevant aspects related to technical development of the 'medium machine' that in combination with a 'medium language specification' creates the medium system that the RESA/FASA system is when viewed from a functional linguistic perspective. This section is roughly structured along the lines of Medium System Development Template (instrument 4a), and also implicitly covers matters as reflected by the questions of the Medium System Development Symptom Template (instrument 4b). Once more we attempt not to be too exhaustive or repetitive; we introduce some new material and in addition briefly summarise some relevant matters that have been discussed before (in a less explicitly technical context).

Data Structure Analysis and Design

At the technical end, there comes a point when functional specification (including medium system language specification) has to be translated into symbols that have operational meaning and are in fact intended for building a machine. Though functional specifications are precise and are aimed at unambiguous communication between Gak process and system designers, and ASZ programmers, they are not specific enough to be translated 1:1 to the formal structures required for the construction of software. This final step is, to a limited degree, supported in the technical domain with traditional information analysis, which aims for formalisation. However, though the activity of information analysis is recognised and incorporated in ASZ and Gak system development process descriptions, no actual method is used for this, at least not consequently, even though the Yourdon method is mentioned (and related use of ER diagrams and data flow diagrams), and also SDM as a more general methodic framework. Formal information analysis is an expensive activity. In this vein, one senior technical ASZ interviewee remarked that "no real methodic approach is used, nor has there ever been an attempt to introduce it, the main reason being that using it would take up too much developing time". In other words, methods are seen as increasing the cost of information analysis, not decreasing.

Since elaborate and detailed functional specifications are already available and need to be followed as closely as possible (though we have reason to believe that basic legal texts are at times also consulted as a backup in this process), only marginal change in medium system language can be allowed to occur. Nevertheless, formalisation may have its consequences for conceptualisation –as has been previously discussed. Limited expertise of ASZ people concerning Gak processes and language entails a certain risk here, in line with the point concerning 'technical agents speaking a different langauge' as made in the previous section. Thus, in medium system implementation as well as in medium language description (documentation), the wish has been expressed that technical agents should be given specifications that are detailed enough for them to work with without having to engage in active conceptualisation.

Another aspect of data structure analysis and design is that it may be subject to standards. For example, some data exchange takes place between Gak and other organisations like the National Tax Office, for which some standardisation is indeed required. LISV also dictated some standardisation. Gak-internally, however, there are few other standards than those mentioned, which are imposed externally. We have been reliably informed that in a mainframe-supported past, this used to be different. Data consistency checks used to be possible but have been abandoned. Measures are being taken to try and capture all system specifications under one coherent framework again, using more contemporary methods and languages (for example, the UML). The result should be closely related to the as of yet fictional 'Gak Data Dictionary' (see Section C.4). The standards and specifications are described at a technical level that generally goes beyond term labeling and meaning description, involving character level specifications and restrictions (for example, the number of characters that a 'Name' field is standardly assigned).

Interface Design

As far as the RESA/FASA system is concerned, we can be brief about user interface design. The concepts underlying the code system may be complex, but the code system as such is straightforward and limited and very tightly coupled with the database. The basic mechanism makes use of database transaction screens. No terminologically relevant 'translation layer' between the database and the transaction interface comes into play, with possible exception of the position of terms and other data items with respect to each other (arguably a matter of two dimensional semi-linguistic syntax). As for machine-machine interfacing, it is for the most part covered by data structure standardisation as discussed in the previous paragraph.

RESA/FASA Technology and Flexibility

As is to be expected given even the current state of the art in computational technology, there is limited capacity for change of the RESA/FASA system, yet from a purely technical perspective, there certainly is some room for adaptation of the system's functionality and conceptual load. Tables that can be defined by Gak and ASZ IT specialists allow for limited but useful change of codes, parameters, and even bits of text while the 'technical system' as such remains untouched. Lacking compatibility between technologies poses a greater challenge, but this is no main problem in a strict RESA/FASA context. However, we have also come across clear signals that technical margins have been running out; recall the limited ability to introduce new RESA/FASA codes, resulting in re-use of old codes (Section 7.3.1). In addition, many codes are set to trigger computational actions, which has over the years resulted in a rather opaque situation concerning triggering relations and cause-and-effect chains[13], making code changes hazardous. The senior ASZ interviewee quoted before has said the following concerning this situation: "Old systems may still be quite flexible in principle, but only if people know how they have been put together. And both with us and at Gak, no people are left who do. So it is not the technology as such that is to blame but *the lack of information about the system*".
In relation to the above (and related to the observed lack of possibility for successful ex post intervening meta-communication), it is observed that lack of flexibility of the RESA/FASA system is for an important part caused by the procedures followed in system development (system releases every three months, as previously discussed in Section 7.3.2). Consequently, hardly any system changes and no local customisation of any kind can be provided for users of the RESA/FASA system.

Resources and Priorities

Finally, we briefly point out once more that general resources for the implementation of system changes is limited, and that most of them are spent on carrying trough law changes. This is a strongly prohibitive factor that has immediate consequences for system flexibility, including possibilities for successful ex post intervening meta-communication, and even for many cases of ex ante intervening meta-communication.

7.3.5 Evaluation of RESA/FASA and Gak Conceptualisation

Having gathered all relevant information that lies within our current reach, we can now evaluate the RESA/FASA MSLC conceptualisation situation at large by applying the matching points that belong to Stage 5 of our method. This will complete our view of how language is frozen in the Gak domain, how successful or problematic this is, and what causes the problems that arise. We will also suggest some potential improvements in the general situation, though this lies outside the scope of our

[13]The term 'spaghetti' springs to mind here.

actual method, which is primarily intended as a descriptive and analytic aid. The suggestions are, however, directly related to our analysis, and if nothing else function as a good illustration of the sort of approaches, processes, and techniques that might be involved in controlled and optimal large-scale freezing of language at Gak.

Matching Functionality of Concepts with Operational Communication Requirements

Legal language and LISV product language AS USED IN THE AG WORK PROCESS DOMAIN

Purpose: authoritative and clear specification of products that need to be delivered and rules that need to be followed, holding across the domain of Dutch social security implementation. Triggers disciplinary action if violated; can be damaging if used incorrectly or irresponsibly. Thus needs to be standardised and open to rather limited differences in interpretation. Used strictly communicational; utterances are generally meant to last. Contextualisation at discourse level should occur within the general frame of social security law, not locally.

Match: languages under analysis match the purpose reasonably well, but the specialist nature of legal language and LISV product specifications causes some problems. Lower educated, administrative staff will lack the expertise to really understand all concepts, which may cause misunderstanding and misuse.

Situational factors: the law and LISV product specification are ever-present aspects of what Gak does. Agents primarily involved in the operational production process cannot be expected to acquire legal expertise beyond a basic level.

Improvements: non-experts should not have to deal productively with legal-like terms they do not thoroughly grasp, i.e. not actively perform actions that somehow require deeper understanding of legal terms. If they have to anyway, they need to be supervised (which is indeed the case). A very ambitious option is 'translation' of legal terms to concrete process actions in the work domain (including its medium systems), which may include terms that are similar to legal terms, but essentially have different, work process related meanings that hold strictly within the Gak production process domain. The latter cannot be expected to be achieved within current possibilities at Gak, since it would involve radical medium system re-design.

AS USED IN THE CENTRAL SYSTEM AND PRODUCT DEVELOPMENT DE

Purpose: similar.

Match: when used by specialists, legal-like language generally works fine. Its standard status for communication with other legally oriented DEs (within and outside Gak) holds, and is functional. However, as an integral part of the development process, medium system language to be used in the work process DE is specified; this is where a mismatch occurs, but it is a meta-communicational mismatch, not a operational language mismatch.

Central process and product language; central RESA/FASA codes AS USED WITHIN THE AG WORK PROCESS DE.

Purpose: supporting the execution of the general (centrally dictated) part of the primary Gak/AG production process. It enables coordination of actions of and among agents, and the recording and communication of any relevant information. Key terminology in the language needs to be authoritative within the production process, and strongly centralised (implying standardisation). It is also a chief component of the RESA/FASA medium system language, hence largely *closed*. Utterances are meant to be persistent, and are necessarily heavily embedded in the work process context.

Match: generally, the language suffices, it does the job. However, using it proficiently demands a high level of specialisation of the people using it, which is reached through training and instruction but also, and chiefly, through experience. Unfortunately, expertise is at times insufficient. In addition, once acquired the language tends to become 'fossilised'; it even becomes slightly socially marked ('our language'). Consequently, people have some trouble (in attitude as well as practically) with changes in the language introduced by central authority. The language is also particularly unfit for use in communication with external DEs (on both social and practical grounds). This poses a classical problem for the service desk employees (who have to inform clients in understandable language) and also for communication with other organisations in general. As for RESA/FASA medium system language: its heavy use of codes introduces ample extra opportunities for miscommunication, but it seems to somewhat improve efficiency. However, its use seems to be tied most closely to the nature of the system interface. The codes are a typical exponent of the 'practical factor'. As a result of its closed nature, RESA/FASA medium system language shows clear signs of not perfectly fitting actual work practices (an observation that lies at the core of our thesis subject), though this mismatch should not be exaggerated. Problems do occur, but are not paramount. Some additional mismatches concern *availability* of medium system language for communication between physically separated but otherwise similar DEs, e.g. district offices communicating with each other by means of the system.

Situational factors: Gak process and product language cannot easily be simplified, nor changed. It is largely dictated by law and legacy; the need for expertise and specialisation is real and unavoidable. It involves a very large group of people with diverse goals, centrally governed yet geographically and (to some degree) culturally divided. Medium language system change is slow and cumbersome.

Improvements: a campaign to simplify and clarify Gak product and process language might be initiated. Better separation of process language and (legally oriented) product language might be encouraged. Introduction of 'local customisation' and operational (system-supported) translation between centralised and local languages might be attempted, but seems very difficult. It goes against general Gak policy and the nature of the organisation.

Local process language; local RESA/FASA codes AS USED IN THE AG work process de

Purpose: supporting the execution of the local, decentralised aspects of the Gak production process. It helps coordinate actions of and among local agents, usually close colleagues working in teams. Standardisation is not required beyond local understanding (which may still entail a fair degree of standardisation under local management). The few local langauge elements used in resa/fasa medium system language should be known to the local community. The language is only closed insofar it uses fixed codes; of course there need to be linguistic conventions of some sort but productive use of concepts is ultimately free to individual language users.

Match: local process language generally works well enough. It seems to be subject to natural mechanisms of fine-tuning, and the sharing of linguistic knowledge occurs informally and mostly intuitively, though it takes time. Local language development and sharing is very much a social proces, which entails a particularly heavy 'social marking' of local terms. Local language partly marks an employee as belonging to one particular local office. This enhances social cohesion, but frustrates exchange of personnel between local offices on social *and* practical grounds. In rare cases, local language may be used (accidentally) in communication with other locations and groups; this may cause problems, but these are not indicated as serious. Somewhat contradictory, alongside centralisation of products, information systems, and the main lines of the production process, localisation of operational processes is part of Gak organisational culture and policy. Problems are only expected to occur when central language and local concepts overlap (unwanted cross-homonymy or cross-synonymy). Such cases have not been found.

Functional specification language as used in the central system and process development de

Purpose: communicating functional requirements for system functionality (including medium system language) to the party implementing the medium system. Communication should be unambiguous: optimally certain. Efficiency is secondary.

Match: the functioning of this language has two sides. As a means of communicating concepts forms, it is unproblematic. The implementing party literally copies them and 'puts them into place'. For many concepts, that is all. Much more problematic is the translation of 'cognitive semantics' to 'operational semantics' that occurs in relation to some concepts. This translation is typically performed by a skilled programmer. Operational semantics is associated with information processing (computational system) rather than with medium systems (human-human communication). Still, operational semantics may have perceivable consequences related to the meaning of medium system utterances: it may eventually end up co-defining cognitive semantics. However, such complexities are not explicitly covered by our current approach to the conceptualisation view on information system development.

Situational factors: technically occupied agents cannot be expected to be Gak process experts. This is embedded in Gak system development policy, which strives for rather strict separation of functional and technical specification and communication.

Improvements: one might make an effort to keep apart 'communicational functionality' and 'computational functionality' in functional specification, and raise awareness about this. An alternative approach might be to let programmers and process and system designers more closely cooperate in order to improve mutual understanding. In fact, this already occurs informally even at Gak/ASZ, but we suspect it may currently do more harm than good since it is not carried through all the way.

In addition to the matching points discussed above, we need to include the observed clash between on the one hand technical, and on the other hand legal, process and product languages. The general idea at Gak is that communication with technical people concerning medium system language is to be contained in a sub-domain of central system and process development DE: what we might call the 'functional specification exchange DE'. This DE is not part of our primary analysis (not seen as belonging to medium system language specification but as implementation of the technical system), though it has occasionally been touched upon. However, the suggested containment is not always respected. In particular, technically oriented agents are at times involved in providing information on the meaning of medium system language. This was repeatedly reported to cause problems.

Matching Meta-Communication with Concept Functionality Requirements

Information Sources: Non-Intervening Meta-Communication .

Purpose: providing accurate, to-the point, and *quickly available* information on centrally dictated terminology, in particular on the meaning of medium system language concepts. This applies especially to the AG Work Process DE, where the need for meta-linguistic information is most acute, but it may also be of use to the Central System and Process Development DE. Standards and centralised aspects must be respected, but if needed, concept description must be tuned to separate target groups (for example, work process agents must be distinguished from legal experts).

Means: diverse and scattered terminological information sources.

Match: means are underdeveloped, bordering on the dysfunctional. Looking up conceptual information is hard if possible at all, which is especially inconvenient given the limited time employees are willing and able to spend on this. Training (ex ante) and experience do the job up to a point, but cannot keep up with changes and cannot prevent ex post information requests from occurring, nor can they anticipate the exact nature of such requests. Written documentation and helpdesk-like support are vital as a complement. Personal communication may help, but is outside central control; it is a factor that is always there, but can hardly be managed (only encouraged or discouraged). In addition, it entails a risk of local amplification of some conceptual misuse. Personal communication is important but should be backed up by centralised documentation. Unfortunately, centralised documentation (including instruction/training material) seems to be the Achilles heel of non-intervening meta-communication at Gak. It fails with respect to standardisation and coherence, but also often lacks completely. Definitions and descriptions are scattered across many sorts of documentation and are indexed nor complete. There is no well-conceived functional differentiation of dedicated terminological resources (for example, dictionaries for particular user groups). Though an intranet is present, only rudimentary technical means can be applied for retrieval of terminological information (keyword search on limited documentation; no terminology-oriented information sources included). In some cases, agents without required expertise (technicians) are actively involved in information distribution.

Situational factors: resources and interest for specifically terminological documentation is limited. Need for better terminological knowledge sharing is paid some lip service, and some plans have actually been made, but nothing has materialised. Despite indicated troubles, people cope. Priority of providing better terminological information is not high enough to result in action being taken.

Improvements: information on centrally dictated concepts can only be kept coherent if documentation, training, and possibly also helpdesks are subject to terminology management; ICT can be of great help here, especially using the intranet. What the optimal balance might be between the amount of training (ex ante) and documentation/helpdesks (ex post) is hard to tell; supporting both *in an integrated fashion* is safest. The most basic solution to centralisation of terminological documentation would be to have a few well authored, dedicated, functionally differentiated terminological sources available, and force all other forms of documentation to conform to these sources and refer to them. A more advanced solution might entail automatic hypertext linking of the proper sources to documentation. Clearly, digitisation and on-line keeping of all documentation is a must in this case. More ambitiously, terminological documentation might be (hyper)linked directly to the concepts as they occur in the information system and in general documentation. Finally, it might be possible to keep better track of concept description and use (responsibilities, agents, motives) by introducing a dedicated 'concept monitoring and management system', covering both documentation authoring and and system development practices. Whether any of this is feasible in view of available resources is doubtful; technically, most options seem possible in principle, though incompatibility and inflexibility of technical systems are formidable barriers.

Change Options: Intervening Meta-Communication .

Purpose: providing possibilities for effective and well-conceived medium system language change where needed and allowed.

Means: ex ante, centralised analysis and design processes, with entailing documentation changes. Ex post medium system language change can be performed only in cases of emergency. Local RESA/FASA system changes are very limited and ad hoc in nature.

Match: quite reasonable given the circumstances. System changes are slow but carefully conducted. Unfortunately, documentation often lags behind. Highest priority is assigned to law changes; central changes originating in local initiative rarely occur, and only if supported in functional terms by a large number of local offices plus central authorities, and in addition are assigned high priority. Changes in local system languages seem also limited in occurrence, but can be implemented quickly and informally, basically by simply using changed 'semi-open' concepts, for example the '400 series' (agreement on this must be arranged locally, but is not documented in any way that is centrally visible, if at all). However, in our definition this falls outside real 'frozen langue' (which has to be *imposed* by an actual medium system).

Situational factors: medium system language change, in line with system change in general, is a centralised process. This is engrained in the technical, cultural, and organisational make-up of Gak. Changing it would require a complete reorganisation of Gak.

Improvements: except for very radical changes in Gak system development and systems architecture, which is totally out of our current scope, we see little possibility for improvement. Given the centralised nature of Gak we believe that centralised terminology control (including medium system language management) should remain the strategy of choice, which is to be extended rather than played down. Improved functional differentiation of medium system language (for the legal domain, production domain, service desk, etc.) can only ever be realised as part of a thorough revision of Gak information systems and information management. Local systems, or local sub-domains within centralised systems, might be developed and conceptually linked to current central systems (possibly even with 'translation layers' between domains). This would create new and more complex MSLCs, requiring more and far more in-depth information analysis and medium system design efforts.

Matching Conceptualisation with Meta-Communication Requirements

We distinguished two main conceptualisation processes: central language specification, and documentation authoring. In view of our integrated view on language, meta-communication, and conceptualisation, we provide an integrated discussion of these conceptualisation processes.

Purpose: providing accurate and functional RESA/FASA medium system language descriptions, ultimately for use in both process and system documentation, and medium machine implementation. The result should be a medium system language specification that, when implemented, is optimally functional as a means of communication, through selection of optimally functional concept labels *and* through provision of optimally functional information concerning concept meaning.

Means: ad hoc, deliverable-oriented conceptualisation (no clear, methodic approach), even in context of large centralised development efforts. Selection of concepts prevails, meaning description is considered secondary. Most defining is done informally. Semi-methodical, formal information analysis is occasionally performed in domains bordering on the MSLC, but strictly concerns computationally oriented formalisation of already specified medium system language.

Match: selection of concepts (phase 3) is performed reasonably successfully, though the process language that results from it is highly specialised, complex, hard to grasp for non-experts, and hard to acquire for trainees. Yet this is for a large part a consequence of choices concerning other matters than conceptualisation as such: most prominently, a centralised language policy, a language legacy, dependency on externally imposed terminology, and centralised development and management of core information systems. In addition, legacy and the presence of considerable imposed terminology (legal terms, LISV product terms) keep the need for 'new' or 'fresh' conceptualisation relatively low. Much more problematic than concept selection is conceptualisation concerning *meaning description* (phase 4.3), which should be well developed and controlled in view of the urgent need for good, coherent terminological information (non-intervening meta-communication). This urgency is linked to the complexity and opacity of medium system and process language, but also to its inflexibility (limited intervening meta-communication). Concept meaning description processes are often cut loose from concept selection, which increases the risk of the spreading of heterogeneous and uncontrolled terminological information. Processes for meaning description are scattered, unorganised, and rather insular (i.e. they do not necessarily relate directly to centrally authored definitions). Meaning description is performed with no clear regard for user-oriented and contextualised functionality of meta-communication deliverables.

Situational factors: high quality conceptualisation has a low priority compared to other matters in Gak process and system development and management. In particular, meaning description is not on the priority shortlist.

Improvements: if well-organised, effective terminology management were to take place at Gak, it should be centralised. Perhaps an initial re-analysis of meaning descriptions might be needed, with current core documentation and system specification as 'raw material' and a scan of peripheral documentation (both Gak-internal and external) in order to see wether things were missed or whether problematic conceptual differences were to occur. After a thorough analysis of functional requirements of *terminological documentation* for the various language user groups to be distinguished, definitive definitions might be authored and published centrally (intranet). After this initial action, integrated conceptualisation processes should be kept active as part of ongoing system development and documentation authoring. Each concept in medium system language (in RESA/FASA, but ultimately also in other Gak systems) should be centrally managed, and every language description in documentation should be linked to this record. Importantly, there should be a close link between conceptualisation and management of concept *forms* (as occur in the system, or in centralised and authoritative law, or in process and product descriptions), and concept *meanings*. Thus, the *integral management of concept forms and meanings* could be achieved. Possibly, a concept management policy could be integrated with general development and management practices (covering system development, training and instruction, quality control, structures of authority and responsibility, etc.). Finally, a more advanced approach to the effective combination of local and central systems and processes would have to include careful differentiation between DEs and the languages used therein. Possibly, separate conceptualisation processes might then have to be initiated and maintained in order to serve local and central needs, and in that case, well-managed exchange of conceptualisation results between conceptualisation domains would be vital.

Matching Medium System Development with Conceptualisation

Purpose: implementing a medium system machine that fulfils (linguistic) functional requirements as laid down in functional specifications (including medium language description); the implementation process does not only concern final results, but also the capacity to answer the needs of particular stakeholders, and the time, effort, and expertise needed to implement particular changes. Requirements are chiefly related to conceptualisation in view of *change options* (intervening meta-communication). We exclude system documentation authoring, which has already been covered.
Means: a centralised standard procedure for change and maintenance of the RESA/FASA system (alongside other systems). In addition, programmers can interfere with the system at a very low level in cases of emergency.
Match: given that combined ASZ and Gak ICT development and support processes as apply to the RESA/FASA system are steered strictly by central authority, system change is implemented reasonably accurately and timely. Organisationally, a thorough but slow procedure is followed for system change, aiming for a new system release each three months. Given availability of sufficient resources, technical implementation of changes could generally be realised much faster than the general procedure allows. Release management and general system develop management slow down technical execution of changes (though with good reason). However, human resources are limited, and priorities lie with the implementation of law changes. Consequently, many changes that are technically possible in principle never materialise for lack of resources. In addition, there are some limitations concerning lost information about details of system design. This is a 'technical knowledge' problem rather than a strictly 'technical problem'. Finally, matters of computation-oriented conceptualisation (involving formalisation) may occasionally become intertwined with the implementation of communication-oriented medium language specifications.

Summary and Discussion

The language situation in relation to RESA/FASA and Gak is characterised by the following points:

- Various terminologies are imposed and need to be respected; centralised terminology control is required.

- Some terminology that requires considerable expertise (mostly legal) penetrates domains where such expertise is not available (process and system language).

- Apart from legal expertise, *different* expertise is required to understand RESA/FASA and process language.

- At the higher levels in the organisation ('stafs', experts, process and system design) legal and process language expertise is not a problem.
- At the 'operational' level (production process) complexity and required expertise are a problem; people generally cope, but errors and misunderstandings occur. Perhaps they should not be confronted with terminology that is beyond them.
- Local work process language plays a minor role in RESA/FASA context. This is a direct consequence of the centralised nature of Gak organisation and the RESA/FASA system. Where used, however, local RESA/FASA language is quite functional; a small local language component is successfully incorporated in the RESA/FASA medium system language.
- People from technical domains understand very little of Gak process and product terminology. They definitely should not be confronted with it, and in particular not perform actions that require specialised terminological expertise.

In terms of linguistic meta-communication, the general situation is as follows:

- Intervening meta-communication is problematic, except where it concerns local language.
- For centralised adaptations, law changes draw all capacity away, leaving little room for more delicate language adjustment on the basis of experiences and requirements at various levels in the organisation.
- Also, many terms cannot be changed simply because they are imposed by higher authorities.
- In other words, there is indeed a frozen language situation; in view of the nature of the Gak organisation, language cannot easily be 'defrosted'.
- This is compensated by the considerable expertise (terminological and otherwise) of the Gak work force. Importantly, the Gak situation *allows* for a high level of terminological expertise to be maintained, but from an organisational (knowledge management) point of view, the situation is far from ideal (potential brain drain, training problems, flexibility problems, compatibility problems).
- The availability of terminological expertise can be complemented and supported by *non-intervening meta-communication*. The best chances for realistic improvement in over-all language functionality lie here.
- Many information sources (channels for linguistic meta-communication) exist. However, they are fragmented, uncoordinated, and quite often insufficient.
- Firstly, terminological information sources should be rigidly *centralised* where possible; documentation should use or refer to standard terminology descriptions. This implies that such information must be *generally and easily accessible*.

- Secondly, terminological information should be differentiated and tuned to particular user groups; for example, legal experts should be confronted with another type of information than production workers, even if the terms covered are the same.

- Technically oriented people should not be burdened with responsibilities and tasks concerning non-technical linguistic meta-communication. Currently, they sometimes are, which causes some problems.

The following can be said about RESA/FASA-related conceptualisation processes:

- Concept selection functions reasonably well given the circumstances. Intervening meta-communication is not hindered significantly by matters of conceptualisation, but by other factors, which fall outside our scope.

- However, concept *description* is unsatisfactory, and could be improved with relative ease.

- Meaning description as an activity is not connected to central activities of concept selection. This increases the risk of the spreading of heterogeneous and uncontrolled terminological information.

- Processes for meaning description are scattered, unorganised, and rather insular; they do not necessarily relate directly to centrally authored definitions.

- Meaning description is performed with no clear regard for user-oriented and contextualised functionality of meta-communication deliverables.

- By using fairly simple, available technical means, better and more centralised conceptualisation for non-intervening meta-communication could be realised. This would require some effort and resources, but more importantly it would require better *recognition* of the need to provide good and effective terminological information.

- Conceptualisation should become and integral part of the change mechanisms for products, processes, and systems at Gak. The resources required would be limited; it would rather be a matter of some operational discipline and the explicit distribution of responsibilities.

This concludes the analysis of the language and conceptualisation situation in the Gak RESA/FASA MSLC. In view of our basic view on frozen language as first presented in Section 2.2, the central conclusion is that *adaptability* of medium system language at Gak is restricted by factors that cannot realistic be changed. In other words, the use of frozen language at Gak is 'unavoidable', and even 'required'. To increase the adaptability of the medium system language, fundamental changes in the Gak organisation architecture, information architecture, and application architecture would have to be realised. Such a change seems quite unrealistic, especially considering the function of Gak as an organisation implementing laws, which are inherently 'centralised' and involve 'standard language'.

In view of limited possibilities for intervening meta-communication, non-intervening meta-communication turned out to be an important mechanism, though perhaps less important than the simple fact that Gak employees are highly specialised, well-trained, and experienced knowledge workers. This explains why there is a 'frozen language situation' but no major 'frozen language problem' at Gak. The Gak situation was not selected because it is exceptionally problematic; it is just *interesting*. It provides a good example of a fairly complex frozen language situation and the various factors and activities related to it. By applying our method, we have gained an understanding of the general language and conceptualisation situation, identified and understood some flaws, and indicated realistic directions for improvement. These were not very radical, and the problems indicated are not too serious. This is only to be expected in an organisation that, everything said and done, *works reasonably well*[14]. In the next chapter, we will suggest some more radical scenarios and more innovative directions.

7.4 Conclusion

Let us now consider whether the goals of the case study as set in the introduction of this chapter have indeed been achieved.

- "It should provide an idea of how the analytical and descriptive method presented in the previous chapter can be applied in a real situation (illustration)."

We can without hesitation claim that this has indeed been done. All diagrams belonging to the method have been used, in a coherent fashion, and hence have been shown 'in action'. Use of templates has less clearly been made visible (though it was explicitly indicated), but items appearing in the templates have been systematically dealt with in the text, which was structured roughly according to the five stages of the method. The matching points have also been demonstrated at some length.

- "It should convincingly show that instances of the concepts introduced can be meaningfully pointed out in the domain."

In our view, this has also been satisfactorily achieved. All conceptual 'pieces of the puzzle' (as listed in Section 6.2) have been relevantly used in the case description and analysis. Because of some limitation to the depth of analysis that we were capable of in the case study, perhaps not all the aspects in the conceptual framework have been demonstrated equally well (for example, conceptualisation techniques could not be studied in much detail). This is due to limitations on resources and authorisation concerning this particular case study. Also, the nature of the case has given rise to emphasis on complexity concerning particular aspects of our framework (for example, non-intervening meta-communication), while other aspects could be dealt with very briefly (for example, Phase 1 conceptualisation). This is related to the case situation chosen.

- "It should help chart and understand problems in the domain that are within scope of the method."

[14]Serious and prolonged failure of the Gak organisation might well cause serious civil unrest.

This too has been achieved: problems have been systematically traced and charted. Also, they have been explained in relation to the conceptual framework, or where appropriate by relevant observations of situational factors outside the immediate scope of our method (for example, general lack of resources for system development). This is in line with the architectural approach, which usually combines various viewpoints as considered relevant to a particular case. In addition, and importantly, we strongly feel that a degree of understanding has been reached that had not been possible without the analytical concepts and instruments devised in our method (for more on this, see below).
The problems that were charted and explained where not on the list of 'high priority problems at Gak', though inquiry showed that some had been discussed and in some cases had some urgency (for example, the problematic acquisition of Gak terms by new employees and the lack of practical terminological knowledge of agents at production process level). Since it was not the goal of our case study to solve urgent problems at Gak, but to explore the language and conceptualisation situation in a real organisation, this is irrelevant to the success of the study.

- "It may assist in finding directions for solutions of indicated problems."

Though the indication of possible solutions to encountered problems was no primary goal of the case study, we have added some suggestions as part of our evaluation in Section 7.3.5. In some cases, we have added a rough indication of how realistic we believe a particular suggestion to be in view of the case situation.
Generally, we can conclude that the case study has been successful with respect to the goals set, and we consider research question 2.2 to be answered satisfactorily. We do not claim, however, to have provided more than an elaborate example and a 'proof of concept'. Most importantly, the case was too limited for demonstration of all major types of situation, structure, and problem that might occur within scope of the framework. More extensive evaluation and validation of the method is most welcome in various respects, but could not be achieved without a considerable extra research effort and was therefore impossible within the scope and time span of the PhD project reported on (for more on this, see Section 8.1.4).
We now evaluate some key concepts and instruments as applied in the case study. They are presented roughly in order of appearance.

1. The notion of Medium System Language Community or MSLC has proved very useful in identification and description of the general case domain. Defining a community of agents in relation to various langauge-related aspects of an information system provided the needed link between human-human communication and ICT.

2. The notion of Discourse Environment or DE has also been crucial in making sense of language situations in relation to various types of ICT-related communication and language use. In particular, the distinction and relation between DEs for language use and for language specification has been essential. Less positively, we were surprised by how difficult it was to find the right granularity in DE distinction; we had some trouble doing this in the Gak case and ended

up with fewer DEs than expected. This is probably due to the still rather general level at which different Gak languages have been described. In principle, more 'dialects' and sub-terminologies might have been distinguished. We expect that it will be possible to usefully distinguish explicit and clear sub-DEs if more in-depth study of concepts in various domains can be conducted. However, we doubt whether in the Gak case, this would have rendered significantly more insight. The combination of various languages within one DE seems to reflect the typical situation at Gak. Therefore, we opted for the distinction of fewer but linguistically more complex DEs, reflecting the centralised linguistic character of the RESA/FASA MSLC and its high level of relatively global language use.

3. Two other central and vital notions in our framework are Medium System and Medium System Language. They reflect a language-oriented view on information systems that enables us to link them to underlying requirements and structures of communication and one particular instrument for this: language. The notions are not very frequently mentioned explicitly, but they are the foundation of our analytical enterprise; in the Gak case, they helped us understand what "RESA/FASA language" really was, and how it functioned.

4. The MSLC Diagram (instrument 1a) has proven useful, though modestly so –given its generic nature. It mostly helped in distinguishing and presenting DEs and explicitly label them as use-oriented and description-oriented.

5. The Language Identification Diagram (instrument 1b) has proved very useful, especially the use of Venn-diagram-like notation for languages and overlap between them. For the Gak case, the distinction of tasks and related roles proved of limited use for the one user-oriented DE, but we expect this to be different if systems include various more distinctive medium system languages.

6. The Language Description Template (instrument 1c), and the underlying *language use typology*, have proved useful, though exhaustive answering of all the questions on the template seemed an exaggeration and has not been performed. As intended, the template primarily served as a 'checklist' for potentially relevant issues, and as in inspiration for considerations concerning languages or terminologies[15]. As for individual types of language use covered by the template: they varied in analytic usefulness, but it was felt that for gaining insight in the character of a language, types viewed *in combination* were more useful than individual types (the whole being more than the sum of its parts).

7. The optional Concept Description Template (instrument 1d) was not actively used in the case study as reported in this chapter, though eight dysfunctional concepts were in fact studied in considerable detail. Even though most of these concepts were briefly used for exemplification, in-depth description was deemed insufficiently relevant in view of the case study goals. Possibly, the template aims at a level of detail that does not fit 'architecture level' description very

[15]Indeed it seems inadvisable to follow this template to the letter. This observation also holds for the other templates, though in varying degrees.

well (too much detail). Nevertheless, some reflection on problematic individual concepts has proved to be very useful for getting a good feel for the nature of language problems in the domain. The Concept Use Symptom Template (instrument 1e) was more relevant in this light, and has been successfully used.

8. The Conceptualisation Situation Diagram (instrument 2a), tightly linked to the Conceptualisation Exponent Template (instrument 2b), proved useful enough for analysis of conceptualisation processes, even though few explicit processes could be found at Gak. The detailed consideration of concrete artefacts as input and output of conceptualisation processes proved particularly helpful. Less successful was the distinction of clearly distinguishable conceptualisation techniques, but this seems mostly due to the fact that these are simply not around much at Gak. Nevertheless, the list of techniques applied provided support resulting in at least one interesting and unexpected finding, namely the identification of corpus analysis as a technique in Gak system management and development.

9. We unexpectedly found that at least in the Gak domain, 'conceptualisation roles' were less useful for analytical purposes than the more regular 'organisational roles' distinguished at Gak. This forced us to introduce a matrix table that is not part of our method (Figure 7.7: Chief Roles involved in Conceptualisation in the RESA/FASA MSLC). We are convinced that in a better (and more centrally) organised conceptualisation situation (also see point 7), conceptualisation roles are analytically more useful than regular organisational roles. However, given that we should also be able to cope with 'unorganised conceptualisation', it is probably a good idea to include the new matrix diagram in the method as a standard option.

10. The phased Conceptualisation Process Reference Model and the related Detailed Conceptualisation Process Diagram (instrument 2c) could only be put to limited use (also see point 7), but they certainly proved their value, for example by providing the insight that concept selection and meaning description are largely separate at Gak. In addition, even the *lack* of activity concerning a certain conceptualisation phase says much about the maturity of conceptualisation in a domain; a reference model can help detect such gaps and deficiencies.

11. The Conceptualisation Process Property Template (instrument 2d) and the Conceptualisation Symptom Template (instrument 2e), though again not used in full, proved quite helpful for the difficult task of analysing and describing the key properties of the two rather abstract conceptualisation processes distinguished in the Gak case.

12. The notion of Linguistic Meta-Communication as introduced in this thesis has proved invaluable for the general understanding of the essence of activities of 'freezing language'. In particular, the distinction between intervening and non-intervening meta-communication has provided a much needed, theoretically backed positioning of some previously blurred and seemingly overlapping types of language use and conceptualisation. In the Linguistic

Meta-Communication Process Identification Diagram (instrument 3a), this is concretely represented through the distinction of 'information sources' and 'change options'. These notions also proved very helpful in finding structure in a relative jumble of documentation types, roles, languages, and means of communication with system development and ICT departments. In addition, the distinction between communication-oriented and computation-oriented meta-communication sheds light on fundamental differences between the strongly entwined forms of language specification often encountered in technical domains.

13. The Linguistic Meta-Communication Process Description Template (instrument 3b) was modestly useful in analysing and describing the seven meta-communication processes distinguished. Because of the complexity of the meta-communication situation (in particular, the large number of different items involved), the 'derivation rules' proved quite useless (explosion of combinations). As an alternative, we opted to focus on information sources and change options as key concepts. Once this was decided, the template served reasonably well as a check list. The Linguistic Meta-Communication Symptom Template (instrument 3c) helped in systematically relating comments recorded in the interviews to particular aspects of meta-communication. Again, lack of particular types of symptom report were as telling as explicit remarks concerning dissatisfaction: the latter reflects awareness, the former suggests ignorance.

14. The Medium System Development Template (instrument 4a) and the strongly related Medium System Development Symptom Template (instrument 4b) have proved moderately useful in describing some situational factors concerning medium machine design, implementation, and general development. This rendered no new insights –but it was not required to, as it represents a strongly reduced overview of many important matters of general system, information, and development architecture. More in-depth analysis in these areas requires the use of dedicated viewpoints. Stage 4 of the method merely provides a link between such viewpoints and our conceptualisation viewpoint, adding no essentially new concepts or instruments.

15. The Meta-Communication Feedback Cycle was not explicitly used in the case study. Since it is not meant as an analytical instrument but as a general indication and illustration of the relation between ex post and ex ante meta-communication in an evolutionary context, this is not problematic. In addition, the case represents an example of a conceptualisation and system development situation that is hardly 'evolutionary', and in particular includes no clear occurrences of explicit ex post meta-communication. Therefore, the cycle is largely irrelevant to the case.

16. Matching Point 1 has proved very useful in helping us create a strictly *functional* view of the languages distinguished in the RESA/FASA MSLC. It created a focus on language dysfunctionality that covers a comprehensive package of factors, which formed a good basis for analysis of case-specific language failure and related meta-communication requirements.

17. Matching Point 2 has also been applied successfully, bringing to the fore strengths and weaknesses in Gak communication about language (most in particular, medium system language). Most importantly, it has helped to systematically uncover details of a serious deficiency in meta-communication: scattered and unorganised meaning description in an organisation that has an urgent need (and a policy) emphasising centralisation and standardisation of systems and work processes (in which central terminology plays an undeniable part).

18. Matching Point 3 helped in identifying particular aspects of conceptualisation as an activity, as an extension of the conclusions resulting from matching point 2. Since clearly defined conceptualisation processes mostly lack at Gak, the installation of conceptualisation procedures and support might be more a matter of 'construction' than of 'evaluation'. In other words, it has not so much provided insight in the current situation than that it helped sketch directions for improvement.

19. Matching Point 4 was used to relate general aspects of system development and design to matters of conceptualisation. It mostly served to outline limitations of the 'central language specification' conceptualisation process as related to 'change options'. It also gained us a few peripheral but still interesting insights, for example that even within the technology-oriented system development domain, strictly technical situational factors are only a minor cause for restricted organisational ability to fulfil changing system requirements; organisational factors and knowledge/information management are much more important. Given that matching point 4 indeed relates mostly to system development als a 'situational factor', this result is quite satisfactory.

Chapter 8

Conclusion

It is time to look back and consider what we have achieved. We will summarise and discuss our results, generally at the hand of the research questions introduced in Chapter 1.2. We divide the results in the problem analysis, the theory developed, and the analytic method derived thereof. In addition, we will suggest some interesting directions for future research, prominently including further validation and development of our current results, but also the exploration of connections with fields of research we were not able to take into account so far.

8.1 Results

Before we discuss answers to the respective research questions, we pay some attention to the observations and argumentation that led to formulation of those questions in the first place.

8.1.1 Problem Analysis and Proposed Solution

We addressed the "frozen language issue" (Chapter 2). The initial, intuitive goal of this thesis was to provide better insight into the frozen language problem, and into ways of solving it. However, to get there we first needed to understand the issue better. The main idea underlying our interest in language and information systems was that the language functionality of many information systems (of many different sorts) seems to be hampered by a discrepancy between, on the one hand, the nature of 'natural language', and on the other hand, the way in which language is built into and used through many information systems. This discrepancy had, to our knowledge, never been elaborately addressed before, least of all in relation to processes of information system development.
In order to make our initial intuition clearer, we first developed a general view on the relation between language (as used for human-human communication) and information systems. We focused on information systems as instruments for communication: people use them to talk to each other. Within the limits of this point of view, we consider information systems to essentially be *medium systems*. Medium systems like databases or web applications normally require a medium specification (audio, video, text, etc.; this is largely a matter of technology) but also a *restricted language specification*. We introduced the notion of "closed language" (as opposed

to "open language"), referring to this restricted form of language, which is part and parcel of contemporary information systems. We then presented a conceptual framework reflecting our view on language and its relation to information systems, distinguishing between an abstract *medium system language* and the concrete, implemented *medium machine* that gives it physical form. Thus, we contributed a more advanced language-oriented view on information systems than was hitherto available.

In traditional disciplines of information systems development (e.g. requirements engineering, information analysis, data structure design) it is generally assumed that the language that is to be used in some information system use domain can be sufficiently captured by having one or more domain experts or analysts list and describe the basic terms and phrases needed to communicate in that domain. This renders a static language specification that is then built into the information system. It is widely acknowledged that it is crucial that medium system language is defined carefully and in line with "user language": the language spoken in the use domain of the system (Universe of Discourse). It is also acknowledged that user participation in defining the user language is vital. However, in practically all cases, medium system language is approached as a static phenomenon, that can be described using static and preferably formal methods. Such specification mostly concerns language form, though meaning may also be involved. Static language specification and its use in information systems is not seen as hampering language functionality. We concluded that this view is somewhat naive. The traditional approach to medium system language design and its underlying assumptions about language are not fully compatible with fundamental views on language as held in various other areas of research (see Section 8.1.2 below). These fields very emphatically look at language as a highly context-dependent and adaptive means to convey meaning. From this perspective, the traditional view as held in information system development misses some important points about what language is, how it works, and what it does. The traditional view is not completely inadequate, but it is very limited and results in sub-optimal, if not insufficient language functionality of information systems. The functionally sub-optimal form of language that typically results from traditional medium system language specification we metaphorically named 'frozen language'[1]. The process of creating such frozen language was named 'freezing language'.

Our problem analysis included the identification of five crucial properties of natural language use that are potentially disabled by the restrictive specification and implementation of medium system language, resulting in damaged language functionality. All these properties are related to communication about language. The 'right' language may not be 'available' in the language use situation, and on-the-fly communication about and adaptation of language is hampered or impossible. In essence, therefore, the frozen language problem concerns *lack of linguistic meta-communication ability*. We are not aware of any existing analysis of the phenomenon based on similar properties; in fact, we believe even the frozen language issue has never been elaborately and explicitly addressed.

Our problem analysis also led to the insight that the traditional approach to medium system language engineering is not without merit, and in particular why this is so.

[1] This term is related to 'closed language', but refers to available language items being built into some medium machine, thus imposing a restricted language on the user.

There are some very good reasons to freeze language, that are tightly related to general attempts to achieve better communication. We introduced the view that, from a functional perspective, closed language generally enables communication with greater *efficiency* and *certainty* than open language –though the lack of possibilities for meta-communication may reduce these benefits. In addition, the need to introduce frozen language for communication may be *technical*: the inflexible nature of some technological means (medium machines) limits the possibility for using open language. In many cases, these three "practical factors" (efficiency, certainty, technology) cannot be seen apart, but distinguishing them helps us greatly in understanding the nature of frozen language, and in evaluating its functionality. We were thus led to develop and adopt a two-sided functional view potentially enabling analysis of the trade-off between pros and cons of the engineering and use of frozen language.

Exploring possibilities for improvement of the functionality of medium system language, we argued that to freeze language more carefully or to standardise language are not options that will solve all the fundamental issues indicated. This is because languaging and communication take place in many different and ever changing contexts, among people who wield various slightly different languages and who adapt their use and interpretation of language to fit particular contexts. The specification of a single-purpose closed language will often clash with the multi-context, adaptive use that is inherent in open language use. Consequently, static language specifications will in many cases keep falling short of the communicative mark.

Since our analysis of the frozen language issue suggested that some of the the limited aspects of frozen language functionality cannot be fully remedied by "better freezing language", but also that the freezing of language is an unavoidable and even useful phenomenon, we proposed an alternative approach to compensate for the negative effects of freezing language. We investigated how the five properties of language use hampered by the freezing of language might be 'repaired'. Those properties are without exception linked to *processes of meta-communication*, most in particular *linguistic* meta-communication. At the core of such linguistic meta-communication lie *processes for explicit conceptualisation* (i.e. processes for describing language), but for such processes to be optimally effective, they need to be understood and implemented in such a way that they fit the particular *language and communication environment* they occur in. As a crucial step in countering the frozen language problem, then, we needed some theory that enabled us to *analyse and understand conceptualisation processes in their environment.* Development of such a theory was made our main research goal.

This led us to the formulation of the main research question:

Q1 Which theoretical framework enables us to describe and analyse processes and environments in which language is being frozen?

In addition, we strongly felt that (for various reasons) our theoretical framework should be made *operational*: it should be developed to a degree which allowed its application to research of real conceptualisation situations, rendering detailed insights in those situations and enabling evaluation thereof. This resulted in a second research question, complementary to Q1:

Q2 What is an operational form of this theoretical framework, by which we might use it for concrete analysis and evaluation of processes for freezing language in an ICT context, in order to improve such processes?

So the first and main result of our research is a theoretical framework which "enables us to describe and analyse processes and environments in which language is being frozen". The second result is a method that, as demanded in research question 2, presents "an operational form of the theoretical framework". The method is not only the operational side of our theoretical coin, it is also a crucial instrument for validation of the theory (more about this in Section 8.1.4).
The nature of our framework and the type of use it is most likely to be put to have led us to shape it as an *architecture viewpoint*: a collection of dedicated analytical instruments (diagrams, question templates) and some methodic guidelines for their application (see Section 1.3.4). 'Information architectures' are chiefly used as a means for guiding and managing information systems development. An architecture viewpoint can be used for architecture-level analysis, negotiation, and description concerning *some particular aspect* of organisations and their information systems. Our method is shaped as an architecture viewpoint for conceptualisation, to be used in context of information systems development. It is therefore an instrument that can help guide the analysis, organisation, and management of information system development processes in order to improve the general linguistic language functionality of the resulting information systems.
Below we will discuss both the theoretical and the methodological results, and in the process check whether a number of 'research sub-questions' have been answered satisfactorily.

8.1.2 Theoretical Underpinnings of Meta-Communication and Conceptualisation

The research sub-questions underlying Q1 were the following:

Q1.1 What are the processes involved in freezing language, and how can we describe them?

Q1.2 What are the environments in which the freezing of language may take place, and how can we describe them?

Q1.3 How do processes and environments for freezing language relate to and influence each other?

Q1.4 What are the linguistic mechanisms and factors underlying the description of, need for, and course of the processes of freezing language?

Q1.5 How are linguistic mechanisms and factors underlying freezing language related to information system development?

Q1.4 and Q1.5 had to be addressed before Q1.1-3 could be satisfactorily answered. Some partial answers to Q1.5 were already provided as part of our initial problem analysis.

A Functionalist View on Language, Meaning, and Meta-Communication

As the basis of our answer to Q1.4 (and for a part also to Q1.5), we needed to back up our tentatively formulated views by a more detailed theoretical account. We presented a comprehensive view on language and language functionality, with special emphasis on meta-communication and conceptualisation (Chapter 3). Language was positioned as an instrument for communication, wielded in combination with a highly intuitive, largely internalised set of skills for 'languaging'. In general, our view on language can be characterised as radically *functional*. We see language as a device for *doing things*, in line with the pragmatic approach to language. Though development of functional language and communication theory was not a goal of the thesis, we combined and complemented existing theories, introducing some concepts of our own and creating a broad and coherent theoretical basis. The functional view on language and communication was more strictly applied and further developed than in most existing work on 'user language' in relation to information systems.

We deemed it particularly important to consider the approach to language *meaning*. Conveying meaning is, after all, the chief purpose of language. Building on various existing linguistic and philosophical arguments, we argued that language meaning can never be 'objectively' or 'truly' described. Representations of meaning can ultimately only describe subjective and abstracted *approximations* of meaning, and yet may do so quite adequately given a limited goal –for example, the goal of reaching *sufficient* agreement about what some language utterance means (in context of a particular communication situation). This conclusion pointed us towards a meta-communication approach to language and meaning description: if meaning cannot be captured 100% 'correctly', it can still be *communicated about*. In our view, this should be the chief if not the only goal of creating meta-linguistic representations. Also, if the use of describing meaning depends on particular situations of language use, then it is possible that 'anticipatory' communication about meaning does not suffice, and that additional, 'responsive' communication about meaning is required.

Because no clear theoretical underpinning of (linguistic) meta-communication could be found in the literature, we set out to provide one. We based our account on an existing branch of communication theory centred round *speech act theory*. In particular, we proposed how linguistic meta-communication can be fit into Van Reijswoud's 'Transaction Process Model' for business communication. This resulted in a communication-theoretical framework for linguistic meta-communication. We also paid some special attention to the analysis of the activity of *definition* from a functional perspective, because of the central role it plays in activities of describing meaning. We based this analysis on work by Viskil. This concluded our fundamental exploration of the linguistic (and, as it turned out, communicational) theoretical background of conceptualisation processes (Q1.4).

Extending the Framework to Fit an Organisational Context

Next, we turned to the development of new theory concerning the embedding of conceptualisation and meta-communication as performed in ICT-supported organisations (Chapter 4). To our knowledge, none of the theoretical concepts and devices mentioned below have been put forward or suggested before, nor has any in-depth analysis of linguistic meta-communication processes in ICT development.

We set out to bridge the gap between functionalist language theory and ICT practice (partly answering Q1.5 and Q1.2). In particular, we extended the formalised conceptual framework introduced in Chapter 2 in order to allow its application to *Medium System Language Communities* (MSLCs), covering all people (or 'agents') involved in *using* or *describing* some medium system language. In order to make distinction possible between multiple, interactive domains of concept use of various different types, we introduced the notion of *Discourse Environment* (DE). The concepts of MSLC and DE enabled us to define and distinguish the various environments in which conceptualisation and language use take place.
Still on the basis of our functionalist view on language, we proceeded to develop a *typology of language use*, aimed at enabling classification of concepts and terminologies in terms of their relation to various kinds of language *domains*, *functionalities*, *priorities*, *knowledge issues*, and *changeability issues*. We needed such a classification to enable a broad analysis of functional language requirements, i.e. the requirements that should be met by the linguistic meta-communication and conceptualisation processes resulting in language descriptions.
We also placed the theoretical notion of linguistic meta-communication into an applied, organisational context. Importantly, we provided a link between linguistic meta-communication and more well-known processes of medium system development, support, and use. This resulted in the creation of a *typology of* ICT*-related meta-communication*, enabling us to describe and evaluate the various kinds of linguistic meta-communication that occur across the various DEs in an MSLC. Most importantly, we introduced distinctions between *ex ante* and *ex post* meta communication, between *intervening* and *non-intervening* meta communication, and between *mediated* and *non-mediated* meta-communication. We illustrated the typology with a number of examples.

Analytical Framework for Concrete Conceptualisation Processes

Activities related to explicit conceptualisation are the most clearly visible, practical exponent of the activity of freezing language. More importantly, they are the primary focus of our chosen 'direction of solution'. Therefore, such activities needed to be covered by our theoretical framework, in order to make their description and analysis possible. We conducted an extensive literature study of conceptualisation techniques, methods, and practices, covering the fields of conceptual analysis and design, computer supported cooperative work, ethnography, knowledge management and knowledge engineering, information retrieval and library systems, and lexicography, translation, and terminology management. On this basis, we constructed a *reference model for conceptualisation processes*. We distinguished 5 main *phases* of conceptualisation that seem to hold more or less universally:

1. acquiring raw material
2. capturing an envisaged UoD
3. selection of relevant concepts
4. naming and defining selected concepts

5. performing quality checks on previously conceptualised concepts

The main phases were divided in various sub-phases. In view of its relevance to freezing language, the 'definition' sub-phase of Phase 4 was described in particular depth, based on existing functionalist literature on definition.
In addition to the phased reference model, we devised a description of typical input and output of the phases in terms of *language artefacts* and *knowledge types*. Also, we discussed the application of six general *process quality attributes* to conceptualisation processes (functionality, efficiency, usability, maintainability, reliability, and portability). Finally, we listed and described a fair number of *conceptualisation techniques*, and related each of them explicitly to conceptualisation phases they might be used in. Also, we gave an estimation of the effect of application of each technique on conceptualisation process quality attributes. We thus provided a rather comprehensive package for the analysis of conceptualisation processes and the activities underlying them.
Together, the analytical conceptual instruments listed constitute our analytical framework for conceptualisation processes, answering Q1.1, Q1.2, and Q1.3 (though the latter was to be further and more explicitly answered by some relations captured in the method). This concluded the theoretical part of the thesis.

8.1.3 A Method for Analysing Conceptualisation Processes in Context

The research sub-questions underlying Q2 were the following (formulated as design goals rather than actual questions):

Q2.1 Develop a conceptual framework for description and analysis of situation-specific aspects of frozen language in the ICT domain

Q2.2 On the basis of the conceptual framework, develop an analytical method enabling the analysis and evaluation of concrete processes and environments of freezing language in real ICT situations

These questions point towards the development of an operational analytical (conceptual) framework and a way of working. The 'Architecture Viewpoint' was already put forward as the intended type of analytical framework we were to develop (Sections 1.3.4 and 6.1). Though our method mostly aims to be analytical, it could also potentially play a role in constructive contexts. However, within our current effort, we had no ambition to validate the use of the method for actual conceptualisation process (re)engineering.
The development of a coherent, integrated method of the conceptual devices of our theory was expected to contribute considerably to the cohesion of the theory as such: it explicitly and coherently interconnects the separate 'pieces of conceptualisation puzzle'. Therefore, development of the method was not only performed to get an answer to questions Q2.1 and 2.2, but also to complete the answer to Q1.3 ("How do processes and environments for freezing language relate to and influence each other?"). We believe our method to be unique in terms of its goal and focus, and its compatibility with other (more conventional) architecture viewpoints. Below we will describe the results in some more detail.

Gathering and Combining the Conceptual Devices Developed

A large part of the development of the method concerned the putting together of a modelling language for describing language and conceptualisation situations in ICT-supported organisations. As a basis for this we used the previously developed conceptual devices. We followed the requirements for architecture viewpoints as described in (Sherlund et al., 2000). Because of this, we had to add some concepts in order to complete the viewpoint. In particular, we provided a generalised overview of typical *stakeholders* (five of them) and *concerns* (seven of them) relevant to our focus (stakeholders and concerns are considered crucial components of any architecture viewpoint). The framework is our answer to Q2.1.
In order to be able to coherently combine various relevant aspects within our viewpoint, we distinguished seven interrelated domains of analysis (reflecting the aforementioned concerns):

1. organisation
2. communication
3. operational communication
4. linguistic meta-communication
5. conceptualisation
6. implementation of concept representations as a medium system
7. information system development

We systematically investigated the relationships between these domains, which resulted in the selection of four domains essential to our viewpoint (domains 3-6). Those domains became the basis of the development of the analytical stages of our method: the *language situation from an operational point of view*, the *explicit conceptualisation situation*, the *linguistic meta-communication situation*, and the *medium system development situation*. The order of these stages in the operational method differs from that of the seven domains. This is the case to enable analysis to start with the most concrete aspects (observable languages and conceptualisation practices) and end with the more abstract, theoretical aspect (meta-communication processes). However, the order does not need to be followed strictly. A fifth, *evaluating* stage was added, in which results from the previous phases are to be *matched*: *needs* as identified in one area of concern are explicitly related to *means* as identified in another. Thus, we enabled comparison between needs and means as related to the respective essential levels of analysis. In this way, we were able to develop the first comprehensive analytical and evaluative conceptualisation viewpoint to date.

An Operational Method Based on the Theoretical Framework

The flesh on the methodological skeleton was added in the form of five *diagrams* and ten *question templates*. They are the *instruments* of the method, reflecting the theoretical framework. The diagrams are specialised descriptive devices based on UML Class Diagrams, making available our theory-based concepts in a uniform and coherent setting, optimally compatible with general architecture practice. The templates are lists of questions or 'checkpoints' aimed at guiding investigation and discussion concerning particular areas of concern. Some of the templates enable guided elicitation or observation of *symptoms*: flaws or problems particular to some area of concern. Finally, four main *matching points* were identified and described: the most important areas for comparison of 'means and ends' as identified through analysis according to the method. Thus, evaluation can be conducted according to some guidelines. However, the matching points are not recipes for comprehensive and 100% systematic evaluation; some analytic creativity will be required in view of the many complex combinations of factors imaginable, and possible dependencies between them. The over-all method serves as an answer to Q2.2, but also extends, illustrates, and further clarifies our answer to Q2.1.

8.1.4 Grounding and Validation

Validation of our theory and the analytical method derived thereof rests on four pillars, as discussed below.

Grounding in Existing Theory

The theory developed is based on existing and previously validated theory (chiefly functional linguistic and communication theory), as indicated through explicit discussion and references, mostly in Chapter 3. The basic concepts and conclusions were summarised in Section 8.1.2.

Link with Existing Practice

The theory is also partly based on an extensive study of existing practices of conceptualisation throughout a number of fields, as described in the literature and indicated by references and examples (see appendices A and B). Our theoretical framework enables the *positioning and evaluation* of concrete conceptualisation approaches and activities with respect to the general language and meta-communication situation in which they are embedded, as indicated mostly by examples (as presented throughout chapters 1 and 3-5).

A Theory-Based Analytical Method

The theory has been successfully translated to an operational analytical method, including a way of working, a number of diagrams and templates, and an approach for the evaluation of analysed situations. This demonstrates the comprehensiveness and coherence (within our scope) of the theory underlying the method. The method is described in Chapter 6. Though the conceptual instruments of the method are indeed based on the theory introduced in earlier chapters, the method has also been influenced by our experiences in a case study (see below). Note that the method in its current shape was never intended to be applied in industrial system development practice; it was devised to make the theoretical concepts more operational, and make it possible to demonstrate their coherence and analytical potential.

Proof of Concept of Method through Case Study

A 'proof of concept' of the method has been provided by means of an elaborate case study in a real organisation, which rendered detailed insights in the organisation's language, meta-communication, and conceptualisation situation. It brought to the surface conceptualisation-related flaws, explanations thereof, and situationally realistic directions for solutions to the problems indicated. This was described in Chapter 7. The case study was of a qualitative nature, involving twelve carefully prepared interviews with people representing various levels in the organisation. A stepwise approach was followed: part of the goal of each interview was to identify new key people and establish contact with them. The interviews were recorded, and analysed later. This involved partial transcription. In addition to the interviews, we were allowed to study some system development policy documents and some 'operational documentation'. We were mildly restricted concerning disclosure of confidential information.

The case study contributed substantially to the development of the final version of the method. Data gathering in the case study was conducted on the basis of a premature version of the method (including very elaborate question templates, but no diagrams); in-depth analysis and reporting on the findings and conclusions, however, was performed using a version of the method very close to the one described in Chapter 6. It cannot be denied that the two-way link between the method and the case study (method development at the hand of the case study as well as validation of the method through the case study) introduced a circularity in the validation set-up concerning the method. However, we believe that this circularity does not damage validation much. After all, the method is still firmly based on theory; the case study contributed little to our fundamental insights, it merely helped put them in an operational, 'crystallised' form. The one undeniable effect of the case study on the method is that the suggested layout of some diagram types seems somewhat biased by the case study results. We consider this trivial in relation to the validity of the theoretical framework as such.

We conducted an explicit validation of the individual instruments of the method, and of some basic theoretical notions; about twenty concepts and instruments were discussed in view of how well they worked and how much they contributed to the case analysis. At the hand of a description of the goals of our case study, we concluded that, at least in context of the case study, the method generally functioned well as an analytical instrument. On the whole, the method clearly "helped chart and understand problems in the domain that are within scope of the method". As a bonus, the method "assisted in finding directions for solutions of indicated problems". Given the nature of the method, i.e. it being (or reflecting) an "architecture viewpoint for conceptualisation", we believe that using the method in an information architecture context does indeed add considerable insight concerning a relevant aspect of ICT-supported organisations that was simply not covered before.

On the down side, the case study did not comprehensively demonstrate every detailed aspect of the method. In particular, the analysis of individual conceptualisation processes and techniques proved too ambitious in view of our particular case study set-up: we were unable to bring to the surface much detailed information on explicit conceptualisation processes in the organisation. We believe this was due to

the case situation and setup, not the method.
Note that the case study was in no way equipped or intended to prove that application of the method in an actual, management-related architecture and alignment context has positive effects on conceptualisation, communication, and ultimately on general operations in an organisation.

8.2 Further Research

Much work lies ahead. Both the theory and the method call for improvement, extension, and –perhaps most importantly– further validation. Also, the framework and method could be used to guide innovation of conceptualisation practices (techniques, methods, tools) and conceptualisation environments, perhaps even as a basis for 'concept management'. And last but not least, our findings could be applied to various fields of research that were hardly touched upon so far in this thesis. We will discuss all these points below.

8.2.1 Further Validation

We acknowledge that validation and consolidation of our present theory and method still leave much to be desired. Thus, serious attention to further validation is due.
Generally speaking, we should apply our method (and thereby the theory underlying it) to more, and more diverse, situations, thus exploring a broader package of structures, configurations, and practices that may occur within the scope of our framework. In other words, we should extend the range of case studies the theory and method are tested in in order to see whether they really cover all the ground required, i.e. whether the concepts enable us to gain insight in the frozen language issue in relation to other situations than the one described in the Gak case. We should attempt for various different cases to evaluate their language and meta-communication situations; to explain why some meta-communication or conceptualisation processes may be sub-optimal and suggest how this might be improved on. In addition to studying issues relevant to the *validation* of our current method and theory, we might eventually work towards improvement of our capacity to *create* tailor-made meta-communication and conceptualisation approaches (architectures, supporting techniques) for various different situations (i.e different 'information system architectures' and 'information system development architectures').
We envisage three main directions in further case study concerning our theory and method:

- Study of how organisations deal with the frozen language issue
- Study of conceptualisation and meta-communication practices
- Study of our method's effect on organisation, management, and architecting

Study of how organisations deal with the frozen language issue

One direction for further (case) study concerns the 'big picture' of the frozen language issue as reflected in the title of this thesis: "freezing language" in "ICT-supported organisations". Various parameters may be explored, but in general two sorts of situations can be distinguished that require a radically different approach:

1. Organisations of which the information systems do not/cannot change much/often

2. Organisations of which the information systems do/have to change much/often

The Gak situation is typically of the first sort, so we already have some empirically-based findings available. In Gak-like organisations, emphasis lies (or should lie) on compensation of the undesirable effects of frozen language by means of non-intervening meta-communication. Therefore, some question that we should now address is: how do slow-changing organisations[2], being confronted with frozen language, cope with this? Further questions: do problems similar to those encountered in the Gak case occur elsewhere (and are there clear patterns)? How large and widespread is the problem? Which solutions have slow-changing organisations found in order to compensate for the negative side effects of the frozen language phenomenon?
As for fast-changing organisations: on the basis of theoretical considerations, we expect that they cannot afford to freeze language to a great extent. This leads to one main question: how do fast-changing organisations prevent the freezing of language (where necessary), or how do they 'de-freeze language'. In particular, are there organisations that actively support intervening ex post meta-communication, and if so how do they do this? Serious candidates for case study are, for example, those organisations offering and marketing fast-changing information-based products in a multi-channel, digitised environment (for example, insurance companies and financial service providers with a strongly web-based sales and service infrastructure). Such organisations are a speaking example of environments with a strong 'e-commerce' or 'e-governance' interest; we expect them to be particularly vulnerable to the frozen language problem. One particular area for exploration that we expect to be quite fruitful concerns situations in which there is a minimum of possibility for 'open' linguistic meta-communication: information exchange based on strictly web-based communication, involving (preferably complex) form-filling and automated processing.

Study of conceptualisation and meta-communication practices

A second possible direction in case study concerns conceptualisation and meta-communication practices. These practices not only deserve to be studied in view of the support they might provide to the general 'Language and Meaning Game' played in an organisation (the 'big Frozen Language picture') but also outside the larger context, i.e. as individual activities with their own dynamics.
In view of the reported limitations of the Gak case, we should first of all try to better demonstrate those aspects of the method that have not been satisfactorily covered in our single case study. We think in particular of extended study of processes

[2]We assume the relation between the change rate of an organisation and the changing requirements of supporting information systems to be sufficiently clear by now, and refrain from further discussion.

for explicit conceptualisation as they occur in real environments. In this light, we also consider the artificial creation of circumstances in which such conceptualisation is performed, which would clearly bring us closer to experimentation. However, since our method aims primarily at analysis at 'architecture level', such experiments should always concern artificially created situations placed in a realistic context; after all, we are ultimately interested in conceptualisation as performed in some wider context of meta-communication. For example, in a real organisation, we might fake a serious language-based communication error or impose a substantial conceptual change, and see how people cope with it in context of their working practice.
Below we list three examples of situations that should be interesting in view of further study of conceptualisation and meta-communication practices.

- It would be interesting to study concrete conceptualisation in situations in which several different but strongly related frozen languages need to be merged or somehow unified, since this is expected to raise all sorts of problems concerning heterogeneity and standardisation of language. For example, consider the merger of Gak with various organisations with a very similar core business, as occurred in the case of the creation of the UWV organisation as dictated by the Dutch government (this transition will in fact take many years; for more on this, see Appendix C).

- Another interesting subject for case study would be any environment in which very few terms are imposed from 'outside', i.e. an organisation in which 'self made' frozen language prevails (probably an organisation involving some unique or innovative process or product). It would be particularly interesting to see how communication with the outside world by means of 'new' concepts is supported in terms of meta-communication and conceptualisation.

- Other interesting situations would be those in which customisation of language within small groups is carried through as far as possible. One might think of a cooperative work environment (perhaps part of a virtual organisation) in which people are encouraged to continuously create and adapt their working language. This suggests effective 'evolutionary' conceptualisation support at a highly individualised level. Such situations are typically the subject of research in the field of Computer Supported Cooperative Work; we would like to explore the language, conceptualisation, and meta-communication aspects of such situations in relation to our method.

Study of our method's effect on organisation, management, and architecting

A quite different, and rather ambitious type of further validation for our method and theory would be to investigate whether application of the method is successful in context of organisation and management practice (including architecture and alignment efforts). Such success must be expressed in terms of observable (preferably *measurable*) improvement of the situation subject to some *change effort*. In other words, it involves the measuring of the effects of management. To do this, in theory

two possibilities exist: the study of real life implementation of changes, or the study of simulations.
To our knowledge, no attempts have been made so far to simulate architecture and alignment processes. We expect such simulation to be particularly problematic in the light of the many and heterogeneous factors that play a role here. This makes modelling highly complex. In addition, very little attention has been paid in architecture research to the *process* of 'architecting'; instead, the subject is generally approached from a representational perspective (i.e. one focuses on the resulting architectures). Attempts to rationalise and model the process (and the assumptions, tacit agreements, and implicit or even unnoticed decisions underlying it) are in very early stages of exploration (Veldhuijzen van Zanten et al., 2003). The possibility for validation through simulation must therefore be rejected at this point.
Thus we are left with study of the effects of architecting (including the use of our conceptualisation viewpoint) in real situations. This requires for the method or components thereof to be adopted and used by process managers and teams. As mentioned, the current method is not meant for industrial use. Though it is possible for (aspects of) the method to be indeed adopted by industrial parties, its specialised nature and current academic status make it likely that the method is to be evaluated through use in a *participatory* setting, i.e. the method should probably be tested in practice by someone who is involved in an information system development project and who *also* has a direct academic interest in the method (for example, the author of this thesis). This would require an approach known as *action research* (Avison et al., 1999). For a good example of the application of action research to architectural description see (Lassing, 2002).

8.2.2 Elaboration of the Theory

The theoretical framework as developed and applied in this thesis covers a rather broad area. It mainly includes functional linguistics, but also touches upon matters of philosophy, cognitive science, communication science, sociology, organisation theory, and computer science (to name but the most prominent fields). Though we would like to think that our theory generally holds, we acknowledge that elaboration and consolidation is called for. To name a few interesting issues:

- Extension of the theory to enable more to-the-point and detailed functional evaluation of language representations (i.e. leaping back from conceptualisation processes to their products).
- Better connection with existing functionalist linguistic theory, working towards more detailed and more comprehensive functional analysis of language in use context (for example, more explicitly including matters of syntax and dialogue).
- Better connection with existing communication theory, in particular those branches focusing on organisational communication and information.
- Improvement and elaboration of sociological, sociolinguistic, and organisational underpinnings of the theory, perhaps working towards simulation of meta-linguistic interaction.

- Improvement and elaboration (and more far-reaching formalisation) of our theoretical foundations, in order to bring our theory in line with existing theoretical practices in fields like computer science and general systems theory; in particular, an exploration of dynamic approaches to semantics (semantic *methods* in addition to semantic *representations*).

- Better understanding of the relation between concepts as used for human-human communication and computational manipulation of those concepts; in other words, better understanding of the relationship between cognitive semantics and operational semantics.

8.2.3 Elaboration of the Method

The method presented in this thesis is limited to aspects of language, conceptualisation, and meta-communication as might be used analytically (and, up to a point, constructively) in an architecture and alignement effort. Apart from refinement of our conceptual framework within its current scope, there are two main ways in which the method as it currently exists might be improved and extended. Firstly, we might attempt to add some form of metrics to the method, making the various factors and the relation between them quantifiable and more concrete. This could lead to a more exact, even formalised wielding of the various factors and Matching Points, allowing us to apply techniques from the realms of formal reasoning and calculus. Given the nature of the matters concerned, however, we doubt whether full scale formalisation and quantification of the 'alignment process' for conceptualisation and meta-communication processes will ever be possible. Nevertheless, we believe that it would be an improvement if we could reach at least a higher degree of formalisation and quantification than is currently the case.
Secondly, we might develop the framework and method from the essentially analytical instrument it is now into an approach for *concept management.* To our knowledge, no comprehensive approaches to concept management (the management of the creation, maintenance, and use of explicitly represented concepts) currently exist. All related approaches either concern instruments for representation creation and maintenance (meta-languages, editors, browsers, etc.) or devices that *use* conceptual representations. What is needed is a framework that helps position and align language requirements in organisations, processes of concept creation and use, and the methods, tools, and techniques supporting them. Based on our framework, which combines a functional view on language with the provision of architecture-level structures, we should be able to develop a package of analytical, evaluative, and *guiding* principles and heuristics for the successful and integrated management of all matters concerning language and conceptualisation that may need managing. Importantly, the approach would have to include clear guidelines for the implementation of *language change* but also of *language consolidation* as well as *translation* where required, in line with various types of meta-communication required in the domain of application. The approach should refer to or include *tools* and *techniques* for the practical support of concept management and explicit conceptualisation, probably 'imported' from other, more representation-oriented approaches to language engineering. Thus, we might achieve an integrated, explicitly *functionally driven*

approach to language engineering and management, positioning and combining existing solutions for freezing language in a set-up that is much better integrated in the larger communicational context than is currently possible.

Managing language partly concerns the question of how to *bring about* 'situationally effective and appropriate use of language'. This will involve the striking of a functionally optimal balance between the factors already included in our method combined with ways to *manipulate* those factors. For principles and guidelines for concept management to be included in our method, we will have to extend our conceptual and theoretical framework with views and notions from the field of *management science*. We will need to include a coherent and functional view on matters like *motivation*, *initiative*, *resources and resource planning*, *authority and control*, *government*, etc.

8.2.4 Improvement of Conceptualisation Practices

Within a shifted and generally more technical focus, we can use our gained insights as a basis for improvement and innovation concerning actual *conceptualisation techniques and practices*. As is extensively discussed in Chapter 5 and appendices A and B, quite a number of such techniques and practices already exist. However, we are not aware of any of these tools being based on explicit requirements analysis in terms of language functionality and general (meta-)communication context. We believe our theory and framework can contribute much to such requirements analysis, and expect our emphatically functional approach to language engineering to be of help in developing effective and realistic conceptualisation techniques. Some suggestions in this vein were already included in Section 5.4.2.

In particular, we are interested in *dialog-based, cooperative conceptualisation*, to be used in context of both ex ante and ex post linguistic-meta-communication. To push beyond existing techniques, we may well have to increase our insight into 'natural' or 'intuitive' mechanisms of linguistic meta-communication. More in particular, we need better insight in detailed mechanisms involved in the *adaptation* and *refinement* of existing concepts to situationally determined needs. The conducting of experiments in this vein should be considered. Also, we should investigate the effective combination of 'manual' approaches to conceptualisation with automated techniques as emerge from extensive work done in fields like knowledge and information extraction.

In line with an integrated approach to language, meta-communication, and conceptualisation as suggested above, our insights may contribute to the creation of effective 'language specification environments', possibly in context of information system description and development in its broadest sense (touching upon such areas as architecture description, conceptual modelling, knowledge engineering, data engineering, requirements engineering, etc.). In addition, we might extend our findings to include principles and guidelines for evaluation and *procurement* of existing conceptual descriptions (ontologies, terminologies) or even of *conceptualisation services* (providing, for example, brokering, negotiation, translation, or customisation of conceptual descriptions).

8.2.5 Application to Other Fields

Finally, let us briefly consider how we might apply our ideas to fields peripheral to the focus maintained in this thesis.

Most closely related to this thesis, we have found that there are promising possibilities for extension of our theory to situations in which matters of 'frozen language' are less emphatically linked to medium system language. As a prominent example, we consider the practice of *architecture description and negotiation* (in the sense of (Sherlund et al., 2000), as discussed in Section 1.3.4). Relevant to our language and communication interest, a primary function of ICT architecture and architecting is to *enable communication between stakeholders* (including the architect). This inherently involves matters of stakeholder vocabulary and terminology, but in current architecting practice it also involves various forms of conceptual modelling. Discussion, negotiation, and definition of concepts (both informal and formal) is a frequent and important sub-activity in ICT architecting activities. In the ongoing ArchiMate project, one of the main goals is development of an 'architecture core language' (Jonkers et al., 2003), as well as adaptation and translation of this core language to a number of architecture and stakeholder languages as developed and used in various large companies. Our participation in the ArchiMate project includes analytical application of a slightly adapted version of our method (Hoppenbrouwers, 2003). The project is also is a good opportunity to perform a case study of conceptualisation in action.

There is an additional, more general link between architecting practice and the method presented in this thesis. We have developed an 'architecture viewpoint for conceptualisation', introducing a language and meta-communication-oriented conceptual framework (a 'language') in order to describe and communicate about an aspect of architecture we consider relevant. There seems to be growing support in the ICT architecture community for the idea that alongside more traditional architecture perspectives (see Section 1.3.4), introduction of various new, non-technical viewpoints is called for. Thus, we might see the development of a legal viewpoint, a privacy viewpoint, a political viewpoint, and so on and so forth. All such viewpoints involve the development of terminologies for description of specialised domains in combination with a holistic approach to organisations and systems. Consequently, the meta-linguistic perspective always plays a role. This guarantees a key role of the conceptualisation perspective in architecture practice. Interestingly, it seems relevant and useful to apply our architecture viewpoint for conceptualisation to architecting as an organised activity.

Freezing language plays a central role in electronic communication ('business to business' communication as well as internet-based communication between clients and businesses/organisations). Also, related but more computationally-oriented data exchange between web-based applications is increasingly integrated with electronic communication. As mentioned before in this thesis, various large web-oriented projects involving extensive language freezing have been initiated in recent years (Kotok, 2002; Berners-Lee et al., 2001). Such efforts acknowledge the importance of freezing language carefully, and also the complex mechanisms required to use and deal with its results. However, largely in line with views held in mainstream information system design, their approach focuses on the development of advanced standard languages and meta-languages (for example, see Aiello et al., 2002), and extensive domain-specific conceptual modelling. In other words, they follow a primarily representational approach. We believe that the expanding globalised exchange and use of language specifications will lead to the increasing and continuing need to cope with language specification *processes*. We have provided some means that may help achieve better insight into such processes, and it seems very much worthwhile to apply our newly developed theory to the world of inter-organisational, web-based electronic communication.

In a quite different vein, we expect our theory of conceptualisation and meta-communication to be applicable in context of Computer Supported Cooperative Work. In particular, we hope to contribute to development of support mechanisms for reaching agreement on and description of concepts, for use in groupware environments and other socio-technical applications of ICT. Interestingly, the theory developed within the Language Action Perspective (an important part of the theoretical background of this thesis) has already been successfully applied to CSCW, for example in relation to use-based specification of cooperative environments (de Moor, 1999); this provides a good basis for a well-founded extension of CSCW to include user-based language specification.

Moving yet further away from our original area of research, we may consider study of language and conceptualisation in relation to various computational technologies that may be applied in some discourse environment or in close relation to a medium system. For example, we might consider relations with information retrieval and (semi-)automatic translation. Particularly interesting situations involving information storage and retrieval concern their combination with some form of 'personal conceptual profiling'. Most approaches to profiling involve automatically derived profiles. Complementary to such approaches, we would be interested in studying situations that include 'manually' specified, individual conceptual specifications, involving some form of semantic description (ontologies). Such specifications could possibly be combined with pre-fabricated, subject-specific ontologies, requiring well-guided and high-quality yet feasible conceptualisation. As for automatic translation: most if not all approaches to this holy grail of computational linguistics require detailed, high-quality lexicons. Keeping these vital resources up to date and in line with translation domains (both source domains and target domains) also requires intensive conceptualisation, which needs to be organised realistically and effectively.

Finally, an ambitious but promising possible extension of our research involves the field of artificial intelligence, in particular the issue of communication between *automated agents*. Such agents will have to be able to communicate with humans, automated systems, or other agents on behalf of some human or organisation they represent. In many cases, this will entail the use of frozen language, that will have to be specified by or for the people or organisations the agents represent. This requires proper conceptualisation support, and a theoretical framework providing insight in the language games played. In addition, if agents are to be capable of communicating outside a fully known, formalised conceptual framework, they will benefit greatly from a built-in capacity for linguistic meta-communication. This may involve effective automated comparison of and translation between pre-specified conceptual frameworks (ontologies), but possibly also strategies and protocols for establishing a common frame of meaning between agents (negotiation of concepts, concept related contracting, decision taking concerning the required level of 'common meaning', etc.). All this requires substantial insight into mechanisms of linguistic meta-communication in context of operational communication and language use. This thesis contributes some theory that may help develop the capacities needed, but quite a substantial additional effort will be needed in order to automate linguistic meta-communication. While we doubt whether agents will ever be able to perform meta-communication with the same flexibility and effectiveness as their human counterparts (in view of our fundamental beliefs concerning language and meaning; see Chapter 3), we do believe linguistic meta-communication *of some sort* will *have* to be part of advanced inter-agent communication. As a direction of solution, we would like to advocate the exploration of the notion of *linguistic reassurance* as introduced in Section 3.4.2. Assuming that *exact* agreement about (cognitive) meaning cannot be achieved through the exchange or creation of meaning representations, it should be possible to reach *sufficient* agreement in view of a certain communicative goal and situation. Such goals should probably be formulated in terms of a functional approach to language and communication.

Appendix A

An Overview of Conceptualisation Practices in Various Fields

The goal of this appendix is to sketch existing approaches and attitudes to actual explicit conceptualisation as they occur in a number of fields, without detailed attention to the various techniques applied. This provides us with a wider context of explicit conceptualisation processes as they currently occur (though admittedly, some approaches have not made it beyond the research lab).
The fields under consideration are not all imaginable fields involving conceptualisation, nor are they selected from some existing list. They have simply caught our attention because they have in common that they are the *most typical* fields that include or make use of *techniques for creating explicit conceptual representations* (the actual techniques will be discussed in Chapter 5.4). However, not all these fields have an established, well described tradition in explicit conceptualisation. Sporadically, conceptualisation is seen as a primary activity, though the actual term 'conceptualisation' is used in none of the fields (not in 'our' sense, that is). Yet all fields include relevant approaches to explicit conceptualisation. Our end goal is to develop a general, comparative view on conceptualisation processes which enables us to discuss particular techniques in relation to particular types of conceptualisation, and to particular circumstances in which explicit conceptualisation processes occur. Thus we may eventually be capable of evaluating conceptualisation processes and techniques in context of their functionality, and perhaps even provide advice and guidance as to the 'engineering' of processes for explicit conceptualisation (however, actual conceptualisation process engineering lies outside the scope of this thesis).
A terminological remark is in order here. A 'field' as it is presented here is a 'focus based on practices shared by some professional community' and as such may reflect various perspectives: it may be based on shared goals, theories, methods, technologies, etc. The fields discussed usually combine several of these perspectives. A 'field' as we use it should primarily be seen as an 'application area'.
The six fields related to explicit conceptualisation we discuss are the following:

1. Information Analysis and Requirements Engineering
2. Computer Supported Cooperative Work
3. Ethnography

4. Knowledge Management and Knowledge Engineering
5. Information Retrieval and Library Systems
6. Lexicography, Translation, and Terminology Management

For each application area we will try to answer the following main questions:

- What goals does each application area try to achieve?
- What role does explicit conceptualisation play in this?
- What typical support is provided for explicit conceptualisation?

In view of our a wider functional perspective on conceptualisation (as presented in Chapters 1 to 5), the following *main goals of conceptualisation* can be distinguished:

Communication-oriented conceptualisation aims for explicit expression of concepts (linguistic representations of some sort) in order to *communicate* with other parties. It aims primarily for human-human communication. This goal is seldom the primary one in the fields discussed; in fact it is hardly ever made explicit.

Implementation-oriented conceptualisation aims for formalisation of concepts in order to make them *implementable* and fit as input to or a program for computational (data) processing. Most typically, software programming includes this approach to concepts. It is a common, primary goal in all design and engineering environments involving the freezing of language: the actual "building of language into a machine".

Analysis-oriented conceptualisation aims for better *understanding* of a problem or situation through making a coherent description of it (and the concepts occurring in it). Unlike the two other goals, this goal has not been elaborately mentioned in this thesis, and is not central to our discussion of 'freezing language', but since it is an important goal in some of the fields to be discussed, it needs to be included.

Our focus in this overview dictates an emphasis on procedures aiming for or involving *communication-oriented* conceptualisation. While we will honour this view, we cannot afford to close our eyes to approaches aiming primarily for other goals, as long as they can realistically be used in support of communication-oriented conceptualisation.

A.1 Conceptual Analysis and Design

Information system development is traditionally associated with engineering disciplines, in which two general phases are commonly distinguished: analysis and design. When it comes to engineering approaches to the language component of information systems, we find that these phases are linked to two well developed, strongly related fields: Information Analysis (IA) and Requirements Engineering (RE). Arguably, IA is part of RE.

Requirements Engineering (or Requirements Analysis) is concerned with the phase of computer system design in which a designer or analyst tries to find out in detail what is asked of the system-to-be. The main target of requirements engineering traditionally is to find out what the functionality should be of the system-to-be, though additional attention had to be payed to non-functional system properties (Chung et al., 2000). If the system in question is an information system (and indeed we focus on this type of system), information analysis is a common phase to precede, or be included in, the requirements engineering process (Burg, 1997; Loucopoulos and Karakostas, 1995; Nijssen, 1993; Halpin, 1995). We will focus mostly on IA, since there communicative-conceptual (user language related) matters are mostly dealt with.

We should also mention here the strongly related discipline of Conceptual Modelling (CM). CM is the practice of drawing up representations (often diagrams of some sort) which present an organised view of some conceptual structure, for example a partial system design, business model, architecture, etc. "Conceptual" in CM refers mostly to the need to abstract from details of implementation in early design stages. In a conceptual model, a (usually simplified) version of some structure is represented, mostly by means of describing a number of elements and relations between them, showing only the relevant aspects of the structure in question.

In principle, most diagrams, from a straightforward graph to a complex schema of an electronic circuit, could be called conceptual models. Also, mathematically described structures are a form of conceptual model. However, there is a narrower interpretation of CM in ICT practice, traditionally relating conceptual modelling to the field of information and data analysis. As such it typically concerns relational diagrams, flow charts, etc., possibly (but in an industrial context: rarely) complemented with the mapping of such diagrams to mathematical representations (typically involving first order logic). In addition to use in hardware, software, and data engineering, CM has increasingly been used to represent organisational structures, business processes, and even knowledge. CM now arguably belongs to knowledge engineering as much as to information analysis (Section A.4). Still, CM seems fairly uniquely connected with the world of information and communication system design in its broader sense (i.e. including the development of knowledge based systems). If a conceptual modelling approach is specifically aimed at the detailed analysis and representation of information structures that should be reflected in the information system, CM becomes almost synonymous with IA.

Analysis-oriented conceptualisation plays an important part in the field under consideration, though this goal seems subordinate to the end goal of implementation-oriented conceptualisation. Communicational conceptualisation plays a relatively hidden role, mostly disguised as 'information system functionality analysis' or 'data structure design'. Information analysis and data structure design are perhaps the most typical activities in which user language is built into the information (medium) system; it is therefore of primary interest as an activity through which the freezing of language occurs.

In the early days of computing, functionality analysis and design was a task almost exclusively involving technical specialists, typically the people dealing with implementation of the system. Technology in fact often dictated all but the most basic choices concerning tasks executed by a system. For example, matters of interface, data structure, and manner of programming were largely given in by the limited technical possibilities offered by the machinery used. Gradually, technology advanced, and systems became more powerful and complex, but were at the same time more often meant for use by non-experts, introducing a need for *user friendliness*. It became clear that at least in principle, system use and users should be better taken into account and as a consequence, user input became increasingly more important.

In this vein, a direction in RE known as *User Centred Requirements Engineering* has developed (see for example (Maguire, 1997) on the RESPECT project), in which explicit techniques are suggested through which analysts can *elicit* from users what they want and expect of a system. Comparable ideas have found their way into common practice in information system design (Wood and Silver, 1996), in particular in large and complex design efforts. Note that the important role of the Universe of Discourse in some IA methods (see Section 4.1.1) is roughly in line with user orientation, and emphasises the role of language in information system functionality.

Information analysts are increasingly burdened with a communication problem: requirements and other input from various heterogeneous sources (and brought forth by different stakeholders) have to be understood and combined, and the resulting (functional) specifications have to be expressed in a way which respects conceptual differences between stakeholders, allows for clear and specific instructions to be given to the implementing party, and enables *validation*[1] of the specifications. In addition, the resources and approaches concerning the creation of such specifications should match the goals of the relevant stakeholders as well as the investment they are prepared to make. The analyst thus increasingly becomes an intermediary person, negotiating compromises between the demands and conceptual frameworks put forward by individual stakeholders (also see sections 1.3.4 and 8.2.5). She has to, as it were, speak the languages of all stakeholders and must provide adequate translation between them. Note that communication with various stakeholders is one of the primary goals of the field of *Information System Architecture* (van der Poel, 2000; Sherlund et al., 2000).

Not all conceptualisation occurring in RE involves communication between use-related stakeholders and/or analysts. Conceptual Modelling is also intensively used in the domain of implementation, even if complete functional requirements specifications are available. Though the tendency exists among programmers to 'think in code', highly complex problems are often tackled by drawing diagrams. Models are also used to communicate solutions between members of the implementation team. Many types of representation may be used, some of them formal, others very informal. The main point is that even conceptual models that strictly serve implementation may have a strong communicative component.

Support of explicit conceptualisation mainly consists of providing many frameworks for representation (meta-languages) and the tools to edit, store, (re)format, and translate representations. The various types of language use as discussed in Section 4.2 are distinguished up to a certain point, but the differing (and often overlap-

[1]The related notion of *verification* is usually interpreted as 'checking syntactic coherence and correctness' and will not be discussed here.

ping) communicational functionalities of languages are rarely acknowledged as such. Languages abound, yet meta-communication remains an unrecognised and therefore rather ad-hoc performed process.

Something most CM 'methods' have in common is that they are only marginally accompanied by instructions or advice concerning the modelling *process*. Most commonly, a language is suggested in which the models or descriptions can or should be expressed. In addition, the use of *data dictionaries* is widespread; these form-and-meaning descriptions of data elements are of course of great interest to our current effort, but they too are a *result* of conceptualisation and do not concern the process of conceptualisation as such. If a process for conceptualisation is suggested, the use of NL text or NL sentences usually plays some role as a basis for analysis. For more on the relation between conceptual modelling and lexical matters, see (Hoppenbrouwers, 1997).

So a methodic 'cook book', often looked for by analysts in a production environment, is seldom available and even then the creative, authoring-like nature of CM seems to clash with the detailed, stepwise description of the process of conceiving and constructing conceptual models. One notable exception to this is an approach called NIAM, the Natural-language Information Analysis Method (Nijssen, 1993), in a slightly different form also known as ORM, Object Role Modelling (Halpin, 1995). The differences between the two approaches are marginal. NIAM and ORM are professedly NL-oriented. They include a detailed analysis process description, and also the underlying formalisation of their representations is highly developed; see (de Troyer, 1993), among others.

On a rather different level of description, methodic (stepwise) approaches to system design exist in abundance (for example, DSDM, as described in DSDM Consortium, 2001), but they usually do not descend below the level of describing roughly what model *types* should be used or delivered at particular stages in the system development process. How the actual models are to be conceived and put together is typically not commented upon.

Interestingly, CASE-tools (Computer Aided Software Engineering tools) may enable software developers to use diagrams or other conceptual representations as a basis for software development. If the representations are sufficiently powerful and formal, they may even be used to generate code from conceptual models. A long standing goal in software industry is, indeed, to restrict requirements engineering to conceptual modelling in implementation-independent terms and leave implementation to automated processes. Ultimately, adjustment of the models should automatically lead to adaptation of the system. This is still mostly science-fiction. Yet even if the tough nut of 'executable requirements' were cracked, the creation and (especially) the maintenance of conceptual-functional specifications would represent a task that should not be underestimated.

In sum: for RE and IA to be carried out, many kinds of conceptualisation may play a role, from a first, sketchy NL description of a system's main functionality, via informal and formal diagrams and mathematical representations, to the creation of executable code. The fields of Information Analysis and Conceptual Modelling form the typical background of those activities that render representations of conceptual systems (derived from NL) for use in the design of information systems. Semi-formal and formal diagrams are popular and the discipline of conceiving and drawing them is perhaps one of the best developed sources of explicit conceptualisation techniques around. Unfortunately, little guidance is provided in the actual, operational process of conceptualisation: the focus usually is on the resulting models and representations, and the languages used to express them in.

A.2 Computer Supported Cooperative Work

As mentioned earlier, in Dutch information industry and research, the term 'Information System' (IS) has largely been replaced by 'Information and Communication System' (ICS). This reflects the fact that the importance of communication in the application of computation in organisations and society has by now been widely recognised. For people to cooperate, they have to be able to share information and make sure it is mutually understood, i.e. cooperation almost equals communication; communication is also tightly related to *organisation* (Taylor, 1993; also see Section 4.1).

Adjacent to the widespread deployment of information systems (typically, database-related applications), there is a field of research called CSCW: Computer Supported Cooperative Work (Baecker, 1993): "computer-assisted coordinated activity such as communication and problem solving carried out by a group of collaborating individuals." (ibid. p.xi). CSCW is closely linked with a particular type of ICT application known as *groupware*: "Groupware represents a paradigm shift for computer science, one in which *human-human* rather than human-machine communications and problem solving are emphasised. [...] CSCW may also be viewed as an emerging scientific discipline that guides the thoughtful and appropriate design and development of groupware (Greenberg, 1991)." (ibid. p.xi). CSCW can thus be seen as a sub-field of information system development (as covered in the previous section), but its goals are much more human-oriented (and group oriented), sufficiently so to treat it as a separate field of application here.

In CSCW, the 'representation of work' is of great importance (Kyng, 1995). Analytic conceptualisation thus plays a big role, probably the chief one. Matters of implementation-oriented conceptualisation are also involved, but are not at all the main concern. Communicative conceptualisation is relatively prominent in this field, in connection to its human/social focus. Though the field is not primarily language-oriented, the CSCW community is actively aware that language plays an important role in social action and communication, and therefore also in information system development (Muller, 1995). However, language is generally dealt with in rather abstract terms, and the field merely touches upon conceptualisation as an explicit, well-distinguished activity.

Conceptualisation is not the central issue in CSCW, but because *human-human* communication language is important, language issues are relevant, and indeed crucial (Robinson, 1991, as reprinted in Baecker, 1993 p42). CSCW entertains relatively advanced views on the balance between natural language and frozen language, which seems in some respects to be related to the notions of the "cultural" and "formal" levels of language, respectively.

> "The "formal" level is essential as it provides a common reference point for participants. A sort of "external world" that can be pointed at, and

> whose behaviour is rule governed and predictable. The "cultural" level is a different type of "world". It is an interweaving of subjectivities in which the possible and counterfactual [...] are as significant as the "given". [...] Interpretation and viewpoint take the place of rules and predictability. [...] The formal level is meaningless without interpretation, and the cultural level is vacuous without being grounded" (Robinson, 1991, p42-3).

Explicit conceptualisation as seen from our "frozen language" perspective, then, is approximately the creation of "formal" level language on the basis of, and as much as possible in tune with, "cultural" level language.
Without exception, CSCW-related design and development approaches are user centred (also see previous section). In addition, the field has a particularly positive attitude towards ethnographic methodologies (see Section A.3). Also, CSCW approaches show a substantial interest in techniques derived from linguistics. For example, see work by Clark and Brennan (1991) as reprinted in Baecker (1993, p.222). Another interesting link between conceptualisation and CSCW concerns the application of *concept mapping* techniques, sometimes supported with dedicated groupware. For an investigation into this link, see (Vermunt, 1996).
CSCW approaches involving conceptualisation often concern the analysis of subcategories within assumed, pre-constructed conceptual systems (for example, a given taxonomy of tasks). Importantly, such frameworks reflect a particular view on the medium system in question that is therefore imposed on the further analysis and design process. Though language and conceptual representations are seen as important, hardly any explicitly communication-oriented support seems to be provided. Models are produced in abundance, but their creation as such typically falls under Sections A.1, A.3, or A.4.
Conclusion: CSCW is (perhaps disappointingly) more a field of application for the ideas presented in this thesis than a source of conceptualisation techniques. In general, it is representation and application of the results of conceptualisation, and in particular their use in design, that the field focusses on. The most interesting aspect of CSCW concerning conceptualisation is the awareness in the field that human-oriented techniques (ethnography, social system engineering, linguistic pragmatics) are relevant in conceptualisation.

A.3 Ethnography

Ethnography basically concerns the practice of describing (domains of) culture and society, and is typically associated with the social sciences. However, language being a primary 'carrier' of culture and a vital instrument of social interaction, the study of language (perhaps rather: of concepts carried by language) plays a key role in ethnography. For example, imagine an anthropologist setting out to study the culture of a newly discovered indigenous tribe in the impenetrable heartlands of New Guinea. Essentially, she will try and understand the culture of its people by making a 'conceptual map' of the way they perceive their world and have conceptualised it. Uncovering a people's conceptual world view (closely connected with words, and sometimes with grammar) then is a crucial instrument –more crucial, perhaps,

than merely observing the behaviour of the tribes(wo)men, which can only indirectly render an insight in how they think. Conceptual analysis in ethnography thus focuses in particular on the 'semantic categories' in some sociocultural domain: 'the way the people in some domain divide their world into meaningful elements –and the relations they see between those elements'.
Given its close ties with the social sciences, it is no wonder that CSCW explicitly and gratefully draws on ethnography for techniques of conceptual description (for example, see Fafchamps, 1993, who applies ethnography to workflow analysis). CSCW is in fact the main field that links ethnography with information and communication system design. Ethnography chiefly involves analysis-oriented conceptualisation, though the role of communication is clearly recognised. However, ethnography is not linguistics, even though some links between advanced linguistic (and even grammatical) analysis and culture have been explored in dept[2]. An interesting exponent of a culturally inspired yet practically oriented brand of linguistics is 'field linguistics' (Brend and Pike, 1977).
(Hammerstey and Atkinson, 1983) discuss the role of concepts in the process of ethnographic analysis as follows:

> "Concepts sometimes arise 'spontaneously', being used by participants themselves. [...] Some forms of ethnography [...] are devoted almost exclusively to the listing, sorting, and interpretation of such 'folk' terms. They are concerned with the more-or-less formal semantics of such inventories. However, many ethnographies, while using such types, attempt to do more than simply document their meaning. Their use will be examined as evidence of knowledge, beliefs, and actions that are located within more general analytic frameworks." (*ibid.*, p.178).

(Spradley, 1980) describes in great detail a language-based approach to ethnographic conceptual analysis that is remarkably similar to comprehensive analytic approaches in requirements engineering and information analysis. However, it is more universal in terms of the situations it may cover, and has a relatively open eye to methodological and practical pitfalls concerning objectivity, bias, and other social phenomena that might hinder sound, scientifically viable analysis (in particular as related to *participant observation*, the most popular methodological context for ethnographic conceptual analysis).
Interestingly, ethnographic methods are often quite thorough in systematically exploring possible conceptual angles; analytic (meta-)conceptual systems used show striking resemblance to views or aspects as distinguished in various fields involving ICT-related conceptual system description. For example, *ibid.* (p78) recommends "nine major dimensions of social situations" to be taken into account:

1. *Space*: the physical place or spaces

2. *Actor*: the people involved

[2]Most people will be familiar with the mythical case of the enormous number of words for snow that Inuit (often incorrectly referred to as Eskimo) people are supposed to use. Erroneous though this may be (Pullum, 1991), it is a good example of linguistics bordering on ethnography.

3. *Activity*: a set of related acts people can do
4. *Object*: the physical things that are present
5. *Act*: single actions that people do
6. *Event*: a set of related activities that people carry out
7. *Time*: the sequencing that takes place over time
8. *Goal*: the things people are trying to accomplish
9. *Feeling*: the emotions felt and expressed

Ethnography is a relatively fruitful source of elaborate, conscientious, and scientifically viable yet practical approaches to conceptualisation, providing considerable guidance in general conceptualisation methodology. However, though these approaches actively contribute to advanced and comprehensive ex ante conceptualisation methodology (in particular in context of CSCW), they are far removed from 'quick-and-dirty', technology-oriented conceptual analysis as practiced in ICT industry. Nevertheless, it is interesting and encouraging to find solid methodological procedures for conceptualisation in a field that is distantly but undeniably related to ICT-oriented conceptualisation.

A.4 Knowledge Management and Knowledge Engineering

The fields of Knowledge Engineering (KE, the creation of representations of knowledge chiefly to be used in knowledge based or 'intelligent' computational systems) and also its application in Knowledge Management (KM, a direction in organisation and management focusing on optimal creation, sharing, maintenance, and storage of knowledge), necessarily concern conceptualisation insofar they concern knowledge represented in terms of language and 'linguistic concepts'.

As examples of knowledge *not* expressed through language, consider a connectionist learning algorithm that is applied to the visual recognition of bad vegetables on a conveyor belt (pattern recognition) or the trained imitation of human body movements by a robot arm (recording and use of spacial and/or mechanical data). Concerning the difference between *knowledge that* and *knowledge how* (see Section 3.2.9), knowledge engineering typically focuses on the former.

The conceptualisation goals in this field are not unlike those in the field of CSCW, but the emphasis on communication-oriented conceptualisation is stronger. This is because explicit conceptual representation (knowledge representation) is central to the field, but also because of the general awareness that knowledge needs to be communicated chiefly through language.

Knowledge engineering does not as a rule focus on *linguistic knowledge* as such. Language is generally approached as a means, not a goal, though it is sometimes hard to say where language ends and general knowledge begins (Sections 2.2 and 3.2.9). On the other hand, in attempts to understand and apply knowledge as expressed through language, conceptual issues cannot be ignored (Duesterbeck and Hesser, 2001). In addition, methods and techniques in KE and KM increasingly make use of (computational) linguistic approaches, for example in order to achieve knowledge acquisition from texts (Dieng and Corby, 2000; Ahmad, 2001). Reversely, knowledge-based approaches to terminological domain analysis have been used (Meyer et al., 1997). KE and (to a lesser extent) KM professedly share many common interests with fields such as database research and development, linguistics, and CSCW.

Analysis-oriented conceptualisation plays an important role in KE and KM, especially concerning *knowledge elicitation* (Firlej and Hellens, 1991). Implementation-oriented issues arise in particular if conceptualisation aims at the creation of *knowledge based systems* (typically part of KE), but are seldom the main focus. Communication-oriented conceptualisation also plays some role, but at an abstract level, mostly in efforts for *knowledge sharing*, which is a primary concern for KM.

Techniques, methods, and tools for knowledge modelling typically belong to one of two levels of description: on the one hand, general analysis of knowledge in context of use, goals, resources, etc.; on the other hand, actual conceptual representation of (elements of) knowledge. Both levels are extensively covered by comprehensive modelling approaches such as CommonKADS (Schreiber et al., 1999). Knowledge models at a general level typically consist of a number of sub-models that at type level resemble forms or templates. Modelling thus takes the shape of selecting relevant form types and then filling them in. This process can in some cases be aided by a "knowledge model editor" (for example, Protégé-2000 as described in Dieng and Corby, 2000, or the CODE system discussed in Meyer et al., 1997). To our knowledge, such editors do not provide extensive help concerning which template needs to be filled in when, or how quality of the information thus provided is guaranteed. Nevertheless, knowledge model editors can be of great help in activities closely related to and including explicit conceptualisation.

Much research effort has been put into the design of knowledge representation languages; so many of them exist that we do not even begin naming them here. There seems to be some overlap between languages for knowledge engineering and languages used for conceptual modelling. In particular, it is sometimes unclear what the real difference is between representations for information analysis and data modelling, and representations for knowledge modelling. The most important difference apparently lies in the particular conceptualisation goals for which particular forms of representation are designed and/or used, but these goals are rarely made explicit.

The conceptual payload of knowledge models (language-related level of modelling) is now very much associated with *ontologies*. Arguably, knowledge engineering is the main discipline dealing with ontology design and use. We do not discuss ontologies in depth here, as we are only interested in their relation with conceptualisation. For more see, for example, (Guarino, 1998; de Vree, 2001; Berners-Lee et al., 2001). While some ontologies define concepts at a very elementary level indeed, covering the basic semantic building blocks of language (for example defining, at the top of the ontology, "entities", "relationships", "actions", etc.), most ontologies in KE are typically tuned to a specific domain (Weigand and Hoppenbrouwers, 1998b). Thus, they provide a limited conceptual space in which a domain can be further modelled at instance level. In this sense there is again a clear resemblance between ontologies and conceptual models and data structures.

For a large part, then, conceptualisation as discussed in this thesis includes ontology building in KE. We have already observed that conceptualisation support as offered in KE mainly consists of the availability of highly developed languages for ontology description and the availability of tools for editing, storing, and exchanging ontologies. We look towards knowledge engineering as a field that holds great promise for conceptualisation support.

However, we feel that at the same time, KE suffers from the same limited focus that we observed in general ICT design: a certain preoccupation with languages, representations, and computerised tools to cope with them. How to deal with knowledge-oriented conceptualisation processes in a dynamic, communicationally challenging environment has not been a main issue in KE so far. Methods and techniques for ontology creation and maintenance that we have studied are without exception both elaborate, labour intensive, and above all product-oriented (as opposed to process-oriented).

One relevant theoretical focus is ignored by KE but has received some attention in KM: the balance between open, natural language and closed, 'frozen' language. For example, value of intuitive knowledge sharing through informal conversation ('war stories', 'coffee corner conversations'), as opposed to exchange of explicit 'knowledge that', is a major topic. In addition, related literature quite explicitly includes notions and support concerning the *tacit dimension* of knowledge (Nonaka and Takeuchi, 1995; Hoppenbrouwers and Weigand, 2000; Hoppenbrouwers and Proper, 1999, also see Section 3.2.7). However, the down side of KM is that it generally deals with such matters at a level of abstraction so high that it looses touch with conceptualisation as a process. Possibly, the ideas put forward in this thesis may contribute something to bridge this gap.

To conclude, KE is a highly relevant field concerning conceptualisation techniques, especially its contributions to ontology building and management. However, KE is not unlike information analysis and system design in that it focuses on conceptual representations (frozen language) rather than the why and how behind their creation and maintenance. On the positive side, an open attitude towards both 'frozen' and 'living' language can be found in KM, but once more this field offers little or no techniques for conceptualisation since explicit conceptualisation falls outside its scope.

A.5 Information Retrieval and Library Systems

Put briefly, the main goal of Information Retrieval is to find documents or smaller textual fragments that fulfil the information need of some party as expressed in a query. For the purpose of our current discussion, a distinction between two basic types of IR can be made: IR from *structured data* and from *unstructured data*. Structured data is the kind stored in well-designed, structured databases, with a clear and known data structure. IR involving structured data is normally identical to querying databases.

Unstructured data typically consists of blocks of NL represented in ASCII-coded strings, though it may also include, for example, digitised images or sound (which we are not concerned with here). Its data structure is not initially known, nor is it well-organised. There are degrees of data 'structuredness': it is quite possible, for example, for a text to be well structured down to paragraph level and be unstructured text 'below' that, or for a structured database to contain large fields of 'open' text.

Databases (involving structured data and 'frozen language') are in the centre of our focus. Conceptual problems in IR may occur in the querying of both structured and of unstructured data. Unstructured data does not typically concern closed (frozen) language, but 'open' language in textual (persistent) form. However, many approaches to IR aim to *create explicit conceptual structure where there is none* as a first step (automatic analysis of texts; indexing), preceding a representation matching process. This often introduces previously selected 'frozen language elements' and imposes them on a text (Roulin, 2001).

All core information retrieval techniques basically aim for some sort of *matching* of conceptual representations (say, data that means something to some human, in the linguistic sense), more in particular the matching of a query representation to a representation either *pointing to* the item to be retrieved or *being* the item to be retrieved. The trick thus is to match representations, and this is based on the assumption that identical representations indeed mean (sufficiently) the same.

The important difference between IR from structured and unstructured data is that in the latter case, the initial data analysis preceding the matching process automatically adds conceptual structure to the text, which may introduce conceptual problems beyond the mere authoring of the representations that are to be matched. The use of (previously created) resources for analysis or indexing (for example, lexicons) adds to the risk that representations may be introduced that do not mean 'sufficiently the same' to the various communicating parties involved (typically, the querying party, the authoring party, and perhaps an editor or librarian).

Representations of queries, or of (more typically) indexes or full texts can exist on many levels and take many forms. To name the most important ones:

- systematically defined categories assigned to documents
- open keywords assigned to documents
- words/phrases from the titles of documents
- words/phrases from the abstracts of documents
- words/phrases from the actual text of documents
- complex syntactic representations (sentences and analyses thereof) derived from document texts
- representations of meaning underlying elementary linguistic items occurring in documents
- representations of meaning of complex linguistic items occurring in documents

Representations of meaning beyond actual forms taken from texts (the latter two of the levels listed above) are, however, quite rare and mostly still experimental in

nature; most IR techniques are still based on processing and matching of word forms, phrase forms, etc.
Various approaches are used in IR to achieve indexing, textual analysis, and above all the matching of representations. To mention a few prominent ones: rule-based computational linguistics, representation-based pattern matching, statistical comparison of the occurrence of particular elements in texts plus their position with respect to other elements (Grootjen, 2001) ; also numerous learning algorithms and training thereof through annotated corpora. IR increasingly involves the application of multiple techniques (hybrid approaches). How successful linguistics-related techniques are is a point of debate, mostly in the light of the relatively greater success of purely statistics-based methods, but it seems that at the least some basic linguistics-based techniques have become a standard part of hybrid approaches. As mentioned before, automated symbol processing falls outside the focus of this thesis, and we will therefore not go further into IR techniques.
Explicit, communication-oriented conceptualisation as performed by human agents does not play a very important role in IR. In relation to automated approaches, lexicons are used regularly, but these are mostly derived from large corpora by computational means (Ahmad and Rogers, 2001). Manually authored lexicons including meaning descriptions are rarely deployed in operational IR. Implementation-oriented conceptualisation represents the mainstay of conceptualisation in IR. Analysis-oriented conceptualisation plays no role to speak of.
If we look at the creation and representation of queries, simple key word search is still most common. Especially the explosive growth of the use of 'search engines' on the WWW has made the use of (combinations of) keywords for searching widespread. Interestingly, users appear to dislike the use of more than a few keywords and complex queries, even if engines allow or even encourage this. Though because of its simplicity it is useful enough, keyword search apparently still has its limitations in terms of effectiveness, even if it is enhanced by means of techniques such as *relevance feedback* (Salton and Buckley, 1990), *query by navigation* (Bruza and van der Weide, 1992; Papazoglou and Hoppenbrouwers, 1999; Hoppenbrouwers, 1998), or techniques for constructing search strategies (Strehlow, 2001). This is partly due to the plain fact that words can mean many things (Chapter 3).
A possible solution to this problem could in theory be to create and maintain a 'personal ontology' and/or 'personal lexicon' for every individual user: a repository of conceptual description, matching the individual's concept use, that accompanies every concept used in a query. Such personalised, semantically or even pragmatically enriched querying currently belongs mostly to the realm of science-fiction, but concept management and explicit conceptualisation (probably in combination with some intelligent, learning computational system) might eventually make it possible (see Section 8.2.5).
As for conceptual resources used in IR: apart from lexicons for automated indexing and filtering, the devising of thesauri and ontologies for classification should be mentioned. In case a highly domain-specific classification is concerned, conceptualisation more closely resembles information analysis in that the selection and meaning of concepts becomes a matter of local preference.
Traditionally, the typical field dealing with document storage and document retrieval is *library science*. Library science evolved before the widespread introduction of

computerised information systems, but has now embraced them. Its usual approach is to *categorise* documents by means of codes and keywords. It also makes much use of abstracts. Document collections are *indexed*, and provided that a query is phrased in terms similar to the terms in the conceptual framework used to create representations of the target document (the index, title, abstract, or even the full text), a useful match between query and target will occur (that is, if a corresponding document is indeed present).

A common way of categorising documents involves the use of a thesaurus, a taxonomy-like structure, usually representing the topics distinguished for a particular collection of documents. Thesauri can be seen as 'descriptions of documentary languages' (Roulin, 2001). Thesaurus construction as it currently occurs is a typical form of ex ante conceptualisation. Large lexicons and *semantic networks* have also been introduced (for example, see Miller, 1995), though again, advanced language-based representations of meaning (beyond form) are rare.

Arguably, an exception to this is (once more) the use of advanced ontologies, which sometimes include some form of meaning expression beyond mere language form. Ontologies have raised a growing interest in particular in combination with web-oriented retrieval techniques. Various large, cooperatively inspired efforts have been initiated to devise combinable ontologies for many specific domains, for example the "Semantic Web" (Berners-Lee et al., 2001). Note that its goals include IR, but also go beyond it (artificial intelligence). The Semantic Web effort is a clear and rather unique example of large-scale, cross-domain language freezing (but see Kotok, 2002). It is also a typical example of representation-oriented conceptualisation, and as such we expect it to be limited in its success unless matters concerning conceptualisation-as-a-process are included in its setup.

Many definitions of conceptual structure or other content-related data depends on *meta-data*. Various meta-data sets for documentation have been suggested, prescribing standard categories for information relevant to IR yet distinct from the primary conceptual contents of a document. For example, the Dublin Core (Weibel et al., 1998) provides standard meta-data categories for author, origin, language, etc. of a document.

Finally, a word about standards. Though the limitations of fixed category systems and other forms of frozen language in IR should by now be clear, the field allows quite some opportunities for standardisation. Conceptual standards can be, and have been, most successfully introduced in relatively controllable conceptual domains such as digital libraries, where 'local standards' are achievable (for example, a thesaurus used for a university library), or where particular functional domains of meta-data seem sufficiently stable and common.

About explicit, manual conceptualisation in IR, including meaning description, we can be brief. In contemporary IR, it hardly occurs at all, least of all in a systematic manner, except in advanced but small-scale experimental language-oriented systems. A further exception is thesaurus creation and maintenance (Roulin, 2001), though here too, computational techniques are often deployed in addition to or replacing manual concept description.

As discussed, IR is strongly related to efforts concerning the gathering of large repository of linguistic/lexical knowledge (including ontology and thesaurus extraction; on the latter subject see Roulin, 2001). This subject falls outside the scope of this

thesis, but interests and applications are essentially related. We will briefly return to automated derivation of lexical resources in Section A.6.
To summarise, conceptualisation is a crucial activity in those IR efforts that at some point involve the creation of natural language representations designed for re-use. Query composition using open language is excluded from conceptualisation in this sense, but explicit conceptualisation in IR does include:

- specification of standard, 'prefabricated' queries
- specification of elements to be used in 'closed' query composition
- specification of thesauri
- specification of any category system used for indexing
- specification of closed sets of keywords covering topics
- specification of lexicons used to analyse open text
- specification of ontologies, for various purposes

Note that the activities above may include both the selection and the definition of concepts. Any techniques used for such activities (that is, non-automated versions thereof) are in principle relevant to our discussion of conceptualisation techniques, but once more only so if they go beyond the mere specification of a representational language or format in which a particular product of conceptualisation is to be authored.

A.6 Lexicography, Translation, and Terminology Management

There is of course a large and well established tradition of word description and definition in lexicography. Dictionaries representing many different approaches to language description have been widely published and used for centuries. Below we list some of the different possible goals underlying the publication of dictionaries. Many dictionaries combine various of the goals mentioned.

- to list common words a national or regional language for basic reference and clarification (descriptive dictionaries)
- to make clear what 'proper' language is (normative dictionaries)
- to describe historical differences and developments within a language (historical dictionaries)
- to explain the origin of words (etymological dictionaries)
- to support first and second language learners (learner's dictionaries)

- to support translation between particular languages (bilingual or even multilingual dictionaries: either learner-oriented or translation-oriented)
- to explain words from specialised branches of language (from countless professional languages to colloquialisms and languages of sub-cultures)
- to describe the meaning and/or the format of data elements

Note that we included 'data dictionaries', though this form of dictionary typically belongs to Section A.1. The existence of data dictionaries, and their relation to lexicography, is recognised in particular in the field of terminology management (Wright and Budin, 1997), and we view them as in principle strongly related to dictionary writing; perhaps even more so than to systems engineering.

Literature on lexicology and terminology (for example, Lipka, 1992, or Sonneveld and Loening, 1993) mainly focuses on descriptive devices and meta-concepts: how to adequately describe concept form and meaning. While this is a crucial issue in linguistic investigation, we have made a particular point of not getting deeply into it in this thesis (for example, see Sections 2.1.1 and 3.2.7). So unfortunately, the field of lexicography is not a very good source of detailed knowledge concerning explicit conceptualisation processes.

In (Viskil, 1994), the topic of definition is approached from a functional linguistic perspective and includes a detailed step-by-step description of the definition process. Viskil's work is rather theoretical in nature. Its most relevant aspects were elaborately discussed in Section 5.2.7, and his findings do not add much within the focus of the current section.

The goal of *translation* is to enable communication between parties speaking different languages. A translator replaces a representation (in the source language) by another (in the target language) which hopefully has the same conceptual load. Translation, and also its 'run time' counterpart *interpreting*, could be seen as very extreme examples of mediated meta-communication.

We pay special attention to translation and translation-oriented dictionaries and terminology control, for two reasons. First, translators (especially technical translators) are particularly closely involved with often minute differences between senses of words in combination with subtle contextual differences. This is of course highly relevant within our focus. Second, translation has developed into an industry, with its own practices and standards, and much operational experience with various forms of dedicated, explicit conceptualisation exists in the field.

In dictionary writing, explicit conceptualisation is not so much an activity that 'plays a role': it is *the* main activity. The writing of a dictionary is the blackboard example of explicit, communication-oriented conceptualisation. Traditionally, the writing of dictionary entries and definitions depends on the insight and skill of the author, and though it may require substantial and systematic inquiry into the occurring uses of some word, in the end it comes down to inspired and creative language use.

The occurrence of analysis-oriented conceptualisation cannot be ruled out completely in this field, especially in those cases where borderline between 'encyclopedic' knowledge and lexical/terminological knowledge is not drawn sharply. Implementation-oriented conceptualisation plays hardly any role, except if machine translation is involved.

As observed, the field of Translation faces the full and more practical challenge of dealing with detailed and specific terminology and interpretation across domains. When translation is involved, not only a particular concept has to be correctly interpreted, but also a conceptual form has to be found in another language, and its contextualised meaning has to be sufficiently similar to the original.

We need to distinguish between classic translation, which (even more than definition authoring) can be considered by and large an intuitive art rather than a well-understood task, and (semi-)automated translation, which involves computational technology and necessarily concerns fully explicit procedures and data (Vasconcellos, 2001).

Centuries of experience accumulated in the discipline of (manual, human-mediated) translation and interpreting make classical translators *potentially* one of the most valuable sources of knowledge about how people deal on a practical level with complex concepts and expressions thereof, and about ways of successfully matching conceptual representations across languages. Unfortunately, translation can indeed still largely be considered an art rather than a routine: the ‘knowledge how’ of translation as performed by humans is mostly implicit and notions like experience and talent play an important role in what constitutes a good translator. So yet again, little can be found on processes for explicit conceptualisation. Hence, we will not go into the process of classical translation as such, but focus on *translation support*, and the conceptualisation processes that may underly the creation and maintenance of *resources* for translation support.

We will also not go into the details of automated translation, expect for the observation that the manual crafting of lexicons for computational purposes (i.e. manually creating lexical resources) is both a labour-intensive and a difficult task, which should be tuned to the machine the translation approach followed (Vasconcellos, 2001). In many cases, automated translation efforts involving large language domains turn to computational techniques for the derivation of large lexicons from text corpora (Ahmad and Rogers, 2001; also see Section A.5). Though this approach lies outside our current focus, techniques for ‘manual’ and ‘automated’ description may well merge in the near future and complement each other (also see Section 8.2).

Definitions for translation support are mostly meant for use by an *expert party* (i.e. a translator), and therefore translation dictionaries fulfill exceptional requirements. Consequently, they are unfit for ‘common use’: the use of high quality, specialised dictionaries that list scores of senses of one word is not as unproblematic as is sometimes (and with disastrous results) assumed by laymen. The more senses of a word a dictionary gives, the bigger the skill required to pick the right one in context of the particular translation.

A crucial role in general terminological support as well as translation support is played by the practice of *terminology management*, which is therefore central to our current focus. An excellent and comprehensive overview of this field is provided by (Wright and Budin, 1997). Importantly, it is one of the few sources we encountered that explicitly connects terminology management with information management and even quality management (see Section 2.2.4).

Traditionally, dictionaries were compiled entirely by hand, with as a central aid a card system consisting of so-called ‘slips’ (of paper) that carried the entry for some lemma or sense. The actual process of filling in these slips was, generally spoken, hardly subject to regulation; it was 'result oriented'. Depending on the kind of dictionary under construction, however, the entries of course had a particular structure (comparable to a data structure), in line with the usage goals of the dictionary. Similarly, the nature of the *research* underlying the authoring of an entry depended greatly on the kind of dictionary produced. For example, compilation of a historical dictionary like the Oxford English Dictionary requires painstaking research of innumerable text sources, leading to verifiable dated quotations to back up attested uses of words through the centuries. Nowadays, the introduction of computerised support tools has radically changed both lexicographic and terminological practises (Sager, 2001). Corpora and analytical techniques from corpus linguistics have become central (also see Section B.27).

A substantial part of translation support, and the one of main interest to us, involves lexis (words, extending beyond terminology). The other main concern in translation is grammar, but we will ignore this subject insofar it does not relate directly to word meaning.

Concerning computational support of terminology management, various types of tools can be distinguished (Meyer et al., 1997); see below. A detailed description of technical aspects of thesaurus-related computational tools can be found in (Roulin, 2001).

Dedicated software for support of terminology management has its specific requirements and may be approached in various ways; for an overview, see (Schmitz, 2001). Experience (including our personal experience) has shown that the use of standard office software (databases, word processors, spreadsheets) for terminology management is a rather bad idea (*ibid.*, p.539): "[one] almost inevitably runs into problems involving compromised data integrity due to inadequate data modelling features, in addition to difficulties manipulating large volumes of data as resources grow over time".

Terminological databases are central to terminology management and translation support. In terminology banks, descriptions and definitions primarily aim for human understanding and usage, not for machine readability. Typically, terminology management requires description of the form of a word (e.g. grammatical category, spelling), of its meaning (by means of a brief gloss), and of cross-language synonymy where required, with as an extra some data concerning the context(s) in which a certain word is used as well as a collection of examples of usage. Translation support systems in particular often adds a repository of *all* particular occurrences of terms in a store of previously authored texts ('translation memory').

Many data structures for terminological databases have been suggested, but by now a reasonable degree of standardisations has been achieved (Wright, 2001a). For a study of lexical database management systems in relation to conceptual modelling, see (Hoppenbrouwers, 1997).

Database-centred computational support for terminology management and translation includes various aspects with their domain-specific functionality and requirements (Schmitz, 2001; Meyer et al., 1997): terminology-oriented browsers and viewers, editors, comparative views on databases, instruments for retrieval (Nkwenti-Azeh, 2001), formal structure and consistency checkers, and text markup formats that help terminology extraction or enable link term bases with texts. This thesis is not the place to discuss these devices at length.

Appendix B

Description of Conceptualisation Techniques

Below we provide a brief description of 29 conceptualisation techniques. For a more elaborate description of most of these plus a more in-depth discussion of their stronger and weaker points, see (Hoppenbrouwers and Weigand, 2003).

B.1 Selecting Artefacts

It is a widely accepted practice to take existing material as input for conceptualisation. In most cases, NL texts of some sort are simply assumed to somehow provide useful input to further intuitive analysis (e.g. in Rumbaugh, 1991); in some cases, text analysis is seen as a crucial and standard phase of the process for which more elaborate, even prescriptive guidelines are provided (e.g. van Dijk et al., 1989; Nijssen, 1993; Kristen, 1994). We found only a few references to guidelines concerning the actual *selection* of raw material in the literature.
Text is most typically seen as an ordered fragment of NL (for example, function descriptions, forms, handbooks, procedures; see van Dijk et al., 1989, p.91), but it can take other forms as well (e.g. 'organigrams', business process schemas; *ibid.*). Perhaps the most important alternative to a plain, unstructured block of text is the *form*[1], and the commonly used technique associated with it is *form analysis* (e.g. Burg, 1997; Loukopoulos and Karakostas, 1995).
The KISS method for OO analysis very explicitly encourages the use of existing documentation as a basis for a conceptual model (Kristen, 1994, p102). An exception to the general lack of explicit guidance on text selection is (Nijssen, 1993), where it is very emphatically stated that the input documents used must be '*familiar*' documents. Nijssen very explicitly determines the *use* that may be made of selected raw material: the text is to be be 'read out' by a person familiar with the domain (see Section B.13).
A different contribution to selection of language artefacts comes from Ethnography. (Hammerstey and Atkinson, 1983) generally typify documents as ranging from 'informal' to 'formal' or 'official'. Incorporating general human culture in their scope,

[1]Here we mean *form* as in 'a form to be filled out' or 'a stack of forms'; not to be confused with 'the form of a circle', or 'a concept form'.

they mention "fictional literature, diaries, autobiographies, letters, and mass media products" (*ibid.*, p.129). These can for example serve to "sensitise concepts": "they can suggest distinctive ways in which their authors, or the people reported in them, organise their experience, the sort of imagery and 'situated vocabularies' (Mills, 1940) they employ, [etc.]". Documents are considered especially useful as complementary to input from interviews and conversations. In ethnographic study of technical domains, many technical documents are considered valuable. For example, (Gamst, 1980, viii), in his study of locomotive engineers, drew on a wide range of documents: "[...] rule books, timetables, technical manuals for use of equipment, and instructional, regulating, and investigating publications of many kinds [...]".

B.2 Creating Texts

A very basic form of creating raw material is plain NL text writing. Generally spoken, our interpretation of the term "text" is quite broad. Any collection of authored, meaningful conceptual structures is included; no particular narrative structure is required. Text creation is of course the general technique for creating language artefacts. A number of techniques mentioned separately below fall in this category (e.g. case description, scenario creation, diary keeping). In addition, the creation of diagrams could also be seen as the authoring of a text –though certainly a text with different characteristics, in particular an essentially non-linear form of presentation. Sometimes, restrictions are imposed to the use of language, for example restricted vocabulary use and the avoidance of complex grammar. For example, in (van Dijk et al., 1989, p.89), authors are instructed to use simple sentences with one noun and no conjunctions (e.g. *after*, *before*), in present tense. More generally, authors are instructed to at least carefully consider whether the vocabulary they use is appropriate, i.e. fits the domain and purpose of the text. However, it is rarely made clear what is meant with 'appropriate' or 'purpose'. Some more general advice is provided by (Rumbaugh et al., 1991): "It is best to get ideas down on paper before trying to organise them too much, even though they may be redundant and inconsistent, so as not to lose important details". Note that this neatly expresses the crux of the 'raw material' phase as opposed to types of more explicitly conceptualisation-oriented, detailed, and definitive analysis.

B.3 Creating Diagrams

A form of representation which is often complementary to NL text and that in our present context cannot possibly be ignored, is the diagram. Whether hastily sketched on a scrap of paper or carefully constructed (for example by means of a computerised drawing tool), diagrams can be of great help in conceptualisation, both as a general type of concept representation (in phases 4 and 5) and as a type of raw material (Phase 1). Diagrams more often than not depend heavily on NL words included in them (Hoppenbrouwers, 1997). Sketched diagrams, in particular fairly generic ones, are also used successfully as illustrations to NL text.

If the diagram language used is well-described, and the exact relations between (graphical) elements as represented in the diagram (the diagrammatic syntax) is clear-cut, diagrams will typically fall outside the realm of 'raw material' and belong to Phase 4 or 5 conceptualisation (see Section B.3); in our experience, care should in fact be taken that existing, 'foreign' diagrams are not simply immediately accepted as Phase 4 conceptualisations. They may be precise, but their quality and usefulness as seen from a domain specific, communication-related 'frozen language' perspective may be quite disappointing.

Nevertheless, diagrams can in principle be used quite well as raw material for conceptualisation just as well as NL text, indeed they are, in our view, a form of text. The main difference between NL text and diagrams is that the former is linear in nature (one-dimensional) whereas the latter is typically two dimensional, possibly even three-dimensional or four dimensional (animated diagrams). This may have consequences for later analysis of the material. It can therefore be useful to *verbalise* a diagram: 'translate' it to text (also see Section B.23). This may of course either be done as a complement or as a replacement of the diagram.

B.4 Describing Cases

Textual descriptions made for the purpose of further conceptual analysis often boil down to varieties of *case description*. They typically concern a (possibly fictional) example of some relevant situation. The level of detail may of course vary depending on the goal. However, description inevitably includes the presentation of elements, their properties, and relations between them. Also (depending on whether a temporal factor is included) which actions are performed by which actors, how this involves other elements, and perhaps even some cause-and-effect description. In information analysis and functional systems engineering, case descriptions are generally –an perhaps too readily– taken as the typical input text of conceptual analysis.

The KISS method (Kristen, 1994), indeed showing an implicit preference for case descriptions as a useful form of textual input, provides some guidelines at least for the style and basic scope of the case description: "The style of the textual description preferably is narrative. The description must explain 'what' an employee does [...] as fully as possible with activities that actually happen, with nothing left out" [...] The eventual aim of the textual description is to provide an insight into 'who or what performs which activity in the organisation on what objects'. Textual description is therefore continued until we have sufficient material to allow us to carry out a grammatical analysis." (*ibid.*, pp.102-3). Rumbaugh (1991) provides some related guidance: "The problem statement should state what is to be done and not how it is to be done" (*ibid.*, p.150).

Case descriptions are prototypical products of Phase 1, though they may also come out of Phase 2. They loose their direct relevance at the latest after being used as input to phase 3. The limitations of case descriptions for conceptualisation on a communicative basis should not be underestimated: case descriptions are typically authored from a rather high-level, organisational point of view and by nature will often include conceptual bias and (over)generalised concepts.

B.5 Writing Scenarios

Scenarios arguably are not unlike case descriptions, but they typically have a procedure-oriented focus. They are more intensely concerned with roles, tasks, and dynamic aspects of the case. This positions scenario writing closer to description of actual operations and therefore closer to actual communication.

Scenarios are often used to describe one particular kind of task. For example, they are often meant to describe a concrete activity of a user working with a computer system in order to achieve a particular goal (Rumbaugh et al., 1991, p170). In his approach, scenarios are *characterisations of users and their tasks in a specified context*. Dynamic modelling is not by nature fit to be expressed by just a set of (grammatically and textually independent) factual sentences; a sequential *story* has to be told, and a good start is to take this literally.

Scenarios are not always explicitly recognised as such: some other kinds of language artefact may share many properties of scenarios without being labelled as scenarios. A good example hereof is a user manual (*ibid.*, p.150), but also the use of various scenario-related diagram types is widespread (for example, UML activity diagrams; see Booch, Jacobson, and Rumbaugh, 1998, pp.257-). Note that though existing language artefacts may be classified as scenarios, they are acquired through artefact selection rather than scenario authoring as such.

Sometimes a scenario is specific enough to include fictional dialogues. In this vein, (Rumbaugh, 1991, p.170) provides the following instruction: "prepare one or more typical dialogs between user and system to get a feel for system behaviour" (also see Section B.15). Rumbaugh further recommends to first specify 'normal' cases, then exceptions.

An advantage of scenarios in general is that they attempt to capture a more or less complete context in which all relevant tasks, and therefore also communication about them, is systematically carried out. If this approach is extended with a reality check as a result of the scenario being acted out (see Section B.15), then scenarios represent a very strong basis for the exploration of communication in specific contexts (Phase 2).

In some cases, CASE tools offer some support for the structuring of the conceptualisation process along the lines of a scenario (Loucopoulos and Karakostas, 1995, p.49). This is a rare but relevant example of conceptualisation (sub-)process support in current practice.

The RESPECT handbook (Maguire, 1997) has much to say about scenarios. Its strongly user-oriented approach goes beyond text writing and includes the *acting out* of scenarios (*ibid.*, pp.112-4).

Generally speaking, scenario writing is a better basis for communication-oriented conceptualisation than its more static counterpart, case description. The process of scenario writing thus is considerably more fit for use in Phase 2 than case description, but it is also a very useful and often wielded instrument in Phase 1. Obviously, scenarios are excellent input for Phase 3, provided their focus matches the focus of the general conceptualisation process at hand.

B.6 Brainstorming

A generically useful technique for the creative rendering of text reflecting knowledge is brainstorming. It is particularly useful in situations where little explicit knowledge is available and new ideas are wanted, within the scope of some set objective. In the RESPECT handbook (Maguire, 1997, p.94), brainstorming is described as "creative idea generation by freeing the mind to accept any idea that is suggested". In its most typical form, it is a group method, which should deliver a "set of good ideas and a general feel for the solution area". However, brainstorming can certainly also be performed individually, and in that case is generally more efficient, and in fact produces more and better ideas (*ibid.*). However, individual brainstorming obviously misses the mark of representing communal ideas and knowledge.

Effective brainstorming in groups requires some organisation and a mediator, who sets objectives and orders the outcome of each round (for example through clustering), channelling the creative output of the participants. In some cases, the use of an Electronic Meeting Room may be helpful. But on the other hand, the simplest form of brainstorming, i.e. the straightforward production of a list of words or short phrases representing ideas, is quite powerful. In any case, results should of course be carefully recorded in some way or another (for example, even a chaotic, half-readable flip-over sheet may be turn out highly valuable; having to copy notes from a whiteboard can be quite cumbersome and may even hinder correct communication of concepts gathered).

(Loucopoulos and Karakostas, 1995, p.43) mention that brainstorming is especially useful in overcoming lack of group consensus. As such it has been effectively applied in the Brainstorming Collective Decision-making Approach (Telem, 1988).

Conceptualisation is not usually a primary goal of brainstorming; rather, intensive conceptualisation is a sub-activity of brainstorming. Rendering fresh but unpolished, indeed 'raw' material, brainstorming typically falls under Phase 1. As a Phase 2 process, it seems much less advisable because it draws on internalised conceptual knowledge in an associative manner, implying a considerable risk of its results being coloured by the informant's insights and understanding rather than the communicational practices in the domain.

A promising direct deployment of the technique, however, would be in harnessing the associative powers of the participants and thereby in finding related lexical items (synonyms, antonyms, etc.). Such an approach may help develop a tight conceptual network in areas where no means of systematic exploration of a semantic field or domain is possible. This specialised use of brainstorming as a technique might be applied in Phase 4.1, or even 4.3.2.

B.7 Keeping Diaries

The technique of diary keeping (as described, for example, in Maguire, 1997, pp.97-8) typically involves the recording of activities in which subjects are engaged throughout a normal day. The form of a diary may vary from open-ended textual description to highly structured, closed forms.

Diaries are a popular and successful instrument in the social sciences (Hammerstey and Atkinson, 1983, pp.134-5). A striking advantage of diary keeping in this context is that it may help capture aspects of the domain and the UoD that are exceptional rather than common. In addition, diary creation may cover relatively long periods, even years. On the other hand, when it comes to operational situations and procedures, it normally lacks the detail and comprehensiveness that may be found in scenario and dialogue description.

Diary keeping in social research is of course not directly aimed at conceptualisation. Perhaps the most obvious use of diaries in explicit conceptualisation concerns corpus gathering (see Section B.27), though such corpus material is not created with the corpus in mind. In either case, the activity involved typically belongs to Phase 1, possibly to Phase 2. However, in a different guise it may also occur in Phases 4 and 5.

If the diary author is skilled in conceptual analysis, diary keeping may be limited to taking notes about striking uses or properties of concepts in context as noted by the author/observer. This practice is common in the process of compiling large dictionaries and in lexicologically oriented ethnographic studies (Spradley, 1980). The diary then becomes a vehicle for the purposeful collection of notes on exceptional concept use. Even this more highly conceptualisation-oriented sort of diary keeping belongs to phases 1 and (possibly) 2. This approach differs from straightforward recording of communicational bulk in that the subject may actively select the occurrences relevant for recording. The process of gathering incidental, focused information can be succesfully aided by the use of *closed forms* or *templates*. The subject in question then writes down highly specific information, either observations or even explicit communications which took place in certain prescribed circumstances or settings. For more on template filling, see Section B.24.

All kinds of diary keeping mentioned may benefit from the use of electronic forms or digital data formats, but this is especially the case if the diary aims at recording occurrences of a certain type of setting or task which is part of an information system. If a specific task is made explicit in the system (i.e. the system is aware of the task), it may even trigger a request for diary entry or copy a particular message as part of the diary entry. Similar techniques may be used in corpus analysis, but in diary keeping reflections or observations of the subject may be included.

B.8 Conducting Interviews

Perhaps the most common, intuitive way of gathering information in cases involving analyst-informant interaction is the interview: a conversation in which one partly asks questions and the other answers them. Good interviewing, however, is almost an art form, just as much as good writing. What triggers the interviewee to express a particular bit of previously hidden information is often unclear, and has much to do with the interviewer's feeling for extreme detail concerning the interviewee's thoughts, interpretations, and emotions.

Interviews are in general a very popular instrument for knowledge elicitation (Firlej and Hellens, 1991). But also in (Maguire, 1997, pp.104-5), in context of requirements engineering, refers to interviews as the "most widely used and abused method of finding out what users want". They point out that there is a misunderstanding that "everyone can conduct an interview because every one can hold a conversation". We suspect that this is an important reason why interviews are ever so often the only means of analyst-informant interaction considered.

The primary parameter in differing styles of interviewing is that of open versus guided approaches. The less the interviewer uses a predetermined set of questions, and the more she lets her questions be inspired by the developing conversation and therefore lets the interviewee influence the discourse, the opener an interview is. Guided interviews seem more efficient, and in some cases they are, but the flexibility of an open interview creates better opportunities to discover truly new things since questions need not be concocted beforehand. In the end, the skill of the interviewer in striking the right balance between 'open' and 'guided' will to a large extent determine how successful an interview will be. "In an elicitation context, the semi-structured interview is generally most fruitful" (*ibid*).

It has been observed, especially in ethnography, that interviews are perceived by the interviewee as an artificial setting, and that this may seriously influence the its value in data acquisition. Interestingly, (Hammerstey and Atkinson, 1983, p.119) acknowledge this limitation, but emphasise that "the distinctiveness of the interview setting must not be exaggerated, and it may be viewed as a resource, not simply as a problem. [...] To the extent that the aim in ethnography is not simply in the provision of a description of what occurred in a particular setting over a certain period of time, there may be positive advantages to be gained from subjecting people to verbal stimuli different from those prevalent in settings in which they normally operate."

(Loucopoulos and Karakostas, 1995, pp.42-3) explicitly mention that vocabulary poses a large problem in interviews. We have already mentioned the importance of the interviewer being sensitive to subtle variations in interpretation; if the culture and domain of interviewee are insufficiently known by the interviewer (as is very often the case, the purpose of the interview usually being to change this very situation), interpretation risks are high. It is good practice to have the interviewee check texts resulting from the interview so they can weed out obvious miscommunications, but this is not always possible and the risk cannot be entirely avoided.

Also in the field of requirements engineering, (Wood and Silver, 1996, pp.51-74) provide an excellent overview of practical matters involving interviewing. Among other matters, they emphasise that interviewees may not only speak a language different from that of the interviewer: their whole conception of problems faced is usually quite different. They include the following illustrative quote from (Wetherbe, 1991):

> Systems analysts assume managers surely know what information they need... The problem is that this technique is akin to a psychoanalyst talking to a patient lying on a couch and asking, "what type of therapy do you need"?

In addition, consider a well-known phenomenon in knowledge acquisition. If people are asked about something they know, they tend to reproduce explicit verbalisations they previously encountered ('book knowledge'; for example, material they studied in training or education, or observations as phrased in a technical report). They may fail completely to verbalise the more applied knowledge most relevant to the particular focus of the interview and its underlying knowledge acquisition effort.

Interviews belong to phase 1 and 2 conceptualisation, though in a very restricted context they may also be used in Phase 4. How useful the outcome of an interview is in a conceptualisation context depends on the ability of the interviewer and interviewee to *focus* on matters of language and communication. In this respect, it seems possible for an interview to be quite open in structure and at the same time focus very much on concept-related matters. To be useful in a Phase 2 context, such correct focus is of paramount importance. In line with this, it turns out that open interviews are more easily conducted with well-informed and well-educated interviewees (Hammerstey and Atkinson, 1983, p.119), since these are better capable to 'sense' the focus of the interview and keep to it.

B.9 Sending out Surveys

A survey involves the administering of a set of written questions to a large sample population. There are two types: 'closed', where the respondent is asked to select from available responses, and 'open', where the respondents are free to answer as they wish (Maguire, 1997, pp.117-8).

There are parallels between interviews and surveys, but an important difference is that surveys usually involve a number of interviewees, often even quite a large number, who all answer the exact same questions. Because large numbers are involved, statistical methods normally play a role in the preparation (user sampling) and analysis of a survey (calculating averages and drawing conclusions from closed questions). However, as soon as more open questions are concerned, the survey results will have to be processed and interpreted by hand, and this may involve a substantial amount of work. The obvious advantage of involvement of a number of interviewees is that general opinions and patterns within a whole group or community can be investigated in a scientifically acceptable fashion. A survey "can provide a quantitative edge to qualitative research" (*ibid.*). A slight disadvantage of surveys in comparison to interviews is that especially in anonymous surveys, interviewees cannot be asked to comment any further on answers given.

Useful though they may be for various other purposes, the survey is one of the least useful ways of conceptualisation in environments and situations central to our focus, even in Phase 1. However, for conceptualisation in large domains, for example aiming for standards or general dictionaries (involving lexical definition), it may provide a substantial amount of conceptual information that can indicate common or average usages. In this respect, we can imagine a survey being used as a complement to corpus research. For Phase 2 conceptualisation, the survey is a technique we would have to advise against. Given the conceptualisation situation is indeed general enough, the technique may be used with considerable success in Phases 4 and 5.

B.10 Making Observations

Observation and related techniques take a special place in Phase 1 conceptualisation. It is typically only the observer who has the intention to perform conceptualisation, not the agents observed. From the point of view of the agents observed, the technique thus generally falls under Phase 1, whereas from the point of view of the observer it may fall under either Phase 1 or Phase 2, depending on how actively the observer's interpretation influences the results in terms of capturing some envisaged UoD.

Only if the conceptualisation effort at hand requires exact copying of the existing UoD (a situation in which the medium system to be developed should fit the existing situation seamlessly), then output of Phase 1 can be used directly as input for Phase 3 (i.e. skipping Phase 2). It is important to distinguish between objective, even mechanical 'recording' of activity and communication (which borders on 'artefact selection' as discussed in Section B.1), and processing of the results of observation involving active interpretation, selection, and production of conceptual material by the observer.

The applicability of observation-related techniques in actual Phase 2 conceptualisation (not just Phase 1 gathering of material) depends on whether the situation in which 'language to be frozen' is to be used can in fact be 'made observable'. If the situation to be described does not yet exist, it may be *simulated* (for example, by means of role playing or techniques borrowed from 'Rapid Prototyping'; see Maguire, 1997, p.112). However, observation then becomes an auxiliary link in the conceptualisation chain and the effort may on the whole no longer involve 'objective observation of activities that are carried out by agents who do not have explicit conceptualisation in mind'.

Observation is a central technique in ethnography (Spradley, 1980). In this field an aspect of observation commonly emphasised and warned against is known as "the observer's paradox": it is quite possible that the fact that some activity is observed changes that activity, most typically by influencing the behaviour of agents observed. Things get even more complicated if *participant observation* is engaged in: the observer then becomes at the same time one of the agents observed and in a sense needs to "*treat [herself] as an informant*" (*ibid.*, p.76).

So as discussed, observation may be used as a technique mainly for Phase 1, but if the observer focusses on conceptualisation as such and actively selects and interprets concepts as part of the descriptive process, it may become a Phase 2 activity. In addition, if observation involves exact recording, material acquired may be used as input for Phase 3 conceptualisation.

B.11 Introspection

We might almost forget it, but probably one of the most popular techniques in thinking up conceptual material that is intended to represent some existing or envisaged set of concepts simply involves 'thinking about it', on a strictly individual level and with a clear focus on the particular matter of interest. Admittedly, this approach will usually hardly be one that is explicitly selected; it is perhaps rather the one default technique that remains if all others are discarded. Still, it seems worthwhile

to include it here and position it against other options.
Introspection is classified as a Phase 2 conceptualisation technique (not Phase 1) because we assume that even if the material that it produces is intended to be a careful rendering of some realistic sample of actual language use in some DE, it is produced by someone with conceptualisation in mind; otherwise, the resulting description might fall under typical Phase 1 activities involving the creation of texts.
We have to distinguish between general introspection concerning design issues (situational or not) and introspection on matters of explicit conceptualisation. The former type occurs widely and –if we accept it can occur even if left implicit– is involved in any development effort that somehow concerns NL-related data engineering. The latter type is far more specific and can only be expected to occur in people capable of focusing on language-related functionality of systems. If such people are also familiar with some techniques or meta-language for representing concepts (mostly to be used in Phase 4), introspection focusing on explicit conceptualisation becomes indeed both very real and widespread.
The benefits and risks of introspection as a Phase 2 conceptualisation technique are straightforward enough. It is highly efficient because it involves no meta-communication with other agents, and can be interwoven with other (possibly more technology-oriented) activities, even with actual system programming. In addition, from the point of view of the performing agent the resulting set of concepts will be reasonably coherent. On the down side, the results will of course completely depend on the knowledge, insights, and skills of the individual agent, are therefore likely (though not certain) to fail to reflect the linguistic conventions of larger groups, and language-related concerns may be hard to keep apart from other (technical) concerns and may be overlooked.
We might say that introspection may also be useful in all other phases of conceptualisation, yet then run the risk to confuse focused introspection with general thinking. However, Phase 4.3 does seem to be a particularly relevant phase for introspection. After all, the creation of some sort of meaning description of a concept, even if lexical definition is concerned, necessarily involves individuals focusing on their individual knowledge of the meaning and use of a word (both general and communal linguistic conventions).

B.12 Creating Think Aloud Protocols

Think Aloud Protocols (also called 'think aloud interviews') are a crossover between a structured interview and the observation of a task. The subject is asked to verbalise any thoughts concerning the task she performs, which are recorded or noted down by the interviewer. Depending on how the subject is instructed, the information she gives may actually vary from practically oriented comments on the tasks performed, to rather abstract, even associative clues reflecting 'what the subject is thinking' during the activity in question.
The technique quite typically belongs to the realm of knowledge elicitation (Firlej and Hellens, 1991; Verhoef, 1993). They are useful in "revealing the subject's problem solving strategies, validation of conceptual information gathered in less structured interviewing, revealing sequences of events, and revealing the accompanying

knowledge that a particular task requires" (Firlej and Hellens, 1991, p.112).
In our own experience, think aloud protocols are also useful to lay bare the attitude of a subject towards the task performed, i.e. whether she likes it or not, and in case a prescribed line of action is followed, whether she agrees with the way the task is to be performed; also, whether the tools provided for the task are satisfactory. Not surprisingly, negative attitudes and opinions are generally voiced more readily than their positive counterparts.
Think aloud protocols require some skill of the interviewee, who has to be able to voice her thought with some ease in order not to hinder the task at hand. On the other hand, the elicitor must understand the language of the interviewee relatively well: because the technicalities of the task are viewed at relatively close quarters, chances are that words are used that have not been encountered before by the interviewer. As the interviewee is better not interrupted too much by questions or directions, it is usually a good idea to debrief the subject afterwards and clarify unclear concepts. An additional advantage of the think aloud protocol is that the very act of performing it may help not only the interviewer and/or analyst, but also the subject, to understand the task and what is required to perform it. For this reason if for no other, a debriefing is a good idea: the subject may have come to realise that additional information can be provided.
Think aloud protocols are not so easily applied in cases where verbal communication is part of the task at hand, because then it may become unclear what is said to who: whether it is invoked by the elicitation technique or belongs to the performing of the task. In such cases, plain observation is safer.

B.13 Listing Examples and 'Reading Out Facts'

An interesting variation on case description is the example text, which is not just a description of some situation or activity, but involves a high level of detail and a very literal approach to fact description. The NIAM approach to information analysis (Nijssen, 1993) makes extensive use of this form, and a procedure for creating example texts (in fact combining various techniques) is suggested in some detail (*ibid.*, p.522):

1. *Visualisation* of problem descriptions provided by the user (making 'sketches'), thus helping her to conceptualise. If necessary, make several sketches and let the user pick the "clearest one". Note that the notion of 'sketches' should be taken broadly here and may include whatever text or diagram may be considered useful as a heuristic for visualisation.

2. *Instantiation.* Populate the chosen form (sketch) with various allowed facts and combinations by 'reading out' (i.e. creatively deriving) facts from the sketches provided. Determine an initial situation and create concrete examples showing the results of changes in that situation. Two examples are recommended per time-dependent situation. If possible, work your way back until the initial situation is returned to.

3. *Explicitness check.* Take care that undetermined elements become explicitly determined and can be 'read out' (verbalised). Note that all facts that are

verbalised become part of the example text, which thus provides at least one representative example of all instantiations (facts) of the UoD (or at least, the informants conception of the UoD) that can be subject to further 'fact type analysis'.

This 'visualisation and instantiation technique' is actually a mixture of informal diagram/text creation (step 1) and case description, with echoes of scenario writing (step 2), plus a check (step 3) concerning the coverage in step 2 of the informal sketch produced in step 1. What makes example texts as created by the above procedure special is their high degree of situational correctness, their comprehensive coverage of the situation analysed, the explicit mental processing involved. Note that even step 1 seems to involve Phase 2 conceptualisation rather than typical Phase 1 conceptualisation. Fact listings resulting from the technique seem particularly fit as input for Phase 3 conceptualisation.

Given our own definition of UoD, any technique that centres on the analytic and verbalising capacities of one informant lacks respect for the variations in vocabulary that may occur within the DE. This could be remedied by selecting more informants and comparing their results, but this we advise against because of the complexity of combining several complete domain conceptualisations. It seems better to take one authoritative conceptualisation as a basis and then somehow explore similar or related concepts that may be used in the domain, or even concepts that fail to be captured by the basis (see Section B.16).

Nijssen explicitly warns against the reading out of *commonalities* (e.g. "figure X depicts a family hierarchy diagram"; this concerns meta-information about the text), and (more importantly) *conclusions* (e.g. inferences made from the text instead of facts literally provided in the text). Note that these restrictions do not apply to the text as such, but to the process of interpretation and the phrasing of normalised sentences.

B.14 Using the Telephone Metaphor

A very effective technique (perhaps just a heuristic) for helping an informant to describe some situation strictly using *forms that can be realistically and effectively communicated* (i.e. capture an envisaged UoD) involves the *'telephone metaphor'*. It can be applied in many situations and in its simplicity is very powerful as a means to enforce a descriptive style.

(Nijssen, 1993, p.81) describes the telephone metaphor as follows: "The subject creating the description is asked to communicate the content [of a certain sketch] to [someone] who is familiar with the [sketch format] but does not have access to a fax machine to receive the actual picture." By invoking associations with an imaginary but familiar situation that requires relatively complete, concise and and well-phrased descriptions, the subject involved hits the right descriptive mode almost automatically. More technically phrased guidelines will not so easily 'switch on' the proper conversational mode in a speaker. The telephone metaphor can be used in Phase 1 and Phase 2, and possibly Phase 5.

B.15 Describing Dialogues

The discussion of the telephone metaphor in the previous section provides a nice introduction to a typical Phase 2 technique that has been touched upon in various other sections: the construction and description (possibly even recording) of dialogues. With this we mean the capturing of actual or fictional dialogues representative of those that actually occur in the domain analysed. Provided that the people uttering or writing them are knowledgeable of the domain and focus their effort correctly, the description of dialogues renders literal utterances as might later be supported by some medium system. If the dialogs involve closed language, they are either a solid way of testing whether the concepts selected 'work', or they may be the basis of further analysis (Phase 3-4) actually rendering a closed language specification.

Dialogue descriptions may be an integrated part of scenarios (Section B.5). In addition, various techniques involving role playing, but also mock-ups or prototypes of medium systems and medium system interfaces may be used to create a realistic environment for *acting out* a dialogue (which then needs to be *recorded*). If dialogue construction is conducted in an advanced enough fashion, it may provide detailed insight into differences between conceptual frameworks as used by particular stakeholders, and the conditions in which particular concepts are used (or not). However, in order to acquire such detailed linguistic information, substantial skills, capacity, and support for observation and analysis need to be available.

B.16 Seeking Conceptual Differences

Provided that some sort of case description, scenario, or dialogue description is available, it may be effective to confront people in a domain with fact descriptions or communicational utterances (presented perhaps even as as a conceptual 'reference model') and to ask them whether they would 'use the same words', or what their own phrasing of the utterance would look like. This technique may be used simply to trace homonyms and synonyms, but it may also uncover deeper underlying differences between the conceptual frameworks that individual people wield in the domain. Of course, whether or not these differences should be taken along in the final conceptualisation (Phase 3-4) then remains to be decided upon.

Though we have found no reference to this particular technique in the literature, it is hard to imagine it has not at some point been applied. The technique seems also fit for application in Phases 4 (concept description) and 5 (checking in hindsight), but its use in Phase 2 may prevent inconvenient surprises at an early moment in the conceptualisation process. In general, the technique seems quite useful for uncovering nuances in a domain that has already been charted. As such it belongs to Phase 2 conceptualisation, and typically complements other Phase 2 conceptualisation.

B.17 Applying Grammatical Analysis

If no explicitly formulated and stored conceptual representations are available for selection, analysis of natural language items is a serious alternative to ad-hoc, intuitive picking of words or phrases. We focus on explicit grammatical analysis here, but given that coping with grammar is a crucial part of the intuitive language capacity of humans, it is of course an instrument that will play an implicit role even in ad-hoc concept selection.

Grammatical analysis, as part of Phase 3 conceptualisation, seems to be restricted mostly to the field of Conceptual Modelling (covering requirements engineering, data and information analysis, and certain approaches to knowledge engineering) and is described sometimes in classic grammatical terms, sometimes somewhat differently. References can be found throughout the literature, for example concerning Entity-Relationship diagrams (Chen, 1983), NIAM/ORM (Nijssen, 1993; Halpin, 1995), KISS (Kristen, 1994; Hoppenbrouwers, 1997), (Frederiks et al., 1995), GRAMMARS (van Dijk et al., 1989), OICSI (Rolland and Proix, 1992); see (Burg, 1997, pp.141-4) for a good overview; also, in a less NL-oriented setting, in (Rumbaugh et al., 1991; Booch, 1994; Booch et al., 1998), and many other OO approaches.

Typically, a combination of syntax-based word categorisation and semantic categorisation is applied (semantic typing will be discussed in Section B.18). Techniques for grammatical analysis generally aim for identification of potentially relevant objects (nouns) actions (verbs) and attributes (adjectives) that may be reflected in some language artefact (usually text), but the analysis may be far more advanced.

To identify isolated concepts in NL text one has to draw upon the explicit marking mechanism ingrained in previously obtained textual material (Hoppenbrouwers et al., 1997). More concretely, grammatical marking concerns word order (for example, in regular English clauses word order is 'Subject-Verb-Object': as in "John kisses Mary") and the inclusion in words of certain morphological elements (for example, the inclusion of typical word endings such as "-s" for plural and "-ed" for passed tense). Because conceptual representations of the kind we are concerned with are rather firmly related to words (Chapter 1), it is no wonder that grammar-based techniques are generally aimed at identification of *word categories* such as nouns, verbs, adjectives etc., and with rather basic phrase structures that show words in a clear relation to each other, displaying *sub-categorisation* restrictions (Radford, 1988). More advanced syntactic analysis is generally superfluous.

Some attempts have been made to automate textual analysis for CM purposes. Automatic 'Part of Speech Taggers' (rule-based, statistics-based, or hybrids) already enable high quality word categorisation. More advanced approaches aim to include the parsing of syntactic structures (and semantic sub-categorisation derived from those), but run into the typical trouble encountered in any attempt at computational analysis of NL. Accuracy and robustness of automatic NL parsing still leave much to be desired.

A different problem in auto-analysis, and one fundamental to CM, lies in *concept selection*: as texts analysed get larger, they render vast numbers of potentially interesting conceptual elements. Whether these elements are relevant to the conceptual model aimed at cannot be determined automatically, and selecting relevant elements and adjusting them to the model is laborious and difficult. The KISS analysis process, for example, (especially when supported by the automatic NL analysis capacity of a dedicated grammatical analysis support tool called the Grammalizer), suffers from this phenomenon.

(Rumbaugh et al., 1991) provides some guidelines for coping with large numbers of identified concepts. The strategy followed is explicitly to first gather a fair number of potentially interesting concepts, and then weed out the irrelevant ones. For example, in order to achieve the latter, the following recommendation is given:

- Eliminate 'actions': some verbs do not represent 'structural properties of the application domain' but 'transient events'.

This is a very helpful instruction, which rules out such phrases as "system interacts with user" or "system accepts cash card". It basically rules out quite a few phrases which can be expected to be seen as straightforward and typical for the UoD by users and experts. Yet the distinction, though analytically valuable, is rather hard to make and cannot be easily supported by example-like guidelines and typical NL phrasing and questioning.
On a more general note, the weeding out of concepts from a large collection as gathered by means of grammatical analysis does not fall within grammatical analysis as such. The need for selecting identified concept forms is noted here mostly because it is a common consequence of the application of systematical grammatical analysis to texts.
In the end, even basic forms of word categorisation and identification of some specific grammatical constructs may be of great help in CM. As (Loucopoulos and Karakostas, 1995) phrase it:

> "Object-oriented analysis of NL provides the analyst with a simple mechanism for representing key concepts in the problem domain. Because the approach uses heuristic rules it relies to some extent on the ability of the analyst to apply the rules effectively, as well as on his or her familiarity with the analysed domain. [...] Nevertheless the simplicity of the approach makes its use as a first-stage requirements elicitation technique worthwhile".

B.18 Identifying Semantic Types

Closely related to grammatical analysis, a source text can be subjected to semantic analysis. With this we mean not an in-depth analysis of the possible interpretations of the text, but (less ambitiously) the recognition of *semantic types* at word level. Most typically, *objects* and *actions* are looked for. The close relation with grammatical analysis lies in the straightforward fact that in most cases, objects and actions are grammatically marked (typically, as nouns and verbs). However, many more semantic categories have been distinguished throughout the literature, and in many cases it is impossible to accurately and conclusively derive semantic types –even the straightforward ones– exclusively on the basis of grammar.
For example, take the various derivations that occur in English. Perhaps the best known example is *nominalisation*. Actions are usually clearly recognisable grammatically (as verbs), but grammatical categories do occur that are nouns, and yet in terms of semantics have actions lying 'beneath the linguistic surface'. Thus, the word "production" is a noun, but implies the presence in some domain of some action or process. Similarly, the English language includes *gerunds* (also known as

'present participles'): "the painting" may stand for either the continuous action of painting (as in "the painting went on for several hours") or the *result* of the action of painting (as in "the painting was stolen"). And more problematically, in the phrase "the blue-coloured vase", it may or may not be the case (or relevant) that someone at some point did some actual colouring. From the point of view of focused, ICT-related analysis, such considerations put the effort on a slippery slope, in particular if analysis is systematical; the number of possible actions identified simply explodes if all possible grammatical and semantic derivations in a text are taken into account. Nevertheless, various methods of information analysis (most notably the KISS method) include analytical steps involving derivations and 'hidden concepts' (Kristen, 1994; Hoppenbrouwers et al., 1997), and as long as these are adopted sparingly and wisely, they may be quite helpful. Automated application, however, raises serious problems.

Once we leave behind grammatical marking as an explicit source of information for identifying semantic types, we are out in the vast and subjective semantic landscape underlying language utterances. Semantic categorisation mainly involves the language-related division of the world in isolated chunks of meaning, varying from highly abstract and generic (categories such as 'entity', 'object', 'action', 'attribute', 'agent') to highly specific names for particular items types ('three-eights Gripley', 'claim assessor'). Basic conceptual instruments normally revolve around the higher level concepts, which constitute the basic meta-language of some modelling language. Actual modelling involves more specific semantic categories. Interestingly, conceptual modelling often involves immediate linking of rather general, abstract concept types (like 'object') to specialised categories (like 'three-eights Gripley'), without making explicit the intermediate level conceptualisation ('tool'; "three-eights Gripley is a tool is an object"). In isolated domains, semantic category systems may become highly idiosyncratic; their terminology can become difficult to grasp and hard to relate to more common conceptual frameworks.

The identification of semantic types as a technique is essentially based on the exploration of associative relations between distinguished semantic items, usually bearing a relation to observable or imaginable phenomena in some real world domain. Apart from grammatical considerations, it thus draws on people's knowledge and conception of the world, and even to the power of imagination. In this vein, (Rumbaugh et al., 1991)[pp.153-162] present a number of heuristics that may aid the OO analysis of source texts. For example:

- "Objects include physical entities, such as houses, employees, and machines, as well as concepts such as trajectories, seating assignments, and payment schedules" (*ibid.*, p.153)
- "Associations [...] include physical location (next to, part of, contained in), directed actions (drives), communication (talks to), ownership (has, part of), or satisfaction of some condition (works for, married to, manages) (*ibid.*, p.156)
- "Attributes are properties of individual objects, such as name, weight, velocity, or colour"

Rumbaugh explicitly hints at the use of 'knowledge of the world' for analysis, and the hidden quality of some analytic categories: "Unlike classes and associations, attributes are less likely to be fully described in the problem statement. You must draw on your knowledge of he application domain and the real world to find them." (*ibid.*, p.161). However, the domain knowledge he draws on blends knowledge of the user domain with knowledge of the technical domain. For example:

- Identification of events: "Examine the scenarios to identify all external events. Events include all signals, inputs, decisions, interrupts, transitions, and actions to or from users or external devices"

- "An entity that has features of its own within the given application is an object" (*ibid.*, p.162)

While such heuristics may be quite helpful, they may also cause unwanted mixing of analytical priorities and may hinder objective conceptual analysis in view of communicational requirements.
Importantly, conceptual frameworks already accepted within the DE (for example, ontologies) may be of strong influence to the semantic categories looked for. Reversely, selection of such frameworks should involve considerations of what their impact will be on conceptualisation practices (Section B.19).
The field of ethnography also embraces the exploration of semantic categories and category systems as encountered in a domain under investigation, which is basically seen as a reflection of its "cultural patterns". (Spradley, 1980)[pp.85-] describes in detail a method for ethnographic domain analysis based on semantic categories and relations, and resulting in taxonomies (*ibid.*, pp.112-). (Hammerstey and Atkinson, 1983) also discuss semantic category based analysis of cultural domains, linking it to general qualitative and quantitative methodology as applied to ethnography.
Semantic type identification thus typically belongs to Phase 3.1b conceptualisation, but it will usually be combined with Phase 3.1a conceptualisation.
We have to take care not to confuse the technique of identifying semantic types with the general enterprise of analysing conceptual frameworks, especially since the two are closely related. After all, the technique is restricted to the identification and selection of concept forms on the basis of them seeming relevant elements of language within some universe of discourse. The technique thus has two separate functions:

1. identify *potential* concepts to be listed for conceptualisation ('longlist')

2. identify *definitive* concepts to be included in a (frozen) language ('shortlist')

For the first function, criteria for concept identification will mostly concern higher level, meta-linguistic categories such as 'object' or 'action'. For the second, criteria will be harder to formulate, and more directly linked to considerations of communicative functionality, prominently including careful demarkation of the UoD involved. This will usually include description of the UoD in terms of community-oriented classification of domains (tasks, departments, etc.; see Section 4.2.1).

B.19 Procurement of Existing Collections or Standards

Especially if a DE involves the use of some sort of standardised terminology (or has strong links with such a DE), it may be a good idea to actively embrace standard concepts. What we mean by 'standards' is *previously described sets of concepts*, in other words, *products of deliberate linguistic meta-communication*. Clearly, identification of individual concepts has already been performed in order to produce some concept collection. The term 'standard' may be too strong in some cases; it is possible that a collection of concepts is not explicitly set as a standard, but just happens to be available and useful. We thus extend the general idea to any explicitly described, coherent collection of concepts.

If an actual, officially described standard is not followed to the letter (if available at all), there are of course less strict forms of conformity, for example embracing the terminology that professionals in some field use, simply based on experience and (linguistic) knowledge as present in agents positioned in that professional group. In the latter case, an approach based on phase 1-2 conceptualisation techniques is required, and we exclude such cases from the approach currently discussed. Also, it may be the case that a standard is *partially* accepted, but for some other part *adapted* for the specific needs of the domain. This approach will be discussed in Section B.25. Sometimes it is a fine line between standard adoption and standard adaption.

When it comes to selection of concepts that are part of an existing collection, it may either be the case that such a collection has already been identified, or that a definite choice yet has to be made. In both cases, the activity engaged in concerns *procurement* (ISPL Consortium, 1999). The question thus arises whether the requirements posed by the situation are satisfied by the product under consideration; if more products are available, the one best satisfying the requirements should of course be selected. Though procurement of terminological standards (but also ontologies, lexicons, or any other form of concept collection) involves a highly specific sort of evaluation, it is expected that general techniques for procurement are similar to those in areas like software procurement and even service procurement. In the latter case, a highly interesting possibility is the procurement of a subscription to (use of) concept collections that are being actively maintained by the servicing party.

We do not go into the many different factors that may be involved in the procurement of conceptual resources here. In the field of terminology management, this subject has been explored by (Budin and Wright, 2001). In any case, both functional and non-functional factors should be taken into account with respect to the concepts and conceptualisation involved. It should be clear by now that not only it may be hard to find a fitting selection and description of concepts for some domain, but also that the processes needed to further deal with such concepts should be considered.

It is possible that (use of) some collection of concepts is not so much selected as *forced upon* an organisation. For example, a sub-contractor may be obliged to conform to some conceptual standard set by the commissioner (e.g. adopt a certain data structure, ontology, or product classification system). In such a case, the game is

played reversely: factors set by the conceptual standard will have to be matched with ways of dealing with them. This may have serious results for conceptualisation and quite possibly even for general ICT practices in the subcontracting organisation. The succes of concept procurement depends on largely its ability to clarify at an early stage whether the product to be acquired will fulfills the basic functional requirements –allowing the procurement process to be stopped quickly if chances of succes seem low. After all, procurement as a process may be very expensive, even if it *results* in the acquisition of a 'cheap' existing conceptual framework.
Selection and procurement of concept collections is a typical Phase 3 activity. If a well-described standard is indeed successfully embraced and used, this will turn further conceptualisation (indeed, also preceding phases of conceptualisation) superfluous. Clearly, the over-all efficiency of such delegated conceptualisation is almost unbeatable. The catch is that cases in which this is possible are relatively rare. If further work on an imported collection is foreseen, the factors that play a role in procurement may change radically: the imported collection needs to fit the conceptualisation process rather than just the operational domain. If an adopted collection requires unforseen fine-tuning or maintenance, over-all efficiency may turn out disastrous.

B.20 Applying an 'Architecture Viewpoint for Conceptualisation'

None of the techniques described in this section so far have directly addressed the matter of *how to select concepts to be included in some closed language*. In particular, techniques, heuristics and approaches presented have not made clear *which criteria should be set* for selection of concepts as intended in Phase 3 conceptualisation. At best, we have shown how concepts can be made visible, and which techniques and procedures may support the application of criteria to a collection of concepts.
Given the fact that hardly any guidelines for selection of 'concepts to be frozen' can be found in the literature, we suspect that they can be better derived from the theoretical findings presented in Chapters 2-5. In Chapter 6, these findings are bundled into a coherent, well-structured framework designed for use in so-called 'architecture approaches' to the development and maintenance of information systems. Without going into explicit criteria, we point out that the use of such a framework (known as an 'architecture viewpoint', see van de Heuvel and Proper, 2002) can realistically be seen as the application of a technique. We thus consider how the technique of using a conceptualisation viewpoint within a systems architecture approach can aid the selection of explicit concepts. Importantly, this use should extend to dynamic mechanisms of conceptualisation, making it possible to include conceptualisation in an evolutionary approach to information systems.
An architecture viewpoint is a descriptive framework aimed at being an aid in analysis, development, and possibly management concerning *some particular aspect of an information system*" (see Section 1.3.4). Conventional aspects are application design and management, business process analysis, or general technological infrastructure. Our particular perspective concerns the less conventional area of conceptualisation for ICT-supported human-human communication (freezing language):

analysis, development, and management of closed language –and all factors and matters relevantly related to it.
A 'conceptualisation viewpoint', then, helps in analysing the ICT-related situation in which language is (to be) frozen, making possible rational consideration, evaluation, and possibly management of concepts and conceptualisation. The minimal use for an architecture viewpoint is for analysis. However, the intensity and depth of the analysis may vary greatly from situation to situation. The use of an architecture viewpoint can thus in principle vary from the application of some informally described best practices and good advice in the light of a sketchily analysed communication/conceptualisation situation, to implementation of a fully explicit, meticulously monitored 'conceptualisation support system' that guides and records every communication and conceptualisation activity in an organisation (for some related ideas, see Veldhuijzen van Zanten et al., 2003).
The main point is that an architecture viewpoint for conceptualisation enables us to *produce an insightful analysis of the language needs and requirements of the various stakeholders in a domain*, which can be *communicated to and validated by stakeholders*, and possibly *enables us to combine consideration of conceptualisation-related issues with that of other issues in system development and maintenance*. Thus, we are provided with an instrument to include matters of conceptualisation in a more comprehensive approach to thinking about information systems, in a way that is compatible with general ICT practices.
The 'architecture viewpoint approach' to conceptualisation relates in principle to all phases of conceptualisation if these are to be analysed or controlled. However, the selection of concepts to be included in a system (Phase 3) is the central activity of the whole conceptualisation effort; in fact, in many cases it is the only conceptualisation activity actually performed (along with Phase 4 'listing of concepts'). This is why it is also the central point of connection between general information system engineering and explicit conceptualisation as discussed in this thesis.

B.21 Sorting Out Homonyms and Synonyms

This technique is fairly simple in essence, and though the importance of recognising homonyms and synonyms (though usually as things to avoid) has been widely recognised, hardly anything is ever said about how one goes about doing it. Let us briefly recapitulate what homonyms and synonyms are:

Homonyms concepts are homonyms if they have the same form, but a different meaning

Synonyms concepts are synonyms if they have the same meaning, but a different form

Though many more nuanced accounts of the two relations exist, for our current purposes the explanation above will do nicely. Nevertheless, much of the nuance in theoretical discussions of homonymy and synonymy is related to two questions that are immediately raised:

1. When are forms 'the same'?

2. When are meanings ‘the same’?

As we discuss operational techniques, not theory, we take a pragmatic approach here: forms and meanings are ‘the same’ if they are considered sufficiently similar to be called so by a representative group of users, *in a particular situation of use*. In other words, one should ask the questions: are forms so similar that they will be recognised as referring to the same concept, or are meanings so similar that the communicative load they have in some situation is the same (has the same linguistic effect, i.e does not result in relevant differences in interpretation)? In cases of doubt, simple testing of homonyms and synonyms is possible: see whether concepts can both be used in the same context (in terms of either form or meaning) without relevant consequences.
Importantly, dealing with homonyms and synonyms in practice is mostly a matter of *explicit form*, not explicit meaning. This is straightforward enough for homonyms, but for synonyms it may seem surprising since they concern similarity in meaning yet difference in form. However, operations concerning both homonyms and synonyms are normally carried out on the basis of concrete forms alone, notwithstanding that the agents performing those operations make use of whatever interpretation they think is related to the form. If people are asked to collect synonyms, they will list the synonymous word forms, for example “client, customer, purchaser”. What it is that these words have in common in terms of meaning is only a secondary consideration that will usually not be made explicit. In most cases, what is deemed important is simply that they “mean the same”. Even in synonym-based lexicographical projects like WordNet (Miller, 1995), informal glosses are added only as an extra, whereas the basic conceptual structure links word forms with relations like ‘synonym’ and ‘antonym’.
Two different activities concerning homonyms and synonyms can be recognised:

- Identification of homonyms or synonyms (mostly a Phase 3 task)
- Deliberate creation/maintaining of homonyms or synonyms (Phase 4.1).

Identification of existing homonyms and synonyms is a matter of understanding similarities between linguistic patterns, usually as occurring in different language domains. Comparison is the key activity. For comparison of both explicit form and of meaning, complete representational similarity is not usually required for items to be considered ‘the same’. For example, differences in spelling or graphical representation (form) or differences in definition format or formulation (meaning) are not enough to declare a concept ‘different’. Consequently, comparison of ‘sameness’ should mainly be carried out on the basis of *active interpretation and evaluation by human agents*, based on ‘intuitions of sameness’ rather than similarity in representation. Unfortunately, this usually hinders the systematic recognition of homonyms and synonyms by means of computerised techniques, which are incapable of such ‘fuzzy’ matching unless they are very advanced; indeed, coping with this fuzzyness is a main challenge for computational linguistics and AI.
Once homonyms and synonyms have been recognised, they may or may not be deliberately maintained. In many CM-oriented approaches, homonyms and synonyms are seen as matters that need to be ‘resolved’ in order to reach a high quality conceptual analysis. This is because homonyms and synonyms may cause ambiguity

of representations, which in many design-oriented situations is indeed undesirable. However, the language situation in complex DEs, in particular inter-DE communication, may require a quite different stance, requiring homonyms or synonyms to be tolerated, even embraced. Here are some simple questions that should be asked in every case:

- Why would it be desirable to maintain a particular homonym or synonym?
- Would it be effective to exchange one of the concept forms in the set of synonyms by another form in that set, thus resolving synonymy?
- Would it be effective to choose a different word form for one of the two similar word forms in a set of homonyms, thus resolving homonymy?

Techniques dealing with homonyms/synonyms are particularly useful in situations in which within a DE, there is a need to respect various terminologies, or where functional gaps between them have to be bridged (emphasising the social factor). Alternatively, they may be of help if one strives for unambiguous concepts in view of optimal communicational functionality within one language domain (emphasising the practical factor). However, considering and dealing with homonymy and synonymy often is not a matter of choice, but a fact of life.

B.22 Writing Definitions

Definition authoring needs to be mentioned as a technique, but there is considerable overlap between the 'technique' and the 'process', which was elaborately discussed in Section 5.2.7. While we focus on operational aspects here, it is recommended that the matters discussed in relation to the sub-phases of Phase 4.3 are considered alongside the aspects mentioned here.
A note on representation of (meta-)linguistic meaning is in order here, even though we avoid further discussion of this immense topic (as explained in sections 3.2.7 and 5.4). The main point is that once we do engage in meaning description, we at least have to be aware of the conceptual framework and semantics we want to apply. This is hardly a problem in environments that have some sort of tradition or standard concerning definition authoring. However, once a number of different language domains is concerned, the way a definition is constructed and represented may be less straightforward, and deserves due consideration (Phases 4.3.1.2-4; for more on this see Section 5.2.7; also Viskil, 1994). Overviews of basic meta-linguistic concepts are provided in (Cruse, 2000; **?**; Sager, 1997). A general overview of styles of terminological definition can be found in (de Bessé, 1997).

Creating Informal Definitions

The writing of definitions in natural language (though possibly with the aid of graphical representations; Galinski and Picht, 1997), is an activity that is deeply rooted the traditional field of lexicography, but also in many professionals fields (e.g. legal and medical practice, science, and increasingly also ICT). It generally results in brief texts aiming to optimally describe the meaning of some word or phrase. What distinguishes it from raw material is its well-developed focus on meaning description and its occurrence in a strict meta-communication context. Traditional textual definitions lack the formal, to-the-point conciseness and careful abstraction of definitions in logical or mathematical definitions. However, it compensates for this because of a different kind of formality which boils down to a great (though sometimes lengthy) precision of formulating, reached through optimal use of the intuitively descriptive and associative force of natural language, drawing on a tradition which is age-old and widespread.

In ICT, perhaps the most typical use of definitions is in data dictionaries (Wright, 2001c, p.472). A good impression of the envisaged use of data dictionaries is given by the following quote from (Rumbaugh et al., 1991, p.156):

> "Individual words have too many interpretations, so prepare a data dictionary for all modelling entities. Write a paragraph precisely describing each object class. Describe the scope of the class within the current problem, including any assumptions or restrictions on its membership or use. The data dictionary also describes associations, attributes, and operations."

In all its variation and versatility, informal definition writing is an activity that should be approach with care and that should even be avoided where possible. It takes meta-communication to a level that is fraught with linguistic and even philosophical pitfalls that non-academics may not be aware of but that may nevertheless be the source of fierce debate and tragic misunderstanding. Much depends on the goal served by a particular definition (see Section 5.2.7).

High cost is probably the greatest downside of definition authoring. Though meaning description may be useful or even vital in some situations, the thought of having to define a number of definitions usually creates an instant feeling of weariness with agents involved and demands an answer to the question: "can we afford to do this"? But more importantly, the *maintenance* of definitions can be a task that requires substantial resources. We believe that this point in particular is responsible for the fact that definition, *if carried through with an open eye to functional quality of the results*, can be very expensive. An interesting way of optimising meaning description is *stepwise lexical decomposition* (Dik, 1989). This approach involves definition of words by use of other words. This recursive process continues until the parties feel sufficiently confident about the definition (in our own terms: 'semantic reassurance' is achieved).

Creating Formal Definitions

Roughly speaking, formal definitions differ (or should differ) from informal definitions in the following ways:

- The meaning of the (meta)language in which they are expressed is much better described
- The meaning of the definitions is expressed by means of a coherent, logical system of symbols that can be reasoned about, and which is 'completely under

control'; for example, within boundaries of formal specification, propositions may be proved correct.

Formal definitions thus do not differ all that much from informal definitions in essence, it is just that they are much more carefully defined, within a conceptual framework that (if constructed properly) leaves no 'loose ends dangling' in terms of meaning. Once formally captured, meaning can be handled with the same kind of operational certainty as a well-designed machine. However, with the advantages this has, some disadvantages come in:

- It takes considerable effort and intellectual capacity to express oneself in formal language (typically, some form of logic or mathematical language, or a derivative thereof –including programming languages)
- It takes even more effort and intellectual capacity to develop ideas and concepts to a level that allows them to be part of a formal framework: demands on conceptualisation are high
- Various modes of human communication and thinking do not readily fit formal thinking; to the contrary, linguistic 'underdeterminedness' can be highly functional (for example, if priority is given to understandability by a large, heterogeneous public)
- Even the 'well described meaning' of formal representations depends on elements the meaning of which may need to be described. What is controllable about formal language is mostly its syntax; in the end, we can always ask: 'but what does X *mean*'? Providing a logical representation is rarely the final answer once we look outside the formal system as such and try to interpret it in view of the world at large

For actual techniques of formal definition, we refer to the enormous body of literature on formal languages. Interestingly, attempts are made to not only provide formal languages, but also operational tools that help agents formalise and reason about (even prove) statements. Such tools certainly make formal languages more accessible to non-specialists, but currently the degree to which this is actually the case remains limited.

Creating Diagrams as Meaning Descriptions

Both informal and formal definition can take the shape of diagrams as opposed to 'linear representations' of either NL or some formal language. Perhaps 'definition' is too strong a term here, but meaning description through diagrams certainly occurs. For a general discussion of diagrams in conceptualisation, we refer to Section B.3. Like linear representations, the syntax and semantics of diagrams may or may not be formally described. It is also possible to translate between linear representations and two, three, or four- dimensional representations (see Section B.23). Thus, the linear-diagram distinction is disconnected from the formal-informal distinction in representations.

We should not confuse general diagram creation with the creation of 'definition diagrams'. Yet as long as the purpose of a diagram is to specify the meaning of something that can be reasonably called a 'concept', the notion of 'definition' indeed seems to apply –even if in one diagram a number of interrelated concepts are described, which in context of linear representation is comparable with either a text built up of sentences and thus concerning complex concepts (relevant to Phases 1 and 2) or a structured listing of isolated (meta-)concepts (relevant to Phases 3-4).
A substantial amount of explicit conceptualisation indeed takes place directly through the construction of diagrams. What makes a 'definition diagram' different from a text is this:

- Its layout (two or three-dimensional instead of linear) provides more options for more meaningful intuitive positioning of linguistic elements, e.g. groupings, degrees of relatedness (distance), hierarchical orderings (top-bottom/left-right)
- There is literally more space for making relations visible. For example, from one central element, a substantial number of related elements can be shown to 'branch off'; the linear equivalent perhaps being a 'bulleted list'

As an example of the latter property: within the ICT community at large, relational definitions in a purely terminological context are quite commonly and successfully represented by means of simple ER diagrams, or less commonly NIAM/ORM diagrams. In other respects, diagrams describing meaning are subject to the same principles as linear representation. In addition, they often include completely linear definitions, simply positioned within a graphically laid-out frame. Let us therefor focus on what makes diagrams special: they open up possibilities for *spatial manipulation* of conceptual representations. Though established drawing technologies (from whiteboards to pencil and paper) are still used widely and successfully, the drawing of models and the diagramming they usually entail is increasingly supported by computerised drawing tools, some of which are also intelligent enough to have capabilities for reformatting, the combining of schemas, verification of syntax, etc. Such support greatly increases the efficiency of diagram construction as well as the possibility to deal with complexity without loosing overview. A drawback of such support is that the flexibility of computerised drawing tools, which are often language-specific, leaves something to be desired, especially if there is no immediate need for controlled, formal diagramming according to some paradigm or method. In many cases, diagram-oriented languages such as the UML (Booch et al., 1998), supported by dedicated drawing aids, are the core of CASE tools (Section 2.3.1).
An interesting form of diagramming support which has caught our imagination because its effectiveness and simplicity in helping creative conceptualisation is what we might call 'hands-on diagramming'. We use this term for all techniques which make it possible for a person or group to freely manipulate basic diagram elements (both their position, name, and attributes) so that instant ad-hoc experimentation with conceptual combinations can occur, as well as group feedback. Though electronic screen-based implementation of this kind of modelling support is possible in principle, in practice we see that most implementations still opt for the use physical objects of distinct, easily recognisable types placed and moved on a two dimensional surface, labelled with names and attributes. Even ordinary yellow "PostIt"

notes function quite well as and aide for improvised hands-on diagramming of the sort discussed. Though 'hands on diagramming' techniques do not support very advanced graphic diagramming features, they serve very well where sketch-like but nevertheless substantial diagramming is required, especially when this is done in groups. The technique seems especially popular with ER-like diagrams and Object Oriented modelling of class diagrams and related entities. The KISS method (Kristen, 1994) includes a game-like support product for the process under the name of 'KISS Domino'. The game is simple in principle, but very effective in stimulating creative modelling in small groups.

B.23 Mapping between Linear Representations and Diagrams

A frequently occurring activity in approaches incorporating a relationship between a linear and a diagram language is the mapping (the 'translation') of linear structures to diagrams and *vice versa*. This practice is particularly popular in Conceptual Modelling. In many CM approaches, the final goal of analysis efforts at large is the production of diagrams, not linear structures; thus, some include explicit instructions that guide the analyst from creating linear, usually NL-based representations to the mapping thereof to diagrams; others leave this step implicit.

The ideal that is often strived for in explicit approaches to mapping is to achieve *'1:1 mapping'*. However, 1:1 mappings from linear to diagram representations can only be achieved if specialised, normalised sentences are used, for example expressing streamlined relational structures, or well-ordered chronological accounts. This requires careful preparation of text; in fact there is little difference with preparation for direct diagram authoring. Note, therefore, that contrary to what some approaches claim, diagrams can never be derived directly from general NL descriptions, i.e. texts that are not carefully prepared or at least have quite specific structural and conceptual properties.

For example, the KISS method maintains a strong relation between its 'Object Interaction' diagrams and NL; *once structured sentences have been made available* (and if existing raw material is used this involves both normalisation and selection of such sentences), a rudimentary diagram can be easily constructed and even generated. However, KISS also includes quite a few diagram types ('models') which cannot possibly be derived directly from explicitly marked NL structures as found in general descriptions (Hoppenbrouwers et al., 1997).

Perhaps somewhat more realistically, a 1:1 mapping between (prestructured) 'e-sentences' and the NIAM schematic language is explicitly supported (Nijssen, 1993, p109). He calls this step 'Make Diagram'. Nijssen insists that *e-sentences in fact are conceptualisations.* We quite agree with this; from a general point of view, the distinction between 'e-sentences' and 'conceptual representations' is a heuristic device and has little theoretical value. The following quote is rather typical (*ibid.*, p.56): "Information has many forms, but the universal form consists of NL sentences". On p.104, Nijssen emphasises that the NIAM schema language "does not necessarily have to be learned by users", which implies that they could in principle get by by using nothing but e-sentences. However, the schema language is thought to be "quicker"

in use.
Nijssen provides a highly detailed description of the mapping of e-sentences to diagrams (*ibid.*, pp.114-20) which is rather unique in modelling literature. It is, however, very complex. In fact it includes the kind of instructions which, through intensive training, help analysts to *internalise a way of working*. It seems doubtful whether they could be used as operational 'cook book' guidelines for non-expert users. Thus, anyone capable of executing the procedures is thereby an expert. We observe that to provide detailed, comprehensive rules for mapping normalised NL structures to advanced diagrams is difficult to say the least, and that systematic, 'mechanical' applicability of such rules has not been convincingly proved in a practical context.
If linear structures (typically a normalised form of NL) are produced on the basis of diagrams, this usually has the goal of communicating the conceptual structures underlying both types of representation to a party that has difficulty understanding diagrams (Kristen, 1994; Frederiks, 1997). Such 'verbalisation' thus makes it possible for an expert without knowledge of the diagram language to evaluate or validate the conceptual structures in question. However, opinions seem to vary concerning which representations are more intuitively understood by stakeholders: diagrams or NL like-representations. In our view, this matter cannot be conclusively resolved at this point.
KISS not only explicitly recommends the 'verbalisation' of Object Interaction models to gain clear, transparent feedback to stakeholders, it also provides a dedicated tool to automate the process: the 'Verbalizer', which is positioned as the counterpart of the earlier mentioned Grammalizer (see Section B.17). Because of the clear-cut, basic nature of the language structures needed to express the main elements of a KISS Object Interaction diagram in natural language, it is quite possible to automatically generate grammatically correct sentences on the basis of such a diagram, or alternatively of a repository of modelled objects and actions. Unlike the Grammalizer, the Verbalizer is not confronted with the problem of parsing complex and versatile grammatical structures.
NL paraphrasing is not exclusively used in diagram-oriented CM approaches. For example, the practice of paraphrasing predicate-like structures as NL sentences is widespread, and in the case of predicate logic is taught in introductory courses and is subject to certain informal conventions. Obviously, linear representations sharing fundamental properties with natural language (most typically predicate structures) will lend themselves better to easy paraphrasing than non-linear representations such as diagrams.
However, especially for informal NL paraphrasing, explicit predicate structures need not be obviously present in a diagram (or some other structure). To someone with an in-depth understanding of a diagram, such predicate structures can always be found through interpretation and then made explicit (for example as in NIAM's 'reading out' activity; see Section B.13).
Bertztiss (Bertztiss, 1997) suggests adding a communal component to paraphrasing (*ibid.*, p.49): "A statement could be paraphrased by several people in order to establish whether uniform interpretation is being given to the statement". This technique is related to the one discussed in Section B.16. However, Bertztiss provides no further clues as to how "uniform interpretation" can be recognised; presumably it is left

simply to acknowledgement by the parties involved that several paraphrasings seem to reflect sufficiently similar meanings. We have discussed similar considerations in Section B.21.

Finally, it is possible to paraphrase diagrams in other than directly NL-related languages. For a reference to non-NL-based paraphrasing as a technique, see (Dalianis and Johannesson, 1998).

The technique of mapping between linear and diagram representations, though mainly applied in Phase 4, is also regularly applied in relation to Phase 2, opening possibilities to bridge the gap between various types of text authoring and and diagrams. However, caution should be taken as to the ease with which this gap can be bridged. In the past, the 1:1 link between diagrams and 'real', un-normalised natural language has at times been overestimated. The technique is also popular for application in Phase 5.

B.24 Filling Templates

Template filling is a technique permitting less freedom than most forms of concept description but also providing better control and overview. In addition, it makes it easier to achieve completeness and conformity to the descriptive meta-framework of choice. Templates can take numerous forms, but essentially they consist of a number of slots which can be filled (either optionally or obligatorily) with entities (instances) of some prescribed type[2]. Good examples are the standard attributes or 'database fields' concerning concept definition (e.g. genus, differentia, synonyms, etc.; Strehlow 1997b), but also standard syntactic or semantic roles and relations as expressed in normalised phrases (as used, for example, in the Grammalizer tool; Hoppenbrouwers et al., 1997).

The main advantage of templating is, obviously, that the person filling in the slots can do so with a relatively clear idea in mind of 'what elements we should come up with, where we should put them, and (possibly) how we should describe them', and can easily spot gaps in the slot filling.

In addition, template filling imposes a relatively rigid working structure on the analyst; use of various types of templates opens up possibilities for procedural and workflow control. A particularly advanced type of template filling is supported by *context sensitive templates*: the templates and slots are made available to the analyst as they become relevant to a particular conceptualisation stage and situation. This normally requires automated tool support. Automated templating aids can also help structured *storage* of concept descriptions, and encourage a standardised format for displaying them.

Templating seems especially useful in cases where a clear structure exists which requires systematic filling in of instances without much variation in form; this implies that some reasonably well-described descriptive framework needs to be in place first. In such cases, the use of templates (even in an informal context) is highly recommendable. This boils down to some use in Phases 1 and 3, but in the sense as discussed mostly in Phase 4.

[2]The obvious metaphor here is that it involves using frozen language –a database-like setup– to create frozen language.

B.25 Adapting Existing Concept Representations

In between the possibilities of importing complete, existing concepts or conceptual frameworks (Section B.19) and creating concepts and frameworks from scratch (discussed in many sections throughout this appendix) there is the obvious but tricky option of adapting existing concept descriptions in order to tune them to the specific goals of some situation. This involves finding a balance between responsible re-use of the results of previous conceptualisation efforts (which may have been painstaking and expensive and thus represent a certain value) and the considerable trouble it may take to change the concept descriptions so as to make them serve their intended purpose. For a general discussion of this approach in context of terminology and thesaurus management, see (Roulin, 2001, p.399).
Note that what we mean here with 'adaptation' is not the use of existing concept descriptions as input for Phase 2 or 3 conceptualisation, but full-scale use of representations as a basis for Phase 4 conceptualisation. This implies that the concepts or framework in question must have been selected (possibly procured) as the result of some Phase 3 activity. Thus, adaption can perhaps more readily be seen as an extension of procurement than as a basic form of conceptualisation 'from scratch'. It clearly involves the carrying out of creative changes, related to productive rather than selective techniques, but nevertheless it seems likely that up to a point, adaption (or *customisation*) of an existing conceptual framework may be incorporated in a line of action (perhaps even a *service*) that also includes the selection and even provision of the existing framework. Yet in other cases, adaptation will stand clearly apart from the initial creation of the framework adapted; often, one will even be simply 'stuck with' some framework and have to somehow cope with it (for example, in legacy situations).
A number of (partly overlapping) adaptation sub-activities can be distinguished, predominantly related to meaning description:

- Mutation of concept labels (form)
- Refinement (meaning)
  - creating sub-types
  - creating new attributes
  - creating new relations
- Simplification (meaning)
- Mutation of relations with other concepts (meaning)
- Mutation of concept attributes (meaning)
- Changing extension specification (meaning)
- Changing terms of descriptive meta-concepts (form/meaning representation)

The list above is not comprehensive, it merely serves to indicate the sort of 'subtechniques' relevant to concept adaptation. Some related actions (change types as

observed in the evolution of conceptual modelling schemas) have been described by (Wedermeijer, 2002). The list can perhaps best be seen as a 'repertoire of changes', or a set of 'change patterns'. Further study of the actions concerned seems interesting and relevant (see Section 8.2).
A complication that may occur in relation to concept adaptation may be of a legal rather than a functional nature. Standards or other imported frameworks may be subject to copyright or other restrictions involving authority. Before adaptation is embarked upon, one should consider whether it is allowed in the first place. In fact, such considerations should be part of the procurement process.

B.26 Reaching Consensus in Groups and Communities

This is of course a topic that in general has been subject to countless studies in numerous fields, including sociology, politicology, law, and management. We can but make some practical observations and remarks with respect to reaching consensus in context of conceptualisation, in particular concept description.
Perhaps most importantly, as in most processes involving individual-based creative thinking, purely democratic principles seem hard to apply. If more than a handful of people are to be involved (say, $n > 5$, though even between two people a mechanism for reaching consensus may be required), three main approaches seem relevant:

- Employ observation/derivation of concept descriptions on the basis of Phase 1-2 material instead of actual Phase 4 democratic concept description
- Collect suggestions from the group (possibly using a brainstorm)
  - organise *all* suggestions made by the group, perhaps categorise them and generalise about them
  - only select particularly good suggestions
- Have someone (person or subgroup/committee) put together suggestions and have the group choose one (e.g. by means of voting or rated lists). This may or may not be accompanied by some form of discussion or debate. A minimal form of this approach is to have someone make a suggestion and merely provide the possibility of raising objections.

The whole range of ways to reach group consensus may be relevant, depending on the situation. Perhaps most importantly, the group should be aware of the difference between lexical and stipulative definition (Section 3.4.2), since this determines whether a conclusion may be drawn from collection of all individual opinions, or whether a constructive dialogue of some sort is required.
It seems possible, even promising, to deploy computerised tools and techniques for reaching agreement in conceptualisation. Such support may be particularly useful if conceptualisation is also more generally supported by computational applications. Inspiration can be drawn from techniques for decision making support and 'Electronic Meeting Rooms' (an exploratory study into this approach was conducted by

Vermunt, 1996). Related community-based techniques that are also very interesting in this respect are 'Conceptual Mapping' or 'Mind Mapping', and 'Dialog Mapping' (for the latter, see Conklin 2003). These approaches combine a very liberal, ad-hoc form of conceptual diagramming with electronic support of group interaction and decision making.
Clearly, techniques for reaching group consensus may apply in principle to all communal description and selection (i.e. decision making) activities: Phases 1.1-2, 2.1, 3.1, 4.1, 4.3; possibly also 5.1 and 5.2. However, concept selection and description involves matters *directly* concerning linguistic conventions, and therefore communal consensus acquires a particular urgency in phases 3.1 and 4.3.

B.27 Analysing Corpora

The analysis of linguistic corpora (collections –usually quite large– of existing texts, gathered with the purpose of performing linguistic analysis on them) is an exponent of somewhat different approach to explicit conceptualisation than most techniques for 'describing concepts'. In principle, it is an auxiliary technique for observation (see Sections B.10 and B.26). As such, it may be successfully used in relation to Phases 1 and 3. However, its primary use is as an aid in concept analysis description (Phase 4 and 5). We emphasise that we are concerned here with *analysing* corpora, not with gathering corpora.
A strong point in favour of the use of existing text as material for concept description is that it is a very good basis of finding representative examples of common word use in a particular context (Dubuc and Lauriston, 1997), even if the context is a highly domain-specific collection of large bodies of text (Ahmad and Rogers, 2001). It provides excellent support for *lexicographically oriented analysis* (Roulin, 2001). Not surprisingly, this technique typically applies to conceptualisation involving lexical definition, and may be a valuable aid in stipulative-lexical definition.
Perhaps the simplest, most illustrative 'sub-technique' for corpus analysis is known as the use of 'concordances'. A concordance is a listing of all occurrences of some concept (word or phrase), each in context of a specified number of words appearing in front of and after the concept in question. Stemming from classic lexicography, involving manual analysis of textual sources, it has become much more powerful in combination with (relatively simple) computerised text searching techniques. Concordances thus allow an analyst to view and compare a large number of real uses of a concept.
A more advanced but less straightforwardly useful set of techniques concerns statistical analysis of corpora, which is an area of ongoing research in fields like computational linguistics, information retrieval, and information extraction. To mention just a few of the possibilities such techniques may offer:

- Counting of word frequencies
- Counting of word co-occurrence frequencies
- Computation of frequency lists of words in view of some particular topic or domain

- Up to a certain point, rating of the *relevance* of some word for a certain topic or domain.

We cannot possibly begin to sum up the full range of techniques available in this field. A good overview of the basic onces can be found in (Pazienza, 1997).
(Sager, 2001) lists a number of aspects of terminology compilation that can be aided through corpus-linguistic techniques. We have added how these aspects fit our CPRM (italics).

- Determining what is a sufficiently large body for reliable term extraction (Statistical means can be used to decide when the addition of more text does not produce any new terms); *Phase 1-2*
- Determining the boundaries of subject fields and subfields (Statistical measures provide frequency patterns and so permit the delimitation of subject fields based on textual evidence); *Phase 3*
- Selecting cohesive and representative bodies of terms and phrases (The delimitation of terms and their distinction from collocations can now be fully justified); *Phase 3-4*
- Identifying terminological synonyms and describing the conditions of usage for terminological variants; *Phase 3-4*
- Supplying documentary evidence for the terminological information provided in an entry or record; *Phase 4, links to maintainable results of phase 1-2*
- Providing genuine contexts and examples of usage; *Phase 4, links to maintainable results of phase 1-2*

B.28 Checking by Using

The most common, 'natural' approach to the checking of concepts is simply triggered by doubts or questions concerning concepts or concept descriptions as encountered as a concept is used. It thus concerns an 'ex post' sort of action. It is usually related to operational language use, though agents involved may only become aware of some deficiency through 'conversation at language level' (Section 4.3.3). In fact, checking by using itself is typically part of such meta-linguistic conversation. However, it goes beyond the mere exchange of information about some concept: it involves the active investigation of what may be *wrong* with some concept or concept description, possibly to the point at which conceptualisation techniques are called in for support. Note that the latter presumes the *availability* of such tools, as wel as the *willingness and incentive* on behalf of the involved agents to engage in active checking.
Checking by using thus typically (but not necessarily) concerns *non-mediated meta-communication.* Results of checking by using may be used directly for the resolving of some language breakdown, or even be linked to some form of intervening meta-communication, i.e. result in a change of the medium system or documentation involved. Results of concept checks, or even just a note concerning the encountered problem, may also be reported to some authority, which then may or may not take further action.

B.29 Systematic Concept Checking

The systematic checking of concepts is an activity that entails both a clear sense of purpose (evaluating –certain aspects of– concepts, or concept descriptions: see Wright and Budin, 2001, p.870-1) and a prescribed method (in order to be systematic, some previously considered, rational line of action needs to be involved, quite possibly carried out iteratively). The approach by nature opens possibilities for automation, but given the technology currently available, in most cases this will necessarily be restricted to the potential deployment of various computational tools (Meyer et al., 1997). Still, systematic checking can of course also be carried out at the hand of protocols.

Systematic checking is detached from initial conceptualisation, but it nevertheless has an 'ex ante' feel to it: it is not so much meant to solve a particular concept-related problem that has emerged, but is aimed at more general improvement of concept quality in a DE, or even in an MSLC, and thus roughly aims at the future wellbeing of the DE as a whole, or at least of some medium system (and its users). As such it can be viewed as an aspect of *general quality management* (Wright, 2001c). However, a systematic line of action may of course be triggered by the occurrence of one or more individual concept-related problems (as may emerge from 'checking by using'). Systematic checking typically is related to *mediated meta-communication.*

When systematic checking is initiated (or whether it is 'installed' as a continuous process) is a matter of general concept management, and obviously depends on how 'expensive' it is in relation to the results it is expected to deliver. In a highly advanced, fully integrated and automatically supported conceptualisation environment, initial conceptualisation (Phases 1-4), checking by using, and systematic checking might be fully integrated.

Appendix C

The Gak Organisation

In this appendix, we provide a description of Gak at a rather general level, sketching the organisation, its main tasks, its culture, and its ICT structure. We then focus on the communicational effort of organising the large-scale implementation of social security law, providing an overview of the main artefacts (texts) that are used for (meta-)communication and are the product of conceptualisation at Gak. We also sketch the ICT global development approach at Gak.

C.1 GAK Organisation

In the words of the acting director of Gak Gelderland-Zuid, Gak aims to "offer an alternative income to people who are unable to acquire that income through regular means (i.e., labour); decide about the necessity of this, determine the amount, and execute actual payment; work with the clients to help find them new possibilities to acquire a regular income".

Gak Gelderland-Zuid employs 650 people, 300 at the Nijmegen office, 350 in Arnhem. The employees are roughly divided in three groups, the *professional* group (est. 100 people), the *management* group (35), and the *administrative* group (500). The professionals are specialists in the Gak domain, divided in *Verzekeringsartsen* ("medical insurance consultants"; 50 people) and *Arbeidsdeskundigen* ("labour consultants"; also 50 people). They are typical knowledge workers, who gather information about clients and give their professional judgement about the client's health, her ability to do certain jobs, and what may be needed for a client to re-enter the labour process (varying from physical treatment to job-oriented education and training).

The professionals serve to inform the key members of the administrative group, the *Claimbeoordelaars* ("claim assessors"), who determine whether a client should rightfully receive a benefit, what the amount of this benefit is, and according to which legal arrangement the benefit is assigned (which has implications for the period during which the benefit is allocated). A Claimbeoordelaar also performs the actual actions (using a collection of large information systems) that initiate (or terminate) the ongoing execution (payment) of the benefit. Once the benefit is allocated, a large group of lower ranking administrative staff take care of timely and correct payment of the benefits, which in some cases requires a fairly complex and labour-intensive process of information gathering, calculation, and administration. About 80% of processed benefits can be dealt with in a bulk process; 20% concern "complex cases". Yet even the bulk process "is terribly complex and requires a large number of tasks to be performed".

Communication with Gak clients involves written forms and notifications, and telephone conversations. There is a service desk, which is meant to answer most general questions. In special cases, clients can be put trough to Gak employees that are part of the assessment and administrative process.

In the Gak process at large, a strict separation is carried through between the WW department (*Werkloosheids Wet*; unemployment) and the AG department (*Arbeids-Geschiktheid*; "Labour Ability", a euphemistic term for "Disability")[1]. WW and AG are traditionally the two pillars of the Gak process. Each have their own administrative process and their own professional community; medical insurance consultants and labour consultants are involved only in the disability (AG) department.

In our case study, we focus mostly on the AG department, because its process and organisation (in particular the diversity of professional groups involved) seemed most interesting to our research. However, the choice between WW and AG was admittedly largely arbitrary; a choice simply had to be made to limit our work load and provide focus. Though considerable differences exist in the detailed processes and organisation at WW and AG, the main line is similar in both departments.

Importantly, the products Gak produces (for example, decisions about and payment of social benefits) are designed and specified centrally, in the Amsterdam headquarters. In this respect, Gak Nederland is very much a centralised organisation. Executive guidelines and product descriptions are managed alongside the portfolio of information systems used at Gak, and those information systems are mostly specified and designed centrally, which greatly strengthens the grip of the central management on operations. The authority within Gak that is responsible for the structuring and control of nationally uniform processes and infrastructure is the *Dienst Distribuerende Zaken* (DZ) or "department of distribution". This is a key department at Gak; its chief manager is also automatically a member of the Board of Directors and the Gak Head of Production.

The general opinion in both the central and regional offices is that centralisation is inevitable and a good thing, given the need to deliver a uniform and legal product (within the margins of interpretation the law allows for, which are not trivial). Another, more hidden drive behind centralisation is that LISV and Gak use an elaborate, standardised accounting system based on a set price for a large range (several hundreds) of "products" that Gak is payed for by LISV.

The *implementation* of the Gak processes are for a considerable part left to local management, who are the "process owners", and in a similar vein it is accepted that some local variation will occur in the interpretation of laws and rules. There are substantial differences between work floor practices of the various local offices. Even between two offices in the same district (for example, the Arnhem and Nijmegen offices in the Gelderland-Zuid district), the differences between implementation, organisation, and office culture are considerable. According to a senior manager, in many cases it would be problematic for an Arnhem employee to move to the Nijmegen office; this would require some re-training. So while centralised specification of products, guidelines, and the main information systems are embraced, local

[1] A third department, *Werkgeverscontacten* or "Employer relations", is completely discarded in our discussion for reasons of relevance and focus.

differences with respect to process implementation are considered a fact of life. Totally centralised, uniform implementation of the Gak processes is not believed to be achievable.
When we look at the supporting ICT infrastructure, doubts about the realisability of full standardisation are reflected in the fact that senior staff in various departments and at various levels (even at headquarters) condone the deployment of 'rogue information applications' that are built and deployed locally, outside the sphere of influence of official specification, design, and distribution authorities. Interestingly, the use of some such rogue applications has eventually spread across many regions, so that a sort of rogue application subculture exists alongside the official, centrally controlled systems. However, such rogue systems are mainly used for the gathering of management information and do not play a major role in the primary production process.
For an organisation that implements laws, law changes present a big challenge. At Gak this challenge appears to be dealt with fairly effectively, largely because of the centralised main infrastructure of Gak, but also because changes in social security laws are a familiar phenomenon in the Dutch political landscape and are explicitly reckoned with in Gak ICT management. Depending on the magnitude of reforms, substantial organisational efforts may be initiated, but these are considered to be simply a fact of Gak life. Nevertheless, part of the complexity of the Gak processes can be explained by its long history of constant adaptation, which has created a large if not uncommon legacy problem both in organisation, culture, and ICT infrastructure and applications.
Not unexpectedly, the Gak organisation is now extremely dependent on its ICT infrastructure. Its main systems can safely be dubbed "mission critical". Of the production processes, 80-90% is "digitised", despite the fact that paper files are still industriously kept of all clients, as paper is still seen as a more "trustworthy" carrier of valuable data. The Gak administration thus is a complex blend of electronically supported (semi-automated or automated) and paper based (manual) tasks, with inevitably a certain amount of redundancy in data storage (both paper and digitised data). Outside the main production process, data is increasingly and by now predominantly kept and processed digitally (for example, executive information, management information, instructions and manuals).

C.2 GAK ICT Culture and General Structure

As a consequence of Gak's centralised structure, but also for technical reasons and because of legacy, the large main computer systems of Gak are centrally controlled, and transaction based (VMS). They are typical of many slightly outdated but safe and functional computer systems that run in large data-intensive environments centred around huge databases (e.g. banks, insurance companies).
As we have seen, the centralised main systems are complemented by a fair number of locally controlled systems, many of which are custom built (though based on standard office software packages) following local initiatives. Some such locally developed systems or applications have later spread to other locations. ICT culture at Gak allows for limited local "interpretations" while striving for "stability and

uniformity". In our case study, we will not deal with locally originated systems, since we focus on parts of the ICT infrastructure that support the primary production process. These systems are all developed centrally.

Technical implementation (or perhaps more correctly, *realisation*) of the main computer systems at Gak is outsourced to a company called ASZ (*Automatisering Sociale Zekerheid*; "Automation of Social Security"). It is now a branch of a large ICT services corporation, but it used to be the internal Gak ICT provider and it is culturally and operationally still very much a part of Gak. Not unexpectedly, ASZ's technological knowledge of the specific Gak systems is unprecedented. The main systems of Gak have been mostly coded in COBOL, which continues to be the main programming language at Gak/ASZ.

There are three Gak computing centers, where the main servers and databases are located. Obviously, local offices are connected to the computing centers through a network. Apart from keeping the systems up and running, local ICT departments provide service and support through helpdesks. These provide some (rather minimal) information to the management about the regular functionality of the systems, but clearly their main role is to assist Gak staff when technical problems occur. Local ICT units are hardly involved in system design and implementation except for the local initiatives that stand apart from the main Gak systems.

It would be unfair to suggest that because of its centralised character, the main Gak procedures, guidelines, and ICT infrastructure are determined by an authoritarian central management that ignores the needs and requests regional offices or other stakeholders. Apart from intensive stakeholder participation in the development process, it is possible at least in principle for local offices to place requests for modifications of the system. Specification and design is carefully orchestrated and involves advisors and testers from all layers of the organisation; modifications are coordinated through *Release Management Teams* (RMTs). Pilots are run, evaluations performed. In addition, best practice records are kept, disseminated, and taken into account in management and policy making.

However, the number of regional offices and other stakeholder groups, and their operational and cultural variety, understandably prevents system development from being truly democratic. Hence the straightforward but tell-tale conclusion of a local ICT manager: "you cannot possibly hope to please them all". In addition, there is of course a score of practical problems to be considered, not the least of which is the purely technical burden of keeping a large legacy system up and running in a storm of changing requirements. The ICT departments, including ASZ, constantly feel the pressure of their huge workload and it is felt that many of the "lesser problems" that are indicated by operational departments "rarely reach the top of the priority list and consequently are seldom dealt with". In practice, therefore, law changes are the main trigger of cycles of the *Gak System and Process Development* process.

C.3 From Law to Operational Process

Since we are concerned not only with system design and implementation, but with a larger picture that concerns meta-communication and explicit conceptualisation, we are interested in principle in the complete chain of events (i.e. process and rule specification events) leading from a law to operational processes at the local Gak work floor. Once we have an overview and basic understanding of the whole specification trajectory (in particular the main steps and deliverables in it), we can focus on the more ICT oriented specification processes at Gak, which thus can be viewed in their larger conceptual context. Finally, we can turn to the underlying and largely implicit layer of *language specification* processes, which occurs across process specification and ICT specification. So as an illustration, we will first sketch all major steps needed to carry through a law change, down to the Gak production process. At that point, we turn to more specifically ICT related issues.

Gak implements Dutch social laws, so the texts at the beginning of the chain are the actual law texts, determining the rights (and limitation thereof) of Dutch citizens to receive some employment-related social benefit if the situation calls for it. After parliament has passed a social law or an amendment, the Dutch government delegates implementation of (most) social laws to the CTSV (*College van Toezicht Sociale Verzekeringen*; the "Supervising College for Social Insurances"), who mainly have a controlling function, and who pass responsibilities on to the LISV[2] for the coordination of implementation on a more applied level. The LISV sets central and authoritative guidelines for social insurance implementation and products, including detailed norms for rightfulness and timeliness, and determines how much the executive organisations will be paid by the government for delivering particular products. LISV product guidelines are still very much legally oriented, but they are much more specific and detailed concerning amounts, time spans, etc. than actual law texts. In the best Dutch *Poldermodel* tradition, for determining the guidelines the LISV consults the UVIs: *Uitvoerende Instellingen* or "Executive Organisations", of which Gak is one. The LISV then passes on the guidelines and arrangements to the respective UVIs, who are contractors for further implementation.

The leading document for Gak implementation of social laws is the *Gak handboek wet- en regelgeving* ("Gak handbook of laws and rules"), also referred to as "the Gak Bible". On the basis of these authoritative rules, further implementation is fleshed out. Central Gak specification of products, and related support through ICT infrastructure, are the responsibility of two "distribution stafs" located in Amsterdam, one for each main branch or "product cluster" (WW and AG). Together they constitute 'Staf DZ'. Implementation of guidelines and infrastructure as it takes place at a Staf[3] is laid in the hands of groups of specialists for each law, and one for each professional group (medical consultants and labour experts). So guidelines are leading, other specifications follow.

The specification process eventually results in four main categories of deliverable (apart from many secondary ones). First, there is the *produktklapper* (liberally translated "product registration specification"), that specifies how a particular social security product needs to be registered, and what a product should look like to be rightfully identified as such. This document is directly related to the contract between LISV and Gak. If a product seriously violates specs, this may lead to serious

[2]Once more, note that a new structure for the implementation of various social laws came into effect on 1-1-2002, and that in this thesis we report strictly about the situation before this change took effect.

[3]In Dutch a "staf" is a group or department supporting the management; not to be confused with English "staff", meaning 'employees'

legal claims. The product specification refers to law texts.

On a more process-oriented level, what comes out are *handboeken* (handbooks) for the professionals, and *leidraden* (guidelines) and *process beschrijvingen* (process descriptions) for use in the main administrative processes. *Handbooks*, *guidelines*, and *process descriptions* have an official, central status, and are the main texts on which further implementation of the Gak production process is based. Where *handbooks* are intended to provide input for expert knowledge workers who are to a considerable degree responsible for the design of their own work processes (the 'professionals'), the *guidelines* are legally oriented and primarily (but not exclusively) input for *Claimbeoordelaars* and quality control. Finally, the *process descriptions* concern "how Gak employees should account for their actions through the administrative system". It is these descriptions that are closest to action-oriented functional descriptions of the supporting computer systems; one could say that the process descriptions are interwoven with instructions on system use. Guidelines and process descriptions refer to each other, but references from process descriptions to guidelines are more common than those from guidelines to process descriptions. Process descriptions are more operational in focus, and can be considered the 'bottom line' in centralised Gak process documentation.

The next step in our descent from law text to execution thereof is from central to local level. AG at Gak GZ receive their instructions and executive orders mainly from DZ staff AG. This information is received by the local AG-BB (*bedrijfsbureau*: operations bureau), "a unit that supports the operational management, guarding processes and process quality. A process manager implements the laws and rules through the operations bureaus." At Gak GZ there are, not unexpectedly, two BBs: WW and AG; each is divided up into 'stafs'. For AG these are a medical consultant staf and a labour consultant staf (covering the two professional groups), and a claim assessment staf (focusing on support of the Claimbeoordelaars, and also the main administrative process). The stafs primarily support the MUAG (Manager Uitvoering Arbeidsgeschiktheidswetgeving): the senior executive in charge of the local AG production process. At this level, local management processes are structured and specified.

One level down, we finally approach the work floor: a number of production departments, each led by a production manager who reports to the MUAG. Importantly, the production departments employ a number of "MBBs" (*Medewerkers Betaal en Beheer*: 'Benefit Payment and Management workers'), who constitute the main administrative contingent. MBBs work alongside Claimbeoordelaars. Though at this level no official management agreements are (to our knowledge) documented, we should not forget that even at the production floor there exists a network of practices, agreements, and conventions, including conceptual conventions.

We have so far mentioned some directive key documents (texts) produced and passed on in the hierarchical decent from law to production process, but little or nothing has been said about the actual processes leading to the creation of such texts. Indeed, we cannot go into the all writing processes here for reasons of capacity and focus, even though we believe that this would in principle be relevant. Yet in our interviews and other investigations we have elicited explicit statements concerning the ad hoc nature of most authoring processes. In other words, though of course authoring is always a process, as far as we can tell the authoring processes in question were not strictly guided or structured, nor explicitly described; they were deliverable-oriented and involved no clear agreements[4]. Conclusively, directive Gak documents generally seem to be authored with a deliverable structure in mind, not a structured process through which these deliverables are created.

C.4 Process and System Development at GAK

Up to now, we have focussed on a series of directive documents concerning the main Gak production process, only marginally speaking about a particular set of deliverables that is produced at the centralised DZ Staf: the information system specifications that are created alongside the administrative products and processes they are meant to support, as well as related specifications, manuals, etc. Yet at Gak (if not universally), process design and (functional) system design can hardly be considered apart, and this is reflected in the Gak way of working. Having explored its context, we can now discuss in some detail the range of activities that constitute *Gak process and system development*, which is after all central to our investigation.
Gak Nederland implements its process and system development according to the *Gak Werkmodel voor Proces- en Systeemontwikkeling* ("Gak Work Model for Process and System Development"). The Work Model (WM) was introduced because it was felt that better development process control was required, in particular better prioritising and planning of development projects and activities. The version we have been allowed to study dates from September 1998 (version 2.0); apparently this version was still used in May 2001. The WM is part of the Quality System for Information Infrastructure. In the introduction of the Work Model document, it is made clear that "a quality system is a 'living' thing. The WM committee [...] evaluates the models on a regular basis in view of [its] goals. When necessary, the quality management section of the IT department will carry through additions and improvements. Suggestions by users are welcome" (GAK, 1998). From our investigations it became clear that the WM is not followed to the letter, but by and large is still the main guideline for system development. Some standard deliverables prescribed in the WM process seem to have failed to ever materialise.
It should again be noted that the product stafs- DZ are mostly involved with the implementation of law changes, and are communicatively rather distant from the Gak work floor, and even more so from the local ICT departments: "Apart from running the occasional patch, we [the ICT department employees] are not much involved with the functional design and operation of our system. Our business is to make the systems run well and fast" (local ICT manager).
Broadly speaking, the product stafs prepare and specify changes in the applications, and then order ASZ to implement them. After testing, the new release of an application is sent to DZ for national distribution. Responsibility for particular phases in and deliverables of the development process lie with specific departments, but the involvement of other departments is also explicitly arranged.
The work model distinguishes six main phases in system development, that "run in parallel with the development of the work process that requires support" (*ibid.*,

[4]Though the authoring of laws lies firmly outside our focus, we would like to refer to (Voermans et al., 1997) for some relevant research on law authoring support.

pp.10 and 12). A boiled-down overview of phases and sub-phases in the system development process, the parties involved, and their respective responsibilities, is presented in Table C.1. In this table, "Line Org." refers to the local departments executing the main work process, "Product Staf" to the centralised product-specific departments, "IT" to the Gak ICT department, ASZ to the external ICT provider, and SVIANED to a network provider (which was not mentioned before and is irrelevant with our focus, but is included for integrity of the table). "I" stands for "Involved" (party is to be involved in the activity by the responsible party), "R" for "Responsible" (party is responsible for execution of activity). "D" stands for "Direction" (party plays the role of 'director': taking care of planning, preparation, and coordination, but not assuming end responsibility).

Phase	Line Org.	Product Dept.	IT	ASZ	SVIANED
1. Information analysis	I	I	R	I	
2. Functional Design					
2.1 Functional specification	I	I	I	R	
2.2 Acceptance of f.s.		I	R		
3. Realisation					
3.1 Technical design				R	
3.2 Construction				R	
3.3 Package selection				R	
4. System test				R	
5. Acceptation	I	R	D	I	
6. Implementation					
6.1 Instruction and education		R		I	
6.2 Conversion				R	
6.3 Technical facilitation				R	I
6.4 Supervising execution	I	R			

Table C.1: Phases, parties, and responsibilities in the Gak work model for system development

The main process deliverables relevant to us are the following (responsible parties are added in parentheses):

- system descriptions (ASZ)
- system user manuals, including 'helps screens' (ASZ)
- process models (Product Staf)
- work process descriptions (Product Staf)
- guidelines (legal expert at Product Staf)

Officially, at the center of the official guidelines for concept-related development at Gak the *Gak-Dictionary* is positioned: a data dictionary that should be kept up to date by the Gak IT-architecture department. Updates to the Gak-Dictionary are deliverables in most of the phases defined; the dictionary itself is defined not as one deliverable but as a living, authoritative conceptual database. However, inquiries at the IT-architecture department quickly made clear that the Gak-Dictionary has actually never come into being. Hence, no central repository of the concepts part of the Gak data structures is available. The only component of the Gak Dictionary that has reached some level of realisation is the “Gak System Guide”, a list of all Gak systems, only sporadically including data structure descriptions, the authoring of which is experienced as “problematic”.

Samenvatting

Dit proefschrift betreft onderzoek naar een aspect van ontwikkeling en ontwerp van gecomputeriseerde informatiesystemen, n.l. het taalaspect. Informatiesystemen zoals bedoeld kunnen bijvoorbeeld traditionele databases zijn, maar ook geavanceerde 'kennis-gebaseerde' systemen. Dergelijke systemen maken intensief gebruik van menselijke taal voor mens-mens communicatie: ze ondersteunen uitwisseling van informatie tussen mensen, gecodeerd in taal.

Wij richten ons specifiek op de ontwikkeling van taalmiddelen (m.n. woorden en frases) die in een informatiesysteem worden ingebouwd, en die dus d.m.v. het systeem aan de gebruiker worden aangeboden. Die taalmiddelen vertegenwoordigen een voornaam deel van de systeemfunctionaliteit. Men kan daarbij bijv. denken aan namen van databasevelden, beperkte woordkeus ter invulling van databasevelden ('menukeuzes'), allerlei soorten termen voor categorisatie, en standaardverzoeken en -meldingen. De taalmiddelen dienen aan te sluiten bij de behoeftes en mogelijkheden die voortvloeien uit de menselijke taalgebruikscontext: het systeem moet bij de gebruiker en het gebruik passen.

'Natuurlijk' taalgebruik kenmerkt zich o.a. door voortdurende aanpassing van de taal aan de gebruikscontext. Dat is noodzakelijk vanwege de zeer vele verschillende gebruikscontexten, maar ook vanwege de veranderende wereld. Als de taalbehoefte varieert of verandert dienen de ondersteunende taalmiddelen zich daar in principe naar te voegen. In dit aanpassingsproces is communicatie over taal oftewel 'talige meta-communicatie' van belang.

Het is echter zo dat taalmiddelen die worden aangeboden middels gecomputeriseerde informatiesystemen soms *niet* in overeenstemming zijn met de taalbehoeftes van gebruikers, of met de taalafspraken tussen hen. Het systeem biedt niet de juiste woorden aan, of woorden die niet of slecht bekend zijn. De gebruikers kunnen daardoor niet of niet goed tegen elkaar zeggen wat ze willen. En zelfs als een systeem qua taal volledig voldoet, kan dat een tijdelijke zaak zijn, aangezien de gebruiksomgeving kan veranderen en de taalbehoeftes dan meeveranderen –en de ingebouwde taal niet; *in iedere geval niet zonder gerichte meta-communicatie*. In de huidige ICT praktijk is dergelijke inflexibiliteit eerder regel dan uitzondering, en als de taal waarover de systeemgebruikers beschikken onveranderlijk gespecificeerd en ingebouwd is, oftewel *bevroren*, dan steken de bovenstaande problemen voelbaar de kop op.

Het geschetste probleem is vrij goed waarneembaar in systeemgebruiksomgevingen, en wordt al enige decennia (doch vooral in algemene vorm) erkend door gebruikers, systeemontwikkelaars, en de wetenschap. Het oplossen of voorkomen ervan had tot nu toe echter een lage prioriteit, omdat het meestal betrekkelijk onschuldige com-

municatieproblemen betrof, terwijl voortdurende aanpassing van en communicatie over taalspecificaties een grote organisatorische inspanning vergt. In het licht van de alsmaar toenemende digitalisering van werk en samenleving valt echter een dramatische toename te voorzien in urgentie van het probleem. Er zullen steeds zwaardere eisen worden gesteld aan taalgerelateerde functionaliteit en flexibiliteit. Enorme aantallen specialistische en algemene taaldomeinen –en veranderingen daarbinnen, en verschillen daartussen– zullen moeten worden begrepen en beschreven.
Ontwikkelingen zoals gedistribueerde, federatieve, en coöperatieve informatiesystemen, virtuele ondernemingen, inter-organisatorische werkstromen, agent-gebaseerde systemen, e-commerce en e-governance leiden onvermijdelijk tot het grootschalig 'bevriezen van taal'. Er zijn zelfs al verstrekkende internationale initiatieven die toewerken naar domeinspecifieke beschrijving en standaardisatie van taal- en communicatie in relatie tot het Internet (denk aan webservices, ebXML, of het *semantic web*). Kortom, wij voorzien dat de 'Bevroren Taal Kwestie' een forse hindernis gaat vormen voor vooruitgang in de informatie- en communicatietechnologie.
Als onveranderlijke taalspecificaties en taalstandaarden in veel gevallen niet voldoen als ze in informatiesystemen worden ingebouwd, dan is er behoefte aan adaptieve benaderingen van taalspecificatie. Helaas is het automatiseren van taalspecificatie vooralsnog een utopie. Handmatige taalspecificatie en herspecificatie zal daarom veel en voortdurend nodig zijn.
De Bevroren Taal Kwestie kan het beste langs een brede linie worden benaderd: door het beter begrijpen en inrichten van het geheel van processen die communicatie over taal bewerkstelligen (talige meta-communciatie processen). Het betreft hier zowel het afstemmen van bevroren taal op de taalbehoeftes (het creëren en veranderen van ingebouwde taal) als het communiceren van talige conventies (informatieuitwisseling over ingebouwde taal).
Veel algemene meta-communicatie processen kunnen impliciet blijven en aan de menselijke intuïtie worden overgelaten, maar voor ICT-gerelateerde processen ligt dat anders. Daarbinnen is expliciete beschrijving van taal veelal noodzakelijk (vorm en vaak ook betekenis). Dergelijke processen voor expliciete taalbeschrijving en -verandering (oftewel processen voor *expliciete conceptualisatie*) dienen daarbij actief geïntegreerd te worden in of afgestemd te worden op systeemontwikkeling en -gebruik.
Het voorgaande heeft geleid tot het opstellen van de volgende centrale onderzoeksvragen:

- Welk theoretisch kader stelt ons in staat processen voor het bevriezen van taal, en de omgevingen waarin dat gebeurt, te beschrijven en te analyseren? (theorieontwikkeling)
- Wat is een operationele vorm van dit theoretisch kader waarmee we concrete analyses en evaluatie van conceptualisatieprocessen kunnen uitvoeren, en deze daardoor kunnen verbeteren? (theoriegebaseerde methode)

Wij namen als uitgangspunt een combinatie van functionaliteitgerichte taal- en communicatietheorieën, die het mogelijk maakt taal en taalbeschrijving in de ICT te beschouwen binnen een meer algemene context van taalgebruik en communicatie (taal als een gereedschap). Tevens werd studie gemaakt van diverse aanpakken en technieken die gebruikt worden in expliciete conceptualisatie. Vervolgens ontwikkelden wij een theorie die ons in staat stelt taalgebruik en -ontwikkeling direct te relateren aan taalbeschrijvingsactiviteiten zoals die voorkomen in het gebruik en de ontwikkeling van informatiesystemen. Dit maakt het mogelijk om taalbeschrijvingsprocessen als onderdeel van systeemontwikkelprocessen te beschrijven, analyseren, en evalueren, in context van ICT-gerelateerde gebruiksomgevingen en de taalsituaties die daarmee samenhangen. Naast een basaal geformaliseerd 'conceptueel raamwerk' zijn door ons ontwikkelde basisonderdelen van de theorie een functionaliteitgerichte typologie van talen en taalgebruik, een typologie van talige metacommunicatieprocessen, en een referentiemodel voor conceptualisatieprocessen.

Op basis van de theorie werden een coherent analytisch raamwerk en een methode ontwikkeld. Voor het aansturen van de ontwikkeling van systemen of systeemportfolios, met inachtneming van vele en vaak tegenstrijdige belangen, wordt steeds vaker 'systeemarchitectuur' ingezet. Onze methode heeft de vorm gekregen van een z.g. *architecture viewpoint*: een analytisch en communicatief instrumentarium dat speciaal afgestemd is op een bepaald aspect van systeemarchitectuur. Wij hebben dus een gedetailleerde, theoretisch onderbouwde suggestie gedaan voor een *architecture viewpoint voor conceptualisatieprocessen*.

De onderbouwing van de theorie en de daarvan afgeleide methode berust op een viertal pijlers. Allereerst is de theorie ontwikkeld op basis van bestaande en al eerder gevalideerde theorieën. Verder schept de door ons ontwikkelde theorie een kader dat positionering en evaluatie mogelijk maakt van bestaande conceptualisatiemethoden en -technieken zoals beschreven in de literatuur. Belangrijker is dat de theorie met succes is doorvertaald naar een operationele analytische methode voor conceptualisatie- en metacommunicatieprocessen (inclusief diagrammen, evaluatiepunten, en een stappenplan). Tot slot (en qua validatie het meest belangrijk) is een '*proof of concept*' van de methode gerealiseerd middels een casus studie in een bestaande organisatie.

Summary

This thesis concerns an aspect of the development and design of computerised information systems: the language aspect. Information systems as meant here can for example be traditional database as well as advanced knowledge-based systems. Such systems make intensive use of natural language for human-to-human communication: they support the exchange of information between people, coded in language.

We focus specifically on the specification of 'language items' (mostly words and phrases) that are to be built into information systems, thus becoming 'means for languaging' made available to the user. Those means for languaging represent an important part of information system functionality. They can be, for example, names of database fields, limited term options for the contents of database fields ('menu options'), various kinds of terms for categorisation, and standard requests and notifications. The means for languaging should be aligned to the language requirements that are posed by the human context of language use: the system should fit the user and the use.

'Natural' language use typically involves continuous adaptation of language to its context of use. This is necessary because of the numerous different contexts of use that occur, but also because of the changing world. If the language requirements vary or change, the supporting means for languaging should in principle be altered to match them. In this adaptive process, communication about language or *linguistic meta-communication* is important.

However, means for languaging provided through computerised information systems are often *not* aligned very well to language needs of users, or with agreements about language that hold between them. The system may not offer the right words, or may offer words that are poorly known to the user. Because of this, users may not be able to 'say to each other what they want', or only do so inadequately. And even if the system is optimally functional in terms of language, this may be a temporary situation, since the context of use (and therefore its language needs) may change –while the built-in language does not; at least *not without deliberate meta-communication taking place*. In current ICT practice, such inflexibility is rule rather than exception, and if the language available to system users is statically specified and built in, or *frozen*, then the problems mentioned above will occur noticeably.

The indicated problem can indeed be observed in system use environments, and has been acknowledged by users, system developers, and scientists for several decades (though in a rather generic form). Solving or preventing it, however, has so far had a low priority, because it resulted in relatively minor communicative problems, whereas continuous communication about and adaptation of language specifications takes a considerable effort. Yet in view of the ever increasing digitisation of work and society,

a dramatic increase of the problem's urgency can be foreseen. Increasingly taxing requirements will be posed concerning language-related functionality and flexibility. Great numbers of specialist and generic language domains –and changes within them, and differences between them– will have to be understood and documented. Developments like distributed, federated, and cooperative information systems, virtual organisations, inter-organisational workflows, agent-based systems, e-commerce and e-government; they all inevitably lead to the large-scale 'freezing of language'. There even are some far reaching international initiatives that work towards domain specific description and standardisation of language and communication in relation to the Internet (e.g. webservices, ebXML, or the semantic web). In sum, we expect the 'Frozen Language Issue' to become a considerable barrier for progress in information and communication technology.

If static language specifications and standards are often inadequate if built into information systems, then adaptive approaches to language specification are required. Unfortunately, operational automation of language specification is still science-fiction. Language specification and re-specification 'by hand' will therefore be needed often and continuously.

The Frozen Language Issue can best be approached along a broad front: through beter understanding and construction of the combined processes that enable communication about language (i.e. processes for linguistic meta-communication). This concerns the alignment of frozen language to language needs (creating and changing built-in language) as well as the sharing of linguistic conventions (information exchange about built-in language).

Many generic meta-communication processes can remain implicit and can be left to human intuition, but for ICT-related processes the situation is different. Often they necessarily involve explicit description of language (form but frequently also meaning). Such processes for explicit description of language (or processes for *explicit conceptualisation*) need to be actively integrated in or aligned to system development and use.

The above has led us to the formulation of the following central research questions:

- Which theoretical framework enables us to describe and analyse processes and environments in which language is being frozen?
- What is an operational form of this theoretical framework, by which we might use it for concrete analysis and evaluation of processes for freezing language in an information systems context, in order to improve such processes?

As a starting point we took a combination of functionality-oriented language and communication theories, enabling us to view language and language description in ICT in the more general context of language use and communication (i.e. language as a tool). In addition, we studied a number of approaches and techniques used in explicit conceptualisation. On this basis, we developed a theory that enables us to directly relate language use and development to language description activities as occur in the use and development of information systems. This makes it possible to describe, analyse, and evaluate conceptualisation processes as part of system development processes, in context of ICT-related working environments and the language situations they involve. Apart from a basic formalised conceptual framework, the

main components of our theory are a functionality-oriented typology of language and language use, a typology of linguistic meta-communication processes, and a reference model for conceptualisation processes.

On the basis of the theory we developed a coherent analytical framework and a method. For the management and development of systems and system portfolios, including the weighing of many (and often opposing) concerns, 'systems architecture' has become a frequently deployed means. Our method has been shaped as a so-called *architecture viewpoint*: an analytical and communicative instrument that is dedicated to a particular aspect of systems architecture. We have provided a detailed, theory-based suggestion for an *architecture viewpoint for conceptualisation processes.*

Validation of our theory and the method derived thereof rests on four pillars. First, the theory has been developed on the basis of existing and previously validated theories. Also, our theory provides a framework that enables positioning and evaluation of existing methods and techniques for conceptualisation documented in the literature. More importantly, the theory has been successfully translated to an operational analytical method for conceptualisation and meta-communication processes (including diagrams, evaluation points, and some guidelines for execution). Finally (and for validation most importantly), we have provided a proof of concept of the method by means of a case study in a real organisation.

Bibliography

Agerfalk, P. J. (2003). Actability principles in theory and practice. In Weigand, H., Goldkuhl, G., and de Moor, A., editors, *Proceedings of the 8th International Working Conference on the Language Action Perspective on Communication Modelling (Tilburg, the Netherlands).* University of Tilburg.

Ahmad, K. (2001). The role of specialist terminology in artificial intelligence and knowledge acquisition. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Ahmad, K. and Rogers, M. (2001). Corpus linguistics and terminology extraction. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Aiello, M., Papazoglou, M., Yang, J., Carman, M., Pistore, M., Serafini, L., and Traverso, P. (2002). *A Request Language for Web-Services Based on Planning and Constraint Satisfaction*, volume 2444 of *Lecture Notes in Computer Science*. London: Springer Verlag.

Aitchison, J. (1994). *Words in the Mind, an Introduction to the Mental Lexicon.* Oxford: Blackwell, 2nd edition.

Andersen, P. (1990). *A Theory of Computer Semiotics.* Cambridge series on human-computer interaction. Cambridge: Cambridge University Press.

Austin, J. (1962). *How to Do things with Words.* Oxford: Clarendon Press.

Avison, D., Lau, F., Myers, M., and Nielsen, P. A. (1999). Action research. *Communications of the ACM*, 42(1):94–97.

Bach, K. and Hamish, R. (1979). *Linguistic Communication and Speech Acts.* Cambridge, MA: MIT Press.

Baecker, R. (1993). *Readings in Groupware and Computer-Supported Cooperative Work: Assisting Human-Human Collaboration.* San Mateo, CA: Morgan Kaufmann.

Bechtel, W. and Abrahamsen, A. (1991). *Connectionism and the Mind, an Introduction to Parallel Processing in Networks.* Oxford: Blackwell.

Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The semantic web. *Scientific American, May 2001.*

Bertztiss, A. (1997). Software methods for business re-engineering. *The Journal of the Operational Research Society*, 48(1):148.

Blair, D. and Maron, M. (1985). An evaluation of retrieval effectiveness for a full-text document retrieval system. *Communications of the ACM*, 28(3):289–299.

Booch, G. (1994). *Object-oriented Analysis and Design*. New York: Benjamin/Cummings, 2nd edition.

Booch, G., Jacobson, I., and Rumbaugh, J. (1998). *The Unified Modeling Language User Guide*. Addison-Wesley.

Brend, R. M. and Pike, K. L., editors (1977). *The Summer Institute of Linguistics: its Works and Contributions*. The Hague: Mouton.

Bruza, P. and van der Weide, T. (1992). Stratified hypermedia structures for information disclosure. *The Computer Journal*, 35(3):208–220.

Budin, G. and Wright, S. (2001). Terminology resources on the internet. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Burg, J. (1997). *Linguistic Instruments in Requirements Engineering*. PhD thesis, Free University of Amsterdam.

Chen, P. (1983). English sentence structure and entity-relationship diagrams. *Information Systems*, 29:127–149.

Chung, L., Nixon, B., Yu, E., and Mylopoulos, J. (2000). *Non-Functional Requirements in Software Engineering*. Boston: Kluwer Academic.

Clark, H. H. (1992). *Arenas of Language Use*. Chicago: University of Chicago Press.

Clark, H. H. (1996). *Using Language*. Cambridge: Cambridge University Press.

Clark, H. H. and Brennan, S. E. (1991). Grounding in communication. In Levine, J., Resnik, L., and Teasley, S., editors, *Perspectives on Socially Shared Cognition*, pages 127–149. Washington, DC: American Psychological Association.

Cleverdon, A. (1984). Optimizing convenient on-line access to bibliographic databases. *Library Service Use*, 4:37–47.

Conklin, J. (2003). Dialog mapping: reflections on an industrial atrength case study. In Kirschner, P. A., Buckingham-Shum, S. J., and Carr, C. S., editors, *Visualizing Argumentation: Software Tools for Collaborative and Educational Sense-Making*. London: Springer Verlag.

Corbolante, L. and Irmler, U. (2001). Software terminology and localization. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Cruse (2000). *Meaning in Language, an Introduction to Semantics and Pragmatics*. London: Oxford University Press.

Crystal, D. (1987). *The Cambridge Encyclopedia of Language.* Cambridge: Cambridge University Press.

Dalianis, H. and Johannesson, P. (1998). Explaining conceptual models: Using toulmin's argumentation model and RST. In Goldkuhl, G., Lind, M., and Seigerroth, U., editors, *Proceedings of the Third International Workshop on the Language Action Perspective in Communication Modelling (Stockholm, Sweden).* Jonkoping International Business School.

de Bessé, B. (1997). Terminological definitions. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 1. Amsterdam/Philadelphia: John Benjamins.

de Moor, A. (1999). *Empowering Communities.* PhD thesis, University of Tilburg.

de Saussure, F. (1916). *Cours de linguistique generale.* Paris: Payot.

de Troyer, O. (1993). *On Data Schema Transformations.* PhD thesis, University of Tilburg.

de Vree, B. (2001). Ontologie: het onopgeloste logigram. Master's thesis, University of Nijmegen.

Dieng, R. and Corby, O. (2000). *Knowledge Engineering and Knowledge Management: Methods, Models, and Tools*, volume 1937 of *Lecture Notes in Artificial Intelligence.* London: Springer Verlag.

Dik, S. (1989). *The Theory of Functional Grammar (Part I).* Dordrecht: Foris.

DSDM Consortium (2001). *Dynamic Systems Development Method, Version 3.*

Dubuc, R. and Lauriston, A. (1997). Terms and contexts. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 1. Amsterdam/Philadelphia: John Benjamins.

Duesterbeck, B. and Hesser, W. (2001). Terminology-based knowledge engineering in enterprises. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Eco, U. (1999). *Kant and the Platypus, Essays on Language and Cognition.* London: Secker & Warburg.

Ekering, C. F. (2000). *De Specificiteit van EDI.* PhD thesis, University of Tilburg.

Fafchamps, D. (1993). Ethnographic workflow analysis: Specifications for design. In Baecker, R., editor, *Readings in Groupware and Computer-Supported Cooperative Work: Assisting Human-Human Collaboration*, chapter 4, pages 218–21. San Mateo, CA: Morgan Kaufman.

Firlej, M. and Hellens, D. (1991). *Knowledge Elicitation: a Practical Handbook.* New York: Prentice Hall.

Frederiks, P. (1997). *Object-Oriented Modeling based on Information Grammars*. PhD thesis, University of Nijmegen.

Frederiks, P., Koster, C., and van der Weide, T. (1995). Object-oriented analysis using informal language. Technical report CSI-R9516, Computing Science Institute, University of Nijmegen.

GAK (1998). Werkmodel voor proces- en systeemontwikkeling. Internal Report.

Galinski, C. and Picht, H. (1997). Graphic and other semiotic forms of knowledge representation in terminology management. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 1. Amsterdam/Philadelphia: John Benjamins.

Gamst, F. (1980). *The Hoghead: an Industrial Ethnology of the Locomotive Engineer*. New York: Holt, Rinehart & Winston.

Gamut, L. (1991). *Logic, Language, and Meaning*, volume 1. Chicago: University of Chicago Press.

Greenberg, S. (1991). *Computer-supported Cooperative Work and Groupware*. Academic Press.

Groot, P. (2000). Computer assisted second language vocabulary acquisition. *Language Learning and Technology*, 4(1):60–81.

Grootjen, F. A. (2001). Indexing using a grammerless parser. In *Proceedings of the 2001 IEEE International Conference on Systems, Man and Cybernetics (Tucson, AZ)*.

Gruber, T. (1995). Toward principles for the design of ontologies used for knowledge sharing. *International Journal of Human and Computer Studies*, 43(5/6):907–928.

Guarino, N. (1998). Formal ontology in information systems. In *Proceedings of the 1998 International Conference on Formal Ontology in Information Systems (Trento, Italy)*, pages 3–15. IOS Press, Amsterdam.

Habermas, J. (1981). *Theorie des kommunikativen Handelns*, volume I. Frankfurt am Main: Suhrkamp Verlag.

Habermas, J. (1984). *The Theory of Communicative Action*. Cambridge: Polity Press.

Halliday, M. (1976). *System and Function in Language*. London: Oxford University Press.

Halpin, T. (1995). *Conceptual Schema and Relational Database Design*. New York: Prentice Hall, second edition.

Hammerstey, M. and Atkinson, P. (1983). *Ethnography: Principles and Practice*. London: Travistock Publications.

Harder, P. (1996). *Functional Semantics*. Berlin: Walter de Gruyter.

Hilliard, R. (2001). Viewpoint modeling. Submission to 1st ICSE Workshop on Describing Software Architecture with UML.

Hoppenbrouwers, J. (1997). *Conceptual Modelling and the Lexicon*. PhD thesis, University of Tilburg.

Hoppenbrouwers, J. (1998). Browsing information spaces. In Prinsen, J., editor, *International Summer School on the Digital Library 1998*. Tilburg: TICER.

Hoppenbrouwers, J., van der Vos, B., and Hoppenbrouwers, S. (1997). Nl structures and conceptual modelling: Grammalizing for KISS. *Data & Knowledge Engineering*, 23(1):79–92.

Hoppenbrouwers, S. (1994). Connecting competence and performance for second language acquisition purposes. Master's thesis, University of Wales, Bangor.

Hoppenbrouwers, S. (2003). Initial analysis of the archimate conceptualisation situation. Archimate project internal technical report, Enschede: Telematics Institute.

Hoppenbrouwers, S. and Proper, H. (1999). Knowledge discovery: de zoektocht naar verhulde en onthulde kennis. *Database Magazine*, (7):21–25.

Hoppenbrouwers, S. and Weigand, H. (2000). *Meta-communication in the Language Action Perspective*, volume 2000-6 of *Aachener Informatik-Berichte*. RWTH Aachen.

Hoppenbrouwers, S. and Weigand, H. (2003). Conceptualisation support for language development. Technical report NIII-R0314, May 2003, Nijmeegs Instituut voor Informatica en Informatiekunde.

ISO/IEC (1995). *Standard 11179:4, Information Technology–Specification and Standardization of data elements: Rules and Guidelines for the Formulation of Data Definitions*. Geneva: International Organization for Standardization.

ISO/IEC (1996). *Kwaliteit van Softwareprodukten*. The Hague: Ten Hagen Stam.

ISO/IEC (2001). *Standard 9126-1:2001 Software engineering – Product Quality – Part 1: Quality Model*. Geneva: International Organization for Standardization.

ISPL Consortium (1999). *Introduction to ISPL*. The Hague: Ten hagen Stam.

Jackendoff, R. (1983). *Semantics and Cognition*. Cambridge, MA: MIT Press.

Johnson-Laird, P. (1983). *Mental Models*. Cambridge: Cambridge University Press.

Jonkers, H., van Buuren, R., Arbab, F., de Boer, F., Bonsangue, M., Bosma, H., ter Doest, H., Groenewegen, L., Scholten, J. G., Hoppenbrouwers, S., Iacob, M., Janssen, W., Lankhorst, M., van Leeuwen, D., Proper, H., Stam, A., van der Torre, L., and Veldhuijzen van Zanten, G. (2003). Towards a language for coherent enterprise architecture descriptions. In Steen, M. and Bryant, B., editors, *Proceedings of 7th IEEE International Enterprise Distributed Object Computing Conference (Brisbane, Australia)*.

Kaldewaij, J. (1986). *Geschiedenis van de Taalkunde*. Dordrecht: Foris.

Knuth, D. (1984). Literate programming. *The Computer Journal*, 27:97–111.

Kotok, A. (2002). New XML e-business model seeks to break semantic inoperability barriers. www.webservices.org/index.php/article/articleview/735/1/24/ (Web Sevices Papers).

Kristen, G. (1994). *Object Orientation: the KISS Method – From Information Architecture to Information System*. Addison-Wesley.

Kyng, M. (1995). Making representations work. *Communications of the ACM*, 38(9):46–55.

Lakoff, G. (1985). *Women, Fire, and Dangerous Things*. Chicago: University of Chicago Press.

Lassing, N. (2002). *Architecture-Level Modifiability Analysis*. PhD thesis, Free University of Amsterdam.

Levelt, W. (1989). *Speaking: from Intention to Articulation*. Cambridge, MA: MIT Press.

Lipka, L. (1992). *An Outline of English Lexicology*. Tuebingen: Niemeyer Verlag.

Liu, K. (1993). *Semiotics Applied to Information Systems Development*. PhD thesis, Twente University of Technology.

Loucopoulos, P. and Karakostas, V. (1995). *System Requirements Engineering*. McGraw-Hill.

Maguire, M. (1997). *RESPECT User Requirements Framework Handbook*. RESPECT Consortium. Version 2.21.

March, J. and Simon, H. (1958). *Organizations*. New York: John Wiley.

Meyer, I., Eck, K., and Skuce, D. (1997). Systematic concept analysis within a knowledge-based approach to terminology. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 1. Amsterdam/Philadelphia: John Benjamins.

Miller, G. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 11(38):39–41.

Mills, C. (1940). Situated actions and vocabularies of motive. *Americal Sociological Review*, 5(6):439–52.

Muller, M. J. (1995). Ethnocritical questions for working with translations, interpretations and their stakeholders. *Communications of the ACM*, 38(9):64–65.

Nijssen, G. (1993). *Universele Informatiekunde*. Beutenaken: PNA Publishing.

Nkwenti-Azeh, B. (2001). User-specific terminological data retrieval. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Nonaka, I. and Takeuchi, H. (1995). *The Knowledge-creating Company*. London: Oxford University Press.

Norman, D. A. (1998). *The Invisible Computer*. Cambridge, MA: MIT Press.

Object Management Group (1997). Unified modelling language specification, version 1.1 (OMG ad/97-08-05). Technical report.

Papazoglou, M. and Hoppenbrouwers, J. (1999). Knowledge navigation in networked digital libraries. In *Proceedings of the 11th European Workshop on Knowledge Acquisition, Modeling, and Management (Dagstuhl Castle, Germany)*. Heidelberg: Springer Verlag.

Papazoglou, M. and Schlageter, G. (1998). *Cooperative Information Systems: Trends and Directions*. San Diego, CA: Academic Press.

Pazienza, M. T., editor (1997). *Information Extraction, a Multidisciplinary Approach to an Emerging Information Technology*. Berlin: Springer Verlag.

Peirce, C. (1931). *Collected Papers of Charles Sanders Peirce*. Cambridge, MA: Harvard University Press.

Pinker, S. (1994). *The Language Instinct*. London: Allen Lane/Penguin Press.

Polanyi, M. (1958). *Personal Knowledge - towards a Post-critical Philosophy*. London: Routledge & Kegan Paul.

Proper, H. (1994). *A Theory for Conceptual Modelling of Evolving Application Domains*. PhD thesis, University of Nijmegen.

Proper, H. (2003). Informatiekunde; exacte vaagheid. Inaugural speech, University of Nijmegen.

Pullum, G. (1991). *The Great Eskimo Vocabulary Hoax and Other Irrelevant Essays on the Study of Language*. Chicago: University of Chicago Press.

Pusey, M. (1987). *Jurgen Habermas*. Chichester: Ellis Horwood.

Putnam, H. (1975). The meaning of meaning. In *Mind, Language, and Reality*. Cambridge: Cambridge University Press.

Radford, A. (1988). *Transformational Grammar*. Cambridge: Cambridge University Press.

Robinson, M. (1991). Double level languages and co-operative working. *AI & Society*, (5):34–60.

Rolland, C. and Proix, C. (1992). A Natural Language Approach For Requirements Engineering. In Loucopoulos, P., editor, *Proceedings of the Fourth International Conference on Advanced Information Systems Engineering (Manchester, UK)*, volume 593 of *Lecture Notes in Computer Science*, pages 257–277. London: Springer Verlag.

Roulin, C. (2001). Computer-assisted theaurus management. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F., and Lorensen, W. (1991). *Object-oriented Modeling and Design*. New York: Prentice Hall International.

Ryle, G. (1949). *The Concept of Mind*. New York: Barnes and Noble.

Sager, J. (1997). Term formation. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 1. Amsterdam/Philadelphia: John Benjamins.

Sager, J. (2001). Terminology compilation: Consequences and aspects of automation. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Salton, G. and Buckley, C. (1990). Improving retrieval performance by relevance feedback. *Journal of the Americal Society for Information Science*, 41(4):288–297.

Sarbo, J., Hoppenbrouwers, S., and Farkas, J. (2002). Thought as a logical picture of signs. *International Journal of Anticipatory Systems*, 12:137–152.

Schatz, B., Johnson, E., and Cochrane, P. (1996). Interactive term suggestion for users of digital libraries: Using subject thesauri and co-occurrence lists for information retrieval. In *First ACM International Conference on Digital Libraries (Bethesda, MD)*, pages 126–133. New York, NY: ACM Press.

Schmitz, K. (2001). Criteria for eveluating terminology database management programs. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Schoop, M. (1998). A language-action perspective on cooperative document systems: Habermas and searle in hospital. In Goldkuhl, G., Lind, M., and Seigerroth, U., editors, *Proceedings of the Third International Workshop on the Language Action Perspective in Communication Modelling (Stockholm, Sweden)*. Jonkoping International Business School.

Schoop, M. (1999). An empirical study of multidisciplinary communication in healthcare using a language-action perspective. In Goldkuhl, G., Lind, M., Seigerroth, U., and Agerfalk, P. J., editors, *Proceedings of the Fourth International Workshop on the Language Action Perspective in Communication Modelling (Copenhagen, Denmark)*. Jonkoping International Business School.

Schreiber, A. et al. (1999). *Knowledge Engineering and Management: the CommonKADS Methodology*. Cambridge, MA: MIT Press.

Searle, J. (1969). *Speech Acts: an Essay in the Philosophy of Language*. Cambridge: Cambridge University Press.

Searle, J. (1979). *Meaning and Expression*. Cambridge: Cambridge University Press.

Searle, J. R. (1983). *Intensionality*. Cambridge: Cambridge University Press.

Sebesta, R. (1999). *Concepts of Programming languages*. Addison-Wesley, 4th edition.

Sewell, W. (1989). *Weaving a Program: Literate Programming in WEB*. New York: Van Nostrand Reinhold.

Shannon, C. and Weaver, W. (1949). *The Mathematical Theory of Communication*. Cambridge, MA: MIT Press.

Sherlund, B., Wade, R. L., Emery, D., and Hilliard, R. (2000). IEEE recommended practice for architecture descriptions of software-intensive systems. Technical report, IEEE.

Sonneveld, H. and Loening, H. (1993). *Terminology: Applications in Interdisciplinary Communications*. Amsterdam: Benjamins.

Spradley, J. P. (1980). *Participant Observation*. New York: Holt, Rinehart, and Winston.

Stamper, R. (1996). Signs, information, norms, and systems. In Holmqvist, B. and Andersen, P., editors, *Signs at Work*. Berlin: De Gruyter.

Strehlow, R. (1997a). ISO 10241: Preparation and layout of terminology standards. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 1. Amsterdam/Philadelphia: John Benjamins.

Strehlow, R. (1997b). Frames and the display of definitions. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 1. Amsterdam/Philadelphia: John Benjamins.

Strehlow, R. (2001). The role of terminology in retrieving information. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Taylor, J. R. (1993). *Rethinking the theory of organizational communication: how to read an organisation*. New Jersey: Ablex Publishing.

Telem, M. (1988). Information requirements specification 1: brainstorming collective decision making approach. *Information Processing and Management*, 24(5):549–566.

Trudgill, P. (1983). *Sociolinguistics*. Harmondsworth: Penguin, 2nd edition.

Turner, W., Langerhorst, R., Eilers, H., Hice, G., Castermans, R., and Cashwell, L. (1985). *System Development Method*. Rijswijk: Pandata.

van de Riet, R. and Meersman, R., editors (1991). *Linguistic Instruments in Knowledge Engineering: Proceedings of the 1991 Workshop on LIKE (Tilburg, the Netherlands)*. Amsterdam: North Holland.

van den Heuvel, W. and Proper, H. (2002). De pragmatiek van architectuur. *Informatie*, (11):12–17.

van der Poel, K. (2000). Stap voor stap naar een definitie van informatie architectuur. White paper, Landelijk Architectuur Congress 2000 (Amsterdam, the Netherlands).

van Dijk, A., van Swede, V., and Visser, J. (1989). *Taalkundige Informatiesystemen Ontwikkeld met Grammars*. Rijswijk: Pandata.

van Reijswoud, V. E. (1996). *The Structure of Business Communication: Theory, Model and Application*. PhD thesis, Delft University of Technology.

van Reijswoud, V. E. and Dietz, J. G. (1999). *DEMO Modelling Handbook*, volume 1. Delft University of Technology, second edition edition.

Vasconcellos, M. (2001). Terminology and machine translation. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Veldhuijzen van Zanten, G., Hoppenbrouwers, S., and Proper, H. (2003). System development as a communicative process. In *Proceedings of the Seventh Multi-Conference on Systemics, Cybernetics and Informatics (Orlando, FL)*.

Verharen, E. (1997). *A Language-Action Perspective on the Design of Cooperative Information Agents*. PhD thesis, University of Tilburg.

Verhoef, T. (1993). *Effective Information Modelling Support*. PhD thesis, Delft University of Technology.

Vermunt, A. (1996). Definiëren tijdens een elektronische vergadering. Master's thesis, University of Tilburg.

Viskil, E. (1994). *Definieren, een bijdrage tot de theorievorming over het opstellen van definities*, volume 12 of *IFOTT Studies in Language and Language Use*. Amsterdam: IFOTT.

Voermans, W., Matthijssen, L., and Verharen, E. (1997). Legimatics in Europe: the LEDA-project. *New Review of Applied Expert Systems*, 3.

Wedermeijer, L. (2002). *Exploring Conceptual Schema Evolution*. PhD thesis, Delft University of Technology.

Weibel, S., Kunze, J., Lagoze, C., and Wolf, M. (1998). Dublin Core metadata for resource discovery. Technical report, the Internet Society.

Weigand, H. (1990). *Linguistically Motivated Principles of Knowledge Base Systems*. Dordrecht: Foris.

Weigand, H. and Hoppenbrouwers, S. (1998a). The dynamic lexicon from a functional perspective. In Olbertz, H., Hengeveld, K., and Sanchez Garcia, J., editors, *The Structure of the Lexicon in Functional Grammar*. Amsterdam/Philadelphia: John Benjamins.

Weigand, H. and Hoppenbrouwers, S. (1998b). Experiences with a multilingual ontology-based lexicon for news filtering. In *Proceedings of 9th International Conference on Database and Expert Systems Applications (Vienna, Austria)*. IEEE Computer Society Press.

Wetherbe, J. (1991). Executive information requirements: Getting it right. *MIS Quarterly*, pages 51–64.

Wettengel, T. and van de Weyer, A. (2001). Terminology in technical writing. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Winograd, T. (1988). A language action perspective on the design of cooperative work. In Greif, I., editor, *Computer Supported Cooperative Work: A Book of Readings*. San Mateo, CA: Morgan Kaufman.

Winograd, T. and Flores, F. (1986). *Understanding Computers and Cognition*. Norwood: Ablex Publishing.

Wood, J. and Silver, D. (1996). *Joint Application Development*. New York: John Wiley, 2nd edition.

Wright, S. (1997). Representation of concept systems. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 1. Amsterdam/Philadelphia: John Benjamins.

Wright, S. (2001a). Data categories for terminology management. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Wright, S. (2001b). Terminology and total quality management. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Wright, S. (2001c). Terminology as an organizational principle in computer integrated manufacturing environments. In Wright, S. and Budin, G., editors, *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Wright, S. and Budin, G. (1997). *Handbook of Terminology Management*, volume 1. Amsterdam/Philadelphia: John Benjamins.

Wright, S. and Budin, G. (2001). *Handbook of Terminology Management*, volume 2. Amsterdam/Philadelphia: John Benjamins.

Zachman, J. (1987). A framework for information systems architecture. *IBM Systems Journal*, 26(3).

Author Index

Curriculum Vitae

Stijn Hoppenbrouwers was born in Nijmegen on July 19, 1970. His secondary school education (Atheneum B) took place at Hertog Jan College in Valkenswaard. From 1988 to 1994 he studied English at the University of Utrecht, specialising in 'Syntax and Semantics'. He graduated Cum Laude, on "Exceptional Premodification in the English Noun Phrase". From 1993 to 1994, he also did a Masters in Linguistics at the University of Wales, Bangor, specialising in Second Language Acquisition. The Masters dissertation he wrote there was titled "Connecting Competence and Performance for Second Language Acquisition Research Purposes".
In 1996, after doing some odd jobs and working freelance as a Specialised Reader for the Oxford English Dictionary, he started as a researcher at the Applied Research Institute of the Economics faculty of the University of Tilburg (informatics department), working first on the Grammalizer project, then on the EU-sponsored TREVI project. In 1998, he began working as a Junior Research Consultant at ID Research B.V., Gouda, which included part-time work on his PhD project. In 2000, he switched to working as an *Informatiekunde* lecturer at the Nijmegen Institute for Informatics and Information Science (University of Nijmegen). Stijn is married and has two children.