

A FRAMEWORK OF INFORMATION SYSTEM CONCEPTS

The FRISCO Report (Web edition)

by

**Eckhard D. Falkenberg
Wolfgang Hesse
Paul Lindgreen
Björn E. Nilsson
J. L. Han Oei
Colette Rolland
Ronald K. Stamper
Frans J. M. Van Assche
Alexander A. Verrijn-Stuart
Klaus Voss**

© IFIP 1998

ISBN 3-901882-01-4

Full text of the FRISCO Report (MS-Word 6.0)

(as available by anonymous <ftp://ftp.leidenuniv.nl/pub/rul/fri-full.zip>)

The **FRISCO Report**, originally completed in December 1996, is now made available as a downloadable file for research and educational purposes. Not to be distributed for profit, in any form, without written permission of the
International Federation for Information Processing
(enquire: ifip@ifip.or.at)

Printed copies (soft and hard cover) are available with permission of the copyright holder, at cost + postage, from the
Department of Computer Science, University of Leiden, The Netherlands
(enquire: verrynstuart@rulcri.leidenuniv.nl)

© IFIP 1996, 1998

ISBN 3-901882-01-4

TABLE OF CONTENTS

PREAMBLE	1
1 INTRODUCTION	5
1.1 The Information System Area	5
1.2 Roots of the Problems in the Information System Area	7
1.3 Setting up a Framework of Information System Concepts:	10
1.4 The Structure of Our Conceptual Framework	12
1.5 The Structure of the FRISCO Report	13
2 A LINE OF REASONING ABOUT INFORMATION SYSTEMS	14
2.1 Searching for a Starting Point	14
2.3 Some Fundamental Questions to be Asked and Some Simple Answers	16
2.4 Towards a Raison d'être	24
3 INFORMATION SYSTEM CONCEPTS: An Integrated Overview	26
3.1 Our "Weltanschauung"	28
3.3 Actors, Actions and Actands	41
3.4 Constructing and Representing Conceptions	44
3.5 Communicating Conceptions: Restrictions Imposed by the Constructivist View	53
3.6 Models	55
3.7 Systems	60
3.8 Organisations, Information and Communication	64
3.9 Information Systems	72
3.10 Computerised Information Sub-systems	75
3.11 Summary of Assumptions and Definitions	83
4 INFORMATION SYSTEM CONCEPTS: A Formal Approach	92
4.1 The Fundamental Layer	94
4.2 The Layer of Actors, Actions, and Actands	100
4.4 The Layer of System Concepts	108
4.5 The Layer of Organisational and Information System Concepts	110
4.6 Summary of Primitives, Axioms, Definitions and Functions	112
5 A SAMPLE APPLICATION OF OUR CONCEPTUAL FRAMEWORK	121
5.1 Description of the Test Case	121
6 ELABORATION ON SELECTED TOPICS	137
6.1 Semiotic Concepts	137
6.2 System Concepts	146
6.3 Modelling Concepts	153
7 REFLECTIONS: Committed Positions on the Report	165
 FRISCO – “Mission * Impossible” ? (Eckhard D. Falkenberg)	165
 Is Information to Become the Phlogiston of the Late 20th Century? (Wolfgang Hesse)	171
 On Collaborative Linking Support for the FRISCO Concept System (Pentti Kerola)	177
 FRISCO - A Way to Insight? (Paul Lindgreen)	182
 Reflections from a Practitioner on the Information Concept (Björn E. Nilsson)	187
 A Dissenting Position (Ronald K. Stamper)	191
 FRISCO Reflected Upon: So What or Aha? (Alexander A. Verrijn-Stuart)	197
REFERENCES	203
GLOSSARY	210
INDEX	218

*Away with a corrupt world!
Let us breathe the air of the Enchanted island.*

George Meredith, The Ordeal of Richard Feverel

PREAMBLE

What is the FRISCO Report About?

The FRISCO report presents the results of the work of the IFIP WG 8.1 Task Group FRISCO¹. The group was proposed in 1987 [FL87] and established in 1988. It organised three conferences on this subject [FL89, FRE92, FHO95] and produced an interim report in May 1990 [Lin90a].

This (final) report is intended to provide a suitable conceptual framework, i.e., wherever possible, simple, clear and unambiguous definitions of, and a suitable terminology for the most fundamental concepts in the information system field, including the notions of information and communication, and of organisation and information system. The report approaches these issues at two levels, namely (1) a broad examination of relevant fields, and (2) a set of definitions and formal modelling principles. It is hoped that based on these definitions, it will be possible to achieve a clear understanding of the various kinds of information systems, such as information systems for companies, government departments, other organisations and communities of people, and of the various kinds of computerised information sub-systems, such as database systems, knowledge-based systems, decision support systems, expert systems, office information systems, management information systems and the Internet, to name but a few.

One thrust of the report is to provide an ordering and transformation framework allowing to relate the many different information system modelling approaches (i.e. sets of concepts for modelling information systems, meta-models) to each other.

Most importantly, the report places the social, information-oriented and technical aspects in appropriate conjunction. In other words, the conceptual framework is not intended to be confined to computerised information sub-systems, but is applicable to information systems in the broader sense.

Who should Read the FRISCO Report?

The report is written such that it can be understood by any person with a practical or theoretical background in organisations and/or information systems. It is particularly useful for everyone who is concerned in some way or another with the problems and issues of communication and information in organisations, enterprises and business environments, such as:

- System analysts and engineers;
- Information analysts and engineers;
- System and software developers;
- System development method engineers (system methodologists);
- Researchers and scientists in the areas of information and communication, system analysis and development;
- Teachers, textbook writers and PhD-students in these areas.

¹ FRISCO is an acronym for "Framework of Information System Concepts"

Why was the FRISCO Report Written?

The objectives of the FRISCO task group grew out of a concern - as early as 1987 - within IFIP WG 8.1 (Design and Evaluation of Information Systems) about the then scientific, educational and industrial situation. This was clearly expressed in the first manifesto [IFIP88]:

"There is a growing concern within IFIP WG 8.1 about the present situation, where too many fuzzy or ill-defined concepts are used in the information system area. Scientific as well as practice-related communication is severely distorted and hampered, due to this fuzziness and due to the frequent situation that different communication partners associate different meanings with one and the same term. There is no commonly accepted conceptual reference and terminology, to be applied for defining or explaining existing or new concepts for information systems".

The real concern (the misunderstanding of what is involved in organisational communication) is still there - in spite of our studies - and one may fear that some of the problems are innate to the various interested parties. The roots lie in the past, for the information system area has suffered from the fact that due to historic accidents, different facets of the area have been dealt with by different scientific disciplines or "cultures", in particular computer science and social sciences [e.g. Gog92]. Too little communication among these different "cultures" has taken place. While developing computerised information systems, too many computer scientists have neglected the essential organisational, cognitive and social aspects of information system development, and have hardly been aware of the central role of information and communication in organisations. This has often resulted in partial or even total failure of development projects of computerised information systems. On the other hand, social and organisational scientists have often been unaware of the importance of considering (the most essential) formal aspects of information system development and thus wandered aimlessly where more precision would have helped them out.

The FRISCO group itself underestimated these problems, in particular the existence of "hidden agendas" of the interested parties. Thus, the study took much time, but was certainly rewarding in that it helped clarify many of the conceptual foundations of the information system area, providing suggestive definitions of the relevant concepts and the basis for a suitable terminology.

What are the Major Global Achievements of the FRISCO Work?

The FRISCO report justifies the information system area scientifically by placing it in a more general context, comprising philosophy, ontology, semiotics, system science, organisation science, as well as computer science. Thereby, the concepts of the information system area become "rooted" or "anchored", that is, related to concepts of these other areas.

The FRISCO report provides a reference background for scientists and professionals in the information system area comprising a consistent and fully coherent system of concepts and a suitable terminology that enables them to express themselves about matters in the information system area in a structured and well-defined way. This framework of concepts can serve as a theoretical basis, as well as for further scientific work such as for the production of textbooks on various levels in the information system area.

At the same time, we must admit that what has been borrowed from different disciplines should not be seen as constituting new - and certainly not a "complete" - re-formulations. In order to adapt foreign theories to our area, we have been forced to make simplifications, which

to professionals from those fields may seem radical. What has been borrowed fits properly into our own domain but it is not claimed that its use covers all of the other discipline. In this way, the FRISCO report does provide a bridge between the various disciplines involved, in particular between computer science and social sciences. It covers both the most essential computer- and technology-related issues and the relevant organisational, cognitive and social aspects.

Why should *You* Read the FRISCO Report?

Since the FRISCO report provides a suitable and widely applicable framework of fundamental concepts for the information system field, you will be able to cut through the technology-contaminated "terminology jungle", in particular by looking through whatever happen to be the current buzzwords.

Regardless of whether you are a practitioner, teacher or researcher in the information system field, reading the FRISCO report can increase and broaden your understanding of many of the relevant issues and facets of this field, in particular of those where you are not already an expert in. It may help you to get rid of your "blinkers", whether you are aware of them or not.

If you are a methodologist or a system developer, the FRISCO report can help you to deliver better development methods or better information systems, respectively.

If you are a teacher or textbook writer, you may find the material in the FRISCO report suitable as a basis for developing lecture notes or textbooks about the information system field or subsaspects thereof.

If you are a practitioner, teacher or researcher in the area of information system modelling, you may find that reading the FRISCO report will increase your understanding of modelling approaches, and of how to relate them to each other.

Who were Involved in Writing the FRISCO Report?

The FRISCO report was written collectively by the members of the FRISCO task group. Its members are:

Eckhard Falkenberg, The Netherlands
Wolfgang Hesse, Germany (since 1993)
Paul Lindgreen, Denmark
Björn Nilsson, Sweden
Han Oei, The Netherlands (since 1993)
Colette Rolland, France
Ronald Stamper, The Netherlands
Frans Van Assche, Belgium
Alexander Verrijn-Stuart, The Netherlands
Klaus Voss, Germany

Eckhard Falkenberg served as chairman and as editor of this report, Paul Lindgreen as secretary and as editor of the interim report [Lin90a].

Acknowledgements

Our group has and had numerous contacts with people and groups working in our area. These contacts helped us to advance our thinking and our approach and to continuously refine our work. We like to thank all of them for their help and interest.

In particular, we like to thank the former members of our group, François Bodart, Belgium, and Sjr Nijssen, The Netherlands, for their valuable contributions to our work.

Furthermore, we have been supported by a number of persons associated with our group, in the form of comments, critique and reflections. These associates are: François Bodart, Belgium, Terry Halpin, Australia, Pentti Kerola, Finland, Kalle Lyytinen, Finland, Doede Nauta, The Netherlands, James Odell, U.S.A., and Yair Wand, Canada. We like to thank them for all their support over the years. We also like to thank Sylviane Schwer, France, for her support concerning our formalisation.



✓ have found that most scientists and philosophers are willing to discuss a new assertion, if it is formulated in the customary conceptual framework; but it seems very difficult to most of them even to consider and discuss new concepts.

Rudolf Carnap, Intellectual Autobiography

1 INTRODUCTION

1.1 The Information System Area

When talking about the information system area, different people may have different views about how broad or narrow this area is, and which other scientific disciplines are related to our area, and in which way.

Even the term ‘information system’ itself is interpreted quite differently by different groups of people. It seems to be interpreted in at least three different ways:

- As a technical system, implemented with computer and telecommunications technology
- As a social system, such as an organisation¹ in connection with its information needs.
- As a conceptual system (i.e. an abstraction of either of the above).

Corresponding to the different interpretations of information system there exist various different dialects of our professional language. Even people who appear to interpret information system in the same way use apparently different sets of concepts to explain it, and they apply a different terminology. The term ‘information’, for instance may stand for ‘data’, the actual and/or intended interpretation of a ‘message’, the ‘knowledge’ gained when receiving the data expressing a message, and so on. What is really meant by conceptual model or universe of discourse, and what are (if any) the differences between concepts such as entity and object, or transition and event, or activity, action, act and process? We need to answer such questions and introduce some terminological clarity.

It is difficult to define a single unified vocabulary for the whole domain of information systems. We have to accept that there are many different subdomains, concerned with different sets of problems, where we find the common, everyday words (such as information, data, communication, process, knowledge, system, etc.) pressed into service with restricted local meanings. FRISCO hopes to provide some insight into this diversity. However, we do not attempt to establish standards, but only clear the ground through a consistent terminological framework and recommendations for its use.

Ideally, the terminology for any selected domain will be placed on a sound conceptual foundation. Such a foundation is not built easily. We are convinced that it can only emerge as the result of a consensus gradually reached by the professional community working in that domain. In a few domains, where there appears to be a sufficient degree of consensus, we have attempted to isolate and clarify sets of fundamental concepts. Our choice of terminology is not a proposed standard but we hope it will recommend itself. More importantly, the concepts we have isolated must stand the test of critical discussion by the community before any terminology is standardised.

¹ The term ‘organisation’ is used here and throughout the report in the most general sense. Not only large companies are meant. One-man companies, profit- and non-profit-oriented organisations, clusters of companies interacting in some way, universities and research institutes are meant as well. Even the community of all Internet users and similar communities may be considered organisations.

Individuals and groups invest heavily in constructing their world views. Tampering with their favourite vocabulary can be disturbing and provoke some quite strong reactions. The first principle to adopt is perhaps to recognise these sensitivities by acknowledging that apparently superficial differences of terminology and concepts should not be suppressed since they may indicate important conceptual distinctions.

This process of sorting out and clarifying information system concepts requires that all participants preserve an attitude of tolerant detachment whilst still being prepared to defend their own positions.

We are privileged to be engaged in working on a new scientific domain, related to a number of other domains. As with all earlier cases, our science is emerging from a field of practice where theories are developed to consolidate and explain our experience and also to allow us to act more rationally in predicting, deciding, designing and advising. As in all other cases, our science will have to be measured, ultimately, in terms of our ability to improve problem-solving in the practical field.

Creating a theoretical foundation of information system science raises difficulties that are not encountered in the natural sciences. The reason is that, in our case, the theory, i.e. the knowledge about the domain, concerns the world of information, communication and knowledge itself. This has implications both for the way we are working and how we understand that work. It is not merely that we, as observers, are part of the phenomenon we study - a problem for any empirical science - we ourselves constitute rather complex information systems. In other words, the fundamental problems lie both at the level of study and at the "meta-level".

Philosophy, the mentor of all makers of science, has provided critical insight into just these concepts we are studying. However, the problems concerning the notions of information, knowledge and communication are by no means resolved from a philosophical point of view nor capable of being so resolved. Nevertheless, we have to make commitments on all these difficult, debatable issues for the purposes of information systems practice. We should try to use the insights provided by philosophy and other relevant disciplines as far as possible. Among these, we might consider ontology¹, semiotics², linguistics, cognitive science³, organisation science, sociology, and system science.

¹ The term 'ontology' is used in different meanings in the literature. Here and throughout the report we shall use this term as meaning that department of metaphysics concerned with "theories" or "views" of how human beings think of (i.e. "conceive") the "world". A particular theory or view will be called "ontological theory", "ontological view" or "ontological position". It is based on an unquestioned (but reasonably described) metaphysical position, or "Weltanschauung". Whereas an ontological view contains and correlates ontological categories or generic concepts representing specific components, aspects and features in a precise way, a "Weltanschauung" is an "inter-subjective" common understanding, expressed in informal terms.

² Semiotics is the discipline concerned with theories of signs.

³ By cognitive science we mean the collection of all those disciplines investigating cognition and cognitive aspects in some way.

1.2 Roots of the Problems in the Information System Area

Apart from the aforementioned
interdisciplinary nature of our area,

there are at least three major sources of problems:

- (a) *the large variety of interest groups,*
- (b) *conflicting philosophical positions,* and
- (c) *the lack of understanding communication.*

(a) Interest Groups

In the IT¹ field, we can distinguish two major and largely opposing camps; suppliers of IT products and services, versus IT users. They have quite different attitudes towards the language used in our field.

The suppliers, the smaller and somewhat more united community, tend to emphasise the technical view of information systems. The social view, which would better serve the interests of users, is more or less neglected by the suppliers. Many a fortune has been made by launching and or merely following a fashion, usually under the banner of some new pseudo-technical jargon. At different times, 'Management Information Systems', 'Decision Support Systems', 'Expert Systems', and so forth, have been the buzzwords under which marketing campaigns have been conducted. Language is used in marketing to create apparent, often illusory or downright fake, product differentiation. Jargon is used to sharpen the competitive edge. A surprisingly large number of terms in the information system area have been coined by suppliers of IT products and services, or by pseudo scientists. Thus we need to be wary of the vocabulary in our area.

Users, the larger group as yet without a common voice, have quite a different interest. For them, it is important to standardise concepts and the professional language used. Our work will probably receive a readier welcome from this side of the market. In general, any clarification of the concepts and the professional language will help management in the user organisations or departments, to specify their requirements more clearly.

Interest groups are power groups. Establishing standards, including the clarification of the concepts and the professional language used, is one way of exercising power. There are cases of institutional rivalry, where the name of a project may be a perfect description of its objectives, but where two potential funding agencies refer the request to the other, on the grounds that the terminology used clearly defines it as not belonging to its own area.

The many interest groups in the academic information system area are easily recognised. One major camp is almost exclusively concerned with formal considerations, and another one as exclusively concerned with informal considerations about the use, support, impact and future of information systems. Types of application serve as foci of interest groups as well: Office Information Systems, Public Administration Systems, Medical Expert Systems, Library Systems, Government Information Systems, etc. Furthermore, different interest groups can be identified as promoters or users of different techniques, formal languages or methods for analysis and design of information systems. Emphasis on the theoretical versus the practical aspects multiply these subdivisions. However, none of all these subdivisions is as significant as that between those who are mainly interested in the formal, technical and computer-science-

¹ Information Technology

oriented aspects, versus the social, organisation-science-oriented and inevitably mostly informal aspects, up to total rejection (or blissful ignorance) of the respective views of the other camp [e.g. Gog92].

The fragmentation of the community does not necessarily matter. Indeed, it can generate healthy, stimulating, creative tensions and friction, provided that the various interest groups overlap enough to permit them to understand the views of their intellectual neighbours. Unfortunately, there is a fairly deep gap between the formal, technical community and the informal, social one, being a major cause of misunderstanding. Being engaged in this study of fundamental information system concepts, and consisting of members of both camps, we, the FRISCO members, are well aware of this root problem, and we are consciously attempting to bridge this gap.

(b) Philosophical Positions

The different philosophical or metaphysical positions ("Weltanschauungen") are seldom made explicit, but they usually account for most of the acrimony generated in disputes about fundamental concepts. It is therefore necessary to consider them explicitly. Here are a few examples:

The technically oriented interest groups tend to hold *nominalist* or *mechanist* assumptions (meaning that to them, the world and all things in it behave according to deterministic principles, so that straightforward descriptions and explanations can always be given). In their field it is quite appropriate to limit concern to that of a world comprising machines and formal languages. The use of formal languages including logical calculi encourages a nominalist position to thrive, because it is so easy to give function and predicate names their meanings. Their study of information systems appears to be very fruitful, provided one does not want to step outside the domains of hardware, software engineering, formal linguistics, or logic. The simplifications of the nominalist or mechanist view make it relatively easy to use powerful formal methods.

The majority of practical system analysts and designers probably are *objectivists* (or *naïve realists*) who view the world as having a real existence outside themselves that merely needs to be mapped to a self-evident consistent set of concepts. They see around themselves an obvious, familiar reality, that can be expressed through "entities", "relationships", "objects" and the like. They are prepared to use formal methods and extend them by giving words and numbers their "obvious" meanings. For information systems of limited scale, not dealing with contentious subjects, this position may be quite appropriate.

When one considers how the descriptions and models mentioned might be interpreted in the social domain, one may easily expose the limitations of the above positions. As soon as one steps out of the engineering workshop into the world of conflicting cultures, the idea of a simple objective reality does not hold water [Sea95]. Two typical views realising that are those of the *subjectivist* (who acknowledges that all conception has a purely private and personal aspect) and the *constructivist* (who in addition to the view of the subjectivist takes the individual experiences of reality and constructs an "inter-subjective reality" through the sharing of those experiences in actions where agreement is reached about categories of things and the boundaries of individuals). The constructivist approach rejects the objectivist assumption that reality simply exists as ready-made individual things and replaces it with a reality that is built through the operations of the members of a community through the social consensus about subjective conceptions that serve their culture.

The tacit adherence to conflicting philosophical assumptions has made our early discussions rather difficult. Thus we found it to be highly important for our work to discuss the benefits or drawbacks of the various positions, and choose one position consciously and explicitly.

(c) Complexity of Communication

Facilitating and improving communication is certainly among the more important functions of an information system. Communication can, for example, be examined in a semiotic framework that can help us to diagnose another important root problem. We can, for example, distinguish the following semiotic layers regarding communication and related issues [Mor46, Che57, Nau72, Sta73]:

- **Physical layer:**
the physical appearance, the media and amount of contact available
- **Empirical layer:**
the entropy, variety and equivocation encountered
- **Syntactical layer:**
the language, the structure and the logic used
- **Semantical layer:**
the meaning and validity of what is expressed
- **Pragmatic layer:**
the intentions, responsibilities and consequences behind the expressed statements
- **Social layer:**
the interests, beliefs and commitments shared as a result

Again, these layers can be divided into two groups in order to reveal the technical versus the social aspect division. Physics, empirics and syntactics, taken together, constitute a domain where technical and formal methods are adequate. However, semantics plus pragmatics plus the social domain can hardly be explored if those methods are used exclusively and without modification. Thus, whilst we may attempt to confine our thinking about the computerised parts of information systems within a limited framework of formal concepts, we will find difficulties if we aim at finding a similarly neat and tidy solution for the semantical, pragmatic and social domain.

In fact, all these layers are relevant when discussing and setting up a framework of information system concepts. Sometimes, a term may be used at one or a few layers only. The terms 'bit' or 'byte' for example are mainly used at the empirical layer. More often however, a term is used at several or even all of these layers. A typical example of such a multilayer-related term is 'message'. One can talk about the physical appearance of a message, about its entropy, about its syntax and semantics, as well as about its pragmatics in a social context. At each layer, the notion of 'message' will inevitably bear a different meaning.

The problem is that people, when uttering those multilayer-related terms, frequently fail to mention the layer or layers they are focusing on, which may result in severe misunderstanding.

1.3 Setting up a Framework of Information System Concepts: Guiding Principles, Options and Choices

We found during our work that for the information system field, there is no a-priori, objectively determined conceptual foundation which one could take for granted and simply use for building upon. We found that this conceptual foundation can be determined only inter-subjectively, through consensus. Thus, when setting up our framework of information system concepts, we inevitably had to face the problem of identifying the various options concerning certain facets of our field, and to choose from these options according to certain guiding principles.

Here are the most important guiding principles we applied whenever a necessity of choice arose:

(a) Global consistency

This is in a way a self-evident principle. We assume that nobody will be interested in a framework of concepts consisting of various independent and incompatible portions. Thus, the aim is one coherent framework where every concept is related to every other one in a specific, well-established way, such that a consistent whole results.

(b) Generality

We realise that there are many specialised subfields in our field. We do not attempt to consider those specialities in any detail. That is, our framework is meant to be as generic as possible, with respect to information systems. However, our framework must of course be specialisable and extensible wherever necessary, to cater for the various specialised subfields.

(c) Simple is beautiful

The resulting framework should be as simple and straightforward as possible. If you, the reader, will find that our framework is indeed simple, maybe even too simple or trivial, we have achieved this goal. In other terms: Our field as a whole is broad and complex enough "by nature", we should not add any *unnecessary* complexity to it.

(d) Anchoring information system concepts in related fields

Within our framework, we try to use, as much as possible and wherever appropriate, well-established concepts from other relevant disciplines. In this way, an isolated conceptual framework, being fairly incompatible with conceptual frameworks of related disciplines, may be avoided. In order to achieve this goal, information system concepts will be "anchored" upon other relevant disciplines. These relevant related disciplines will be introduced and treated, but only to the absolutely necessary level of detail.

(e) A conceptual foundation to be built upon

For practical reasons, it is inevitable that our conceptual framework is limited in scope, that is, not all facets of the information system field can be treated. However, it must be possible that our framework serves as a conceptual foundation, as a kind of "crystallisation kernel", from which one can build further, by introducing specialisations (guiding principle b) or other extensions.

As already stated, we found numerous options to choose from. In the following, the most important ones are mentioned together with the choices we made.

We found that is hardly possible to treat a conceptual framework in reasonable depth without choosing some, and preferably well-justified philosophical position to start with. This is due to

the fact that the formulations of most of the significant concept definitions or explanations are to some extent dependent on the (consciously chosen or tacit) metaphysical assumptions. They tend to creep in everywhere. We have chosen as a basis for our framework on a constructivistic metaphysical position¹. We found it to be the most reasonable one in the context of information systems, because it guarantees the highest level of generality in this context (guiding principle b). However, we do not deny that other positions are possible in this context as well.

We found that as soon as we seriously want to discuss and introduce any framework of concepts, we have to assume *a priori* some basic, underlying ontological view. For example, suppose we want to say that concept A refers to a thing with such and such properties, or concept A is an entity associated with concepts B and C by specific relationships. This is only meaningful if we *a priori* assume that there are "things" with "properties", or "entities" associated into "relationships", respectively. Otherwise, the very nature of the concepts we are dealing with remains unclear and illusive. For the purposes of introducing our framework, we have chosen a simple conceptual basis (guiding principle c), which will be further explained in chapter 3. It is important to note that this choice does *not* imply that we recommend this simple conceptual basis for *modelling all* sorts of information systems. Additions to this conceptual basis may be advisable when modelling, for example, complex or sophisticated information systems.

It is well-known that there exist a large variety of such conceptual bases (frameworks of modelling concepts, modelling approaches, meta-models) for modelling information systems in general, and domains of discourse for information systems in particular. To the inexperienced observer, the modelling issue appears to be like *Pandora's box*. Although we claim that the existing diversity is too large, we do not deny that a certain degree of diversity in this area is appropriate to cater for the different, special requirements of special sorts of information systems, such as plain database systems, time-oriented or planning systems, document retrieval systems, rule-based systems, etc. Consequently, we do not advocate having one and only one conceptual basis for modelling all the various sorts of information systems. We favour forming a solid basis for an *open ordering and transformation scheme*, which allows us to relate the various modelling approaches to each other, as well as to place new, emerging modelling approaches into this scheme (guiding principle b).

Another area of concern has been the level of formality of our framework of concepts. In order to demonstrate that our framework is, as a whole, consistent in itself and with respect to the chosen conceptual basis (guiding principle a), we provide a formalisation, starting with undefined primitives and axioms, and defining further concepts upon these primitives. However, for the sake of a proper understanding of the whole framework, we have to explain these primitives and axioms informally - rather than "define" them - using "common sense". On these grounds, we have chosen to split the presentation of our framework into two major parts, one emphasising explanations and understanding, and one comprising a purely formal approach, which stresses the consistency of the framework.

Finally, a whole bunch of options has to do with the question which other disciplines or facets of other disciplines are really needed for properly understanding information systems and the concepts directly related to them, that is, for reasonably "anchoring" information system concepts. We do not claim that we have a complete answer to this question. However, we

¹ This choice will be explained in chapter 3.

tried to set up our framework in such a way that specialisations and extensions in various directions can be done easily (guiding principles b, d, e). Since this question is highly relevant for the essence and structure of our framework, it will be treated in some detail in the following section.

1.4 The Structure of Our Conceptual Framework

What is needed to understand information systems? In the following points, the most important conceptual dependencies and pathways of understanding are listed, beginning with information systems:

- (a) Information systems exist exclusively within organisations, to support their work, and to fulfil their information and communication requirements. To understand information systems, we therefore need to understand organisations, what they are, how they work, what their components are, and what their structure and behaviour is. Thus, we will borrow from *organisation science*. Organisations can be viewed as systems, organisational systems. In this view, information systems are specific sub-systems of organisational systems.
- (b) To understand the information and communication requirements within organisations, we have to understand the notions of information and communication. These issues are closely related to *cognitive science* and to *semiotics*, from which we consequently will borrow.
- (c) Information systems and organisational systems both are systems. To understand them, we have to understand systems in general. Thus we will borrow from *system science*.
- (d) Systems are specific conceptions (in the minds of people), and can be represented in some (modelling) languages. Thus, in order to understand systems, we have to investigate the issues of conceptions, models and languages. Again, these issues are closely related to *cognitive science* and to *semiotics* (b), from which we consequently will borrow again.
- (e) To be able to investigate any of these issues on firm grounds we have to rely on some basic *ontological view*, as well as on some suitable *philosophical position*.

During our work, we have been investigating the above pathways of understanding. As a result, our framework of concepts provides bridges between the field of information systems and organisation science, computer science, system science, cognitive science, semiotics, and certain aspects of philosophy.

To put things into perspective, we have deliberately avoided a bottom-up sequence as above (a-e), choosing instead the reverse (top-down) sequence, from generic to more specific considerations. This leads to a structure whereby the reader has to follow a somewhat lengthy road before reaching full understanding of the core concept, information systems.

1.5 The Structure of the FRISCO Report

Before going into more detailed framework problems, we will investigate the questions why we need such a framework in the first place, and what the objectives of such a framework should be. We consider these questions from various points of view, which - taken together - add up to a full line of reasoning about information systems (chapter 2, a condensed version of which is included in this summary).

The core of the report is formed by chapters 3 and 4. There, our framework will be presented in an explanatory fashion and in a more formal way, respectively. Chapter 3, in condensed form, is included in this summary, but Chapter 4 is not.

In chapter 5, we will demonstrate how our framework can be applied in a concrete sample case. It is not included in this summary.

For those readers who are interested in a deeper elaboration, we have compiled more detailed treatments of various selected facets of our framework in chapter 6 (not included here).

Since we do not claim that our framework is all-embracing and perfect we have established chapter 7, where individual FRISCO members and associates will give their reflections and/or controversial views on selected issues. This chapter is not included.

Finally, we have compiled a reference list, a glossary and an index. Of these, those references that are quoted in the summarised material will be found at the end.

The framework presented in this report is useful for several reasons and for a variety of persons. Firstly, the mere analysis starting from simple questions (such as chapter 2) points the way to aspects that are normally just taken for granted. Secondly, the constructive tutorial of chapter 3, because of its common sense basis, is not only helpful for the understanding of students and professional information system practitioners, but should also trigger some fundamental thinking in the "information user", including the managerial level, that depends so much on that use in a broader sense. Chapter 4, obviously is intended for the theoretically inclined person, who wishes to be assured regarding the essential consistency of the framework concepts.

The worked example (chapter 5, not included in this summary) takes one by the hand showing how the thrust of the FRISCO concepts extends into a real-life situation. The additional material of chapter 6 and the reflections in chapter 7 (neither included in this summary) are food for further thought. Thus, all categories of readers mentioned in the preamble will profit.

**Was jedermann für ausgemacht hält, verdient am meisten untersucht zu werden.
(What everyone considers settled, deserves urgent investigation.)**

Georg Christoph Lichtenberg, Bemerkungen vermischten Inhalts

2 A LINE OF REASONING ABOUT INFORMATION SYSTEMS

2.1 Searching for a Starting Point

Are we, as developers of information systems, basing our reasoning and our basic concepts on a proper foundation?

The answer to this question is certainly linked to the viewpoint of the reader. For some, logic and mathematics is the most obvious answer when searching a solid foundation for formal reasoning, others might look deeper into semantics, linguistics and different theories around meaning, while still others would prefer an outlook based on social sciences.

Within our discipline, while a great variety of formalisable topics have been extensively treated, questions concerning the very nature and purpose of information systems seem not to have been addressed in a way acceptable to the community we are serving. To test a foundation for its usefulness, the goals have to be found outside the area of research itself. From this perspective, our discipline seems to be working within a too limited and barren research paradigm. To search the boundaries: What is it that makes a concept relevant?

The answer is actually very simple. In the domain of applied information system science, a concept should ultimately contribute to the creation of value in the processes of organisations - otherwise it is not relevant. Please note that, in our context, the term 'organisation' is meant in the broadest sense. It will encompass most kinds of organised human enterprises, where socially designed, goal-oriented and co-ordinated action is aimed at. A business, a project, a hunting expedition, a company, or a nation are all examples of organisations in our context. The only reason to introduce a specific concept in our domain is to create value through its use. How then does a concept create value when used?

The contribution of value by an established concept originates from its ultimate effects on the organisation as a consequence of its proper application during the analysis and design of processes and their possible IT support, enabling effective communication processes. Naturally, the application, and thereby the value addition, of a certain concept might be both direct and indirect, but, any concept not ultimately contributing to these ends ought to be severely questioned within a system of concepts for applied information system science.

Apart from enabling value addition in the analysis processes, a set of fundamental concepts ought to be as time-invariant as possible. Is it possible to avoid the contemporary terminology mess?

The key to this problem lies in proper abstraction. Naturally, the concepts required in the processes creating value, change over time. At the same time, however, the underlying concepts seem to be very stable. In our small set of concepts, you will *not find* popular concepts such as network, database or object orientation but, rather, their underlying or motivating abstractions.

2.2 Shifting the Perspective

In most human enterprises, value addition results from co-ordinated action. In any organisation of even moderate complexity, co-ordinated action is by necessity based upon effective human communication. Today, IT-based systems are instrumental in the control and management of most organisations. This might support a shift of the currently common perspective on IT-based information systems.

In our area, the predominant, technology-biased, view is that IT-based information systems mainly have the purpose of acquiring storing, processing and disseminating data. In contrast to that, the FRISCO report proposes a shift of viewpoint from concentrating exclusively on data and information, to encompass human communication and its role of supporting the creation of value in the actions within an organisation. In loose language, underlying the set of fundamental concepts is a shift from a limited how-perspective to a broader why-perspective, based on thinking in terms of effects in the organisation by the application of the conceptual framework.

Regarding information systems primarily as vehicles for communication, rather than means of data storage and processing, to some extent, demands new and complementary kinds of methods for analysis and design. This need emerges even more strongly when the value of communication acts are to be explicitly expressed in terms of their effects on other kinds of actions, performed by humans or by machines in the organisation. One should also be aware that communication - being about things and situations in a world of action and interaction - is at a level of reasoning which is quite different from a level where one is reasoning about that which is actually referred to in communication actions. In semiotics, one discusses six layers reflecting such aspects, a social layer, a pragmatic layer, a semantical layer, a syntactical layer, an empirical layer and a physical layer. We shall see that our line of reasoning in fact touches on all of these. In most contemporary views, the distinctions between, as well as the different approaches of reasoning required within, these layers are not taken into account.

Although current development models and methods sometimes employ powerful and sophisticated techniques for a great variety of analysis, some fundamental aspects of analysis and design of information systems seem to be lacking. What are the deeper reasons behind this?

The first reason is, as mentioned, that the basic goals behind, and the working principles of, information systems have not been analysed in sufficient depth. As basic a question as how information actually is used in the decision process is not properly researched [Lan66a, Ear89]. Even the very concept of information as such is used in many different meanings both in practice and in literature [e.g. SW49, CB53, Bar64, HS70, Sta71].

In our discipline, academic research has mostly concentrated on formalisable problems rather than on problems that are relevant from an economic and social point of view in the organisations and in the society at large. Similarly, implementation-oriented methodology has mostly centred around techniques used in efficient production of information systems, rather than concerning itself with effective production of information systems based on their ultimate business impact, the value addition within the organisation. How did this situation occur?

There is a second, related reason. The niche of IT has so far not really been subjected to normal economic control and proper management. It is difficult to understand that the third largest organisational cost type, surpassed only by personnel and capital costs, has been left

more or less unmanaged for so long. Although suitable application of new technologies is considered to be intimately linked to competitiveness, the strategic perspective is still mostly concerned with other matters.

In most large organisations, information is often said to be a strategic resource. At the same time, the information has no value in the balance sheet - while the technical equipment for its processing has. The information is normally not insured - the technical equipment often is. This is a very strange situation, as we know that the actual costs behind the two are of the proportion a hundred to one. What would be considered a proper ratio of value between the two?

A bizarre situation is the inadequacy of accepted principles regarding the setting of value of information. In spite of decades of serious usage and experience, there is no established view of economic calculation concerning information and IT-based systems. The resulting lack of proper economic incentives has led to an improper basis in both the use and further development of methods.

Now, how does all this link to a proper line of reasoning when determining a set of fundamental concepts?

2.3 Some Fundamental Questions to be Asked and Some Simple Answers

As mentioned, a framework for reasoning about and, ultimately, for developing methods and techniques for analysis and design, may have a variety of starting points. In search for such a starting point, we have found the following question the most obvious and fundamental one to be asked:

- **Why do we actually develop information systems at all?**

The answer is, actually, very simple:

Within an organisation, information systems are developed - or re-developed - with the aim to support value addition.

Note that the contribution to value addition in a specific business process or situation can be difficult to ensure in advance.

Based on this criterion of relevance [DS90, Kee81, Ear89, Ham90] with its origin in organisational survival, the line of reasoning is continued in seven pairs of statements, each of which is discussed and further clarified later. Before we go into more detail, our line of reasoning is presented as a compact matrix (figure 2.3-1).

We will now study how this works out in greater magnification. Our aim is now to relate the questions and their answers to a suitable set of concepts. Let us start all over again.

I: Why do we actually design and implement information systems in the first place?

In our view, organisational survival is a good starting point which implies the perspective that:

Information systems are designed and implemented within an organisation to support adequate action such that value addition may be achieved.

Starting off from business relevance means that one must elicit a common abstraction underlying most conscious efforts undertaken in an organisation. We assume that, on a high level of abstraction, the most useful starting point is the concept of value addition [McF84, PM85].

Realm	Question	Answer
Organisation economy: Aspects of organisational value / value addition	Why do we actually design and implement information systems in the first place?	Information systems are designed and implemented within an organisation to support adequate action such that value addition may be achieved.
Culture economy: Social aspects	What is the context within which actions actually are performed?	An organisation constitutes a social system, where action is performed within the frame of more or less well established goals, norms and rules of behaviour.
Communication economy: Pragmatic aspects	How do we acquire the necessary basis for decision and co-ordinated action?	The primary basis for co-ordinated action is established through communication.
Information economy: Semantical aspects	What is actually communicated?	Exclusively, conceptions or models are communicated.
Presentation economy: Syntactical aspects	How are models represented?	Models are expressed as sentences in defined languages.
Storage and transmission economy: Empirical aspects	How are sentences made time-persistent, i.e. manifest?	Sentences are encoded in patterns (subject to corruption and noise).
Encoding and detection economy: Physical aspects	How are patterns implemented?	Patterns are produced as detectable traces on physical substrates.

Figure 2.3-1: An overview of our line of reasoning

At a fundamental level, self regulation and adaptation to environmental and internal change is the basis for survival of any living system, such as a human being or an organisation. One may well agree that, on an abstract level, value addition is the key to long term survival and prosperity in almost any fruitful human undertaking. It is normally achieved by getting certain things done in a socially well-adapted manner (encompassing such elements as efficiency, effectivity, social relevance, technical innovation, etc.), and in co-operation with other people and/or machines. When several parties are involved in coherent and goal-oriented action, mutual messaging is needed. Naturally, value addition does not necessarily have to be in monetary terms.

We are then reasoning about the organisation on the abstraction level of business or organisational relevance and enterprise economy where one is concerned with concepts such as (please read the following text as one long sentence):

organisation - which might be regarded as a
system - for which different directions and aims are set, as
goals - towards which the organisation strives in order to create
added value - which normally is accomplished by coherent
actions - using certain
resources - meaning that these actions are performed by
actors - on
actands - and where these actions are aiming at changing the
state - within or external to the organisation in a desired way;

but how do current methods take such concepts into account?

Current methods of analysis and design are quite adequate in dealing with things such as value chain analysis and work-flow or process management, including the formal aspects of sequencing and synchronising actions as well as logistics. On the other hand, lack of methods encompassing efficient analysis and design of goals and key values - as well as the formal coupling between, for instance, goals and actions hampers progress [SMW91, MSW91, DS90, Ham90].

In terms of the more formal models of contemporary practice, a simple or composite action has the purpose of achieving a change from what is conceived as constituting an existing state of affairs to another state of affairs, normally called goal state. An organisation model usually defines the possible states of the organisation, with a (partially corresponding) data model defining the allowed states of a database. Organisational rules control change (effects of action) and may be said to define allowable transitions. Intentions and goals may be regarded as such rules (which may or may not be obeyed).

We have earlier established that goal-oriented, value-adding, action is paramount to survival and success of organisations. This leads to the next question:

II: What is the context within which actions actually are performed?

We take the view that culture economy is close to the essence of human interaction effectiveness and propose the perspective that:

An organisation constitutes a social system, where action is performed within the frame of more or less well established goals, norms and rules of behaviour.

Goal-oriented action is performed within, and cannot be separated from, its social context. As a consequence of the view expressed, an organisation may be seen as a system of social contracts.

When contracts are reasonably well established and generally accepted, they are said to constitute norms. These control the behaviour within the grouping they apply to. Within the frame of such contracts, possible actions and their consequences are evaluated, decisions are taken, and actions are performed.

The norms of our society also shape our world view or philosophical position. They actually form our very ability to conceive and reason about the world. They underlie the surface level of our notional action and value systems [BL66, Leo88, MH79, GD85, Gib+84, Mum83, SL94, Sta85, Sta94, Web47, Wri63].

The social level includes aspects linked to motivation, reasoning and decision making. The analysis and design work within this layer has extensive economic effects.

We are reasoning about the organisation on the social level of abstraction handling culture economy or culture effectiveness, which is concerned with concepts such as:

social system - within which the
behaviour - to a large extent is controlled by agreed-upon
norms - which normally are in correspondence with established
value (value systems) - underlying the formation of explicit organisational
rules (business rules) - which, in turn, often determine
responsibility - and authority concerning taking
decision - in matters of affecting action in the organisation.

Now, on this social level of abstraction which is mainly concerned with questions concerning culture economy, evaluation and decision, how do our methods measure up?

With a few exceptions contemporary methods, tend to, more or less, disregard culture economy. In practice, they lack adequate facilities for analysing and further dealing with the social aspects of information systems. The economic effects of applying proper methods for analysis and design of information systems on this level of abstraction seem to be severely underestimated. However, the awareness of the economic impact of handling cultural and institutionalised social questions in a professional manner is slowly having an increased impact on the methods used in practice.

As a consequence, aspects of organisational culture - the actual cement and driving forces within an organisation - normally are not analysed as part of the information systems development process. This is a significant factor contributing to the low acceptance and, hence, low effectiveness of many information systems.

Most actual organisational rules concern agreements on how things ought to be done, i.e. they are synthetic, social constructs. Although not always recognised as such, most rules implemented in IT-based systems, are of this kind too.

All conceived restrictions on the transition from one state of affairs to another one, may well be seen as rules. The transitions between states are naturally not exclusively restricted by laws of nature, but, more importantly, by "business rules" that impose different constraints on or rules for action and behaviour.

Most contemporary methods provide very weak facilities for rule extraction. Similarly, they do not allow handling large amounts of rules and structuring these in an efficient way. Object-oriented approaches suffer the same weakness, in spite of a pretence to the opposite. Methods in the expert system field - where explication of rules is the very essence - are rudimentary in linking rules to norms, and increasingly rely on conventional modelling techniques.

Consciously applying efficient management principles, today, also means that the intentions and aims have to be effectively disseminated throughout the organisation. Sometimes this is formulated as a slogan: "Act locally, think globally". Given relevant information, people must understand what to do and act accordingly - rather than merely doing what they are told. Thus, one must act on information in ways which are not customary in our current organisations on low levels of management.

New instruments seem to be required for the analysis of organisations with a high degree of autonomy and local responsibility and new schemes devised to increase competence of action within large groups of personnel.

Business development methodology, on the other hand, in some areas, shows passably strong techniques for the analysis of a variety of social aspects. Some of these techniques are currently migrating into contemporary information system development methods.

To sum up the views adopted so far, coherent and goal-oriented action is considered paramount to survival and success of organisations. Action is taken to be performed within a social system with more or less well established contracts or protocols, that are expressed in terms of norms or rules. Action is, furthermore, based on evaluation and decision. This begs a further question:

III: How do we acquire the necessary basis for decision and co-ordinated action?

As mentioned earlier, we regard communication and communication economy as the lynch pin of our discipline, and promote the perspective that:

The primary basis for co-ordinated action is established through communication.

Within the context of an organisation, seen as a social system, co-ordinated action is brought about either by communication between actors or through direct observation of the state of affairs - where adequate action is determined by common experience and norms, which, in turn, are the effects of earlier communication.

Communication has the ultimate purpose of sharing intentions regarding future behaviour - including the formation of norms within which action is undertaken and valued. In other words, it concerns what is to be done in a specific situation or keeping the world view up to date. Naturally, communication may supplant direct observation, e.g. by merely providing a description of the state of affairs. On the other hand, in special cases, when norms are sufficiently action-oriented, direct observation of the state of affairs may substitute explicit communication acts. Among other things, communication serves the purpose of providing a basis for comparing the existing state of affairs versus some desired state, thus enabling the evaluation of alternative actions, possibly negotiating alternatives between various interested parties.

We are reasoning about the organisation on the pragmatic level of abstraction handling communication economy, which is concerned with concepts such as

observation - may lead to actions such as
speech acts - which are performed in accordance with established
protocols - and among other things convey the
intentions - necessary to obtain required
effects - in terms of actions to be performed.

At this level, intentions behind and effects of various messages are in the focus of analysis. Yet, contemporary methods are very weak in analysing these aspects [Aus62, Hab84, Who47, McC91, DS90, Sea69, SV85].

Future methodology development, in general, might benefit from a paradigm shift. Tradition of the past two decades has been that the fulcrum of the information system is its database. It contains data that has to be acquired, processed and delivered to various parties at the right

points in time. In business, this paradigm is increasingly invalidated because of the extent to which the modern organisation is making itself dependent on fresh information for decision making. While most traditional approaches in information analysis have worked well for situations that are controlled by a sender or distributor, those approaches do neither work well for a consumer-controlled information acquisition situation nor for a negotiation-oriented work situation with more autonomous actors.

Positive trends may be observed in the emerging application of speech act theory and, to a limited extent, object-oriented methods for organisation analysis and design. Speech act theory is slowly gaining recognition, bringing the pragmatic level of business abstraction into focus. The theory has to be further developed to encompass other types of acts as well as being broadened in context [Aus62, ALL88, AHL92a, AHL92b, FGHW88, LL86, HKL95, MWFF93, Sea79, Sea95, SH96, SV85]. Object-oriented organisation analysis methods, where messages are modelled so as to invoke protocol determined services from the objects they are sent to, may also lead to improvement in the basic view of how an organisation works. So far, however, existent analysis methods are fairly technical and lack the necessary integration of knowledge from different disciplines such as philosophy, organisation science, sociology, psychology and linguistics. Inclusion of such views is indispensable if relevant effects on relevant semiotic levels of information systems are to be captured by one's method.

Naturally, communication, itself, constitutes action. It follows more or less established protocols or norms. Taking part in such an action demands sufficient adherence to applicable norms (or protocols in terms of social contracts), so as to achieve controllable effects.

To sum up the views adopted so far, coherent and goal-oriented action is considered paramount to survival and success of organisations. Action is taken to be performed within a social system with more or less well established contracts on the basis of communication. This begs a further question:

IV: What is actually communicated?

As a basis for reasoning about information economy, we would like to propose the slightly unconventional perspective that:

Exclusively, conceptions or models are communicated.

Basically, communication aims at establishing sufficient conceptual correspondence or dissonance between actors to enable adequate action. Note that, in the FRISCO setting, a model or a conception can be given a multitude of representations. Another way is to say that communication aims at conveying beliefs and intentions to facilitate action.

We often forget our complete dependence on conceptions or models. When a person tells something about something to another person, matters are, of course, about conceptions of that something, not about the something as such. Even if instruments are used for direct measurements, underlying assumptions - often called the observation model, which might be seen as a model on a higher level of abstraction - will affect both the outcome and the interpretation of one's findings [Wil91, Dru92, Che81].

Semantically, conceptions or models are made known without regard to how that is done, i.e. from a semantical point of view we are indifferent to what medium, e.g. speech or handwriting, or what language is employed - as long as the effects are the intended ones.

Meaning (semantics), as a concept, may be treated within a variety of more or less unstable theories. Often, the notion of semantics is equated with the intended or actual interpretation of an observed sentence. It may also relate to a correct and/or useful interpretation of an observation of a state of affairs. This means that concepts such as truth and correspondence are treated at this level of reasoning.

Meaning and semantics are often coupled to the notion of total understanding. At the same time, conveying to a receiver all the connotations a sender has to a certain concept is probably impossible.

However, in daily life and in running an enterprise, there is no need for total understanding. To achieve sufficient model correspondence between actors to facilitate adequate action is close to the goal of most information systems. Often, not even understanding but rather adequate response in terms of action is what is demanded [Nil79].

We are reasoning about the organisation on the meaning level of abstraction, thus handling aspects of interpretation and expression economy, which is concerned with concepts such as

conceptions - or
models - present in the minds of people as
knowledge - transferred and conceived through communication as
information - all of which carry a certain
meaning.

Now, to what extent do contemporary methods cover this level of abstraction? To some extent very well, to some extent only poorly.

The concept of communication has both a semantical and a syntactical aspect which have to be considered together.

The cost of mis-interpretation of available data is very high, although it may be difficult to assign a motivated value in every instance. At any rate, this is a totally neglected area with vast economic consequences [Car47, Cho68, Oco75, Kos76, Lan66a, Lan66b, Lee76].

Understanding data about the organisation requires a context for correct interpretation. This context is normally provided by what is usually called business model, data model or conceptual schema. Within these, rules for interpretation are given by structural implications (part of the semantics is the placement in a conceptual structure) and by general textual descriptions of different conceptual components.

Conceptual modelling (data modelling, process modelling, behaviour modelling and the interplay between these techniques) has been given very much attention during the two last decades. There is, still, a most unfortunate focus on theory concerned with the syntactical aspects of concept modelling, while the problems related to model semantics and to the pragmatics of utilising the models are often neglected. This leads to unnecessary costs in terms of demands for change of applications and databases. Although the practitioner is well aware of the fact that a multiperspective view is needed in modelling, the academic concentration on a few perspectives has led to a noticeable lack of well founded theory and instruments.

To sum it up: Coherent and goal-oriented action is needed to create value. Action is performed within an organisation which might be regarded as a social system with established contracts. Action is, furthermore, based on evaluation and decision which, in turn, is based on the availability of information in terms of results from direct observation or communicated conceptions or models. Now, this leads to a further question:

V: How are conceptions or models expressed?

Reaching the level of data economy, i.e. presentation and manipulation economy, we take the perspective that:

Conceptions or models are expressed as sentences in defined languages.

When reasoning about the organisation on the syntactical level of abstraction, we are concerned with (agreed rules for) languages. This means that on this level of abstraction, we deal with the economics of formulation, presentation and manipulation and are concerned about concepts such as:

representation - of conceptions or
model denotation¹ - being equivalent to a set of
sentences - which are expressed in a
language - which has a defined
grammar - stating the structure within which
references - or terms are to be interpreted, i.e. related to concepts.

Actually, like the rest of the levels of abstraction treated so far, proper reasoning within one level only is not possible. In this chapter, this is done only as a didactic instrument. A central concept such as reference, for instance, has to be treated on both the semantical and syntactical level in parallel.

Nevertheless, we are now on a level of abstraction where good theory is available as well as good working models and procedures. Actually, most of the methods available concentrate on this level.

To sum it up again, coherent and goal-oriented action is needed to create value. Action is performed within an organisation, a social system with established contracts for action - including communication. Action is based on decision. Decision is based on the availability of information resulting from interpretation of direct observation or communicated conceptions or models. Now as action takes place over time and the knowledge required spans the gap of time, we have to ask:

VI: How are sentences made time-persistent or manifest?

Sentences are encoded in patterns.

On this level of abstraction we are concerned with storage and transmission economy. The next question to ask, to complete our line of reasoning, is obviously related to what actually carries these patterns and makes them detectable.

Rather good theory exists on these levels of abstraction, e.g. signal theory [e.g. SW49, Khi57, Sca82]. On the other hand, these theories are only rarely taken into account in the analysis related to organisational evolution and information system development - although the concepts are valid on most levels of analysis.

¹ The term 'denotation' has several meanings in the English language. We use it exclusively in the sense of 'precise and unambiguous representation'.

In this context, the following concepts may be worth considering:

pattern - which is directly linked to the requisite
variety - where the communication may be influenced by
noise - which might be countered by introducing
redundancy - into each or the stream of
physical tokens.

We will now conclude our reasoning without really in depth considering the last question in the previous table (figure 2.3-1) - which links back to basic ontological aspects - but rather go back to the question of layering a line of reasoning. The questions asked above have significance on different semiotic levels. While behaviour and norms mainly belong to the business and social level, models, for instance, have syntactic, semantic, pragmatic and social aspects, depending on what one wishes to stress. When assigning a specific meaning in connection with a particular choice, one must be clearly aware of the semiotic level one addresses, because the impact of these choices is of a fundamental nature. Restricting oneself to one level only, may result in serious difficulties of mutual understanding within our realm of research.

The approaches to a number of philosophical questions are, in fact, the ones that have the most dramatic practical impact during analysis and design of organisational systems and their information systems. As an example, the choice of a hard-nosed naïve realist's position will, in practice, almost certainly be a good guarantee for creating a modelling disaster. At the same time, a theory built on this position might look beautiful and withstand any formal attack.

2.4 Towards a Raison d'être

The report proposes a change of emphasis regarding method development and information system analysis. We judge it profitable for the community at large to expand from its strong data and information orientation towards a broader view including human communication and its effects in terms of actions in organisations. Why?

Value addition is created through action. The basis for co-ordinated, goal-oriented action in organisations is inter-human communication. It is asserted that information systems, therefore, ought to be regarded primarily as communication vehicles rather than only means for storing and processing data [Nil79]. Adopting this view evidently means that different demands are placed on methods for analysis and design, and, as a consequence, on the basic conceptual framework.

Following our criterion of relevance as based in value addition, the concepts of information and communication cannot be treated profitably without a fairly broad interdisciplinary involvement, among other things as a consequence of the need to view the organisation as a social system.

This is not saying that the traditional formalisms, methodologies and conceptual frameworks are without value. However, many might profit in their development from following a different line of questioning, which derives from goals such as the ones discussed.

2.5 Summary of Concepts

Many of the concepts mentioned in section 2.3 are explicitly defined in chapters 3 and 4, while others will appear in the accompanying text of chapter 3. The following list is meant to tell where these concepts will appear ¹ :

Concept:	Chapter 3:	Chapter 4:
organisation	3.8 : A[m], E39	4.5 : D45
system	3.7 : A[l], E30	4.4 : D40
goal	3.3 : E17	4.2 : D23
added value	3.8	
action	3.3 : E14	4.2 : D17
resource	3.3 : E15	4.2 : D21
actor	3.3 : E13	4.2 : D19
actand	3.3 : E15	4.2 : D20
state	3.2 : E7	4.1 : D9
social system	3.8 : A[m]	
behaviour	3.2	
norm	3.8 : E39	
value	3.8	
rule	3.2 : E12	4.1 : D16
responsibility	3.1	
decision	3.8	
observation	3.1, 3.4	
speech act	3.8	
protocol	3.2, 3.5, 3.8	
intention	3.5	
effect	3.3	
conception	3.4 : E20	4.3 : D26
model	3.6 : E26	4.3 : D30
information	3.8 : E36	4.3 : D36
knowledge	3.8 : E33	4.3 : D33
meaning	3.5	
representation	3.4 : E23	4.3 : D28
model denotation	3.6 : E26	4.3 : D31
sentence	3.4 : E22	
language	3.4 : E22	4.3 : D28
grammar	3.4 : E22	
reference	3.4 : E24	
pattern	3.5	
variety	3.5	
noise	3.5	
redundancy	3.5	
physical token	3.5	

¹ Legend:

3.n or 4.n indicate the relevant sections (chapter 3 or 4, respectively).

En or Dn indicate the corresponding definitions in chapters 3 or 4, respectively.

A[x] indicates the corresponding assumption in chapter 3.

A scientific field can arise only on the base of a system of concepts.

Russell L. Ackoff

3 INFORMATION SYSTEM CONCEPTS: An Integrated Overview

"Information systems" concern the use of "information" by persons or groupings of persons in organisations, in particular through computer-based systems. A proper understanding of the nature of such systems requires viewing them in the context of the organisations that employ them. Consequently, concepts from a variety of disciplines are needed for describing any part of those systems and certainly for providing a sufficiently coherent picture of them as a whole. This chapter attempts - in tutorial form - arriving at a pragmatic minimal set of interrelated concepts that may serve this purpose.

A fundamental problem when trying to reach consensus is the difference in philosophical or metaphysical position of those concerned. This does not only hold for the culturally different communities of information system professionals and information system users but even applies within those groups. They may feel that there is a common understanding of many general terms, for instance "world", "thing" and "person". Yet, even for these - and certainly for more specific concepts - various philosophical positions may be observed of how they are viewed. We will concentrate on just two of them¹:

- **Objectivist (Naïve Realist):** somebody who believes that "reality" exists independently of any observer and merely needs to be mapped to adequate descriptions; for the objectivist, the relationship between "reality" and some model thereof is trivial or obvious.
- **Constructivist:** somebody who also believes that "reality" exists independently of any observer, but who is aware of the fact that we only have access to our own (mental) "conceptions"; for the constructivist, the relationship between reality and conception is principally subjective, and may be subject to negotiation between observers; any agreement - which we call "inter-subjective reality" - may have to be adapted from time to time.

As we shall justify later, the *constructivist position* is felt to be most appropriate for our purpose. Many concepts in our area are not yet fully established. Hence, we find ourselves participating in a negotiation process whenever we discuss and hence "construct" any piece of "reality" [Sea95]. Fortunately, this does not hold for every term occurring in the following sections. This sort of negotiation process applies also to the development or re-development of any concrete information system, where the developers have to construct appropriate models of the "reality" of the organisation.

Because a "professional language" has to be based on an area of practical experience where a common societal negotiation and construction process has already resulted in a framework of everyday language terms, we need not repeat this process here. Some terms are either widely accepted or will be understood with only limited further explanation. Nevertheless, we will try to build up our framework of concepts "from scratch". This is because we feel that due to the

¹**Warning:** The terms 'objectivist', 'objectivism', 'constructivist', 'constructivism', etc. are sometimes used in this meaning and sometimes in various other meanings in the literature!

many terminological uncertainties in our field as a whole, it would be quite dangerous to take any term used in our field for granted or well understood.

Our framework of concepts is based on a number of assumptions, which we will state explicitly. The "definitions" we will introduce in this chapter are not meant to be "formal" in any sense. They are rather intended to be explanatory, because of the tutorial character of this chapter. To avoid that the number of definitions would be didactically overwhelming, we will often group closely related definitions to appear in one cluster.

Our ultimate objective is to define and discuss concepts that are specifically related to the area of "information systems". We found however that it is necessary to built up our conceptual framework from a more general background.

As a starting point (section 3.1), we present an overall view of the "world", including our metaphysical or philosophical position, in other words, our "Weltanschauung". It takes the form of a number of assumptions in which all concepts of interest are introduced and sketched informally.

Having laid this intuitive foundation, we will present the most fundamental layer of our conceptual framework, that is to say, a coherent and logically consistent set of concepts for viewing the "world" (section 3.2). In section 3.3, we will apply the concepts of the fundamental layer, to discuss and introduce the notions of actor, action and related issues, which are highly relevant for our purpose.

In sections 3.4 - 3.11, we will use the previously introduced concepts to explain the various concepts we ultimately need for our conceptual framework, going step by step from general concepts to more specific ones.

As a consequence of our constructivist approach, we must take into account what representations or signs we use to express our conceptions (section 3.4). Extending the view of an individual user of representations or signs to that of joint usage and exchange, i.e. actual communication, we find that a number of restrictions apply (section 3.5). This will lead to a discussion of models (section 3.6). A further section (3.7) is devoted to systems - the complex dynamic structures one encounters throughout our area.

In sections 3.8 - 3.9, we will apply the previously defined concepts to our specific area of interest, i.e. that of "organisations" and the "information" they use. "Organisational systems" and their components are discussed first (section 3.8). This is followed by a general analysis of the true objective of our study, the "information system" (section 3.9).

We will specify the components that make up "computerised information sub-systems", and we will mention a few classifications thereof (section 3.10). These sub-systems are what most technical people (sloppily) call "information systems".

Finally, section 3.11 provides a summary of our conceptual framework, i.e. all assumptions and definitions.

3.1 Our "Weltanschauung"

First, let us consider the organisational context. An "action" within or in connection with some organised social structure, such as an enterprise, will be undertaken because it is deemed useful or necessary. How does an individual or grouping of individuals (for instance a "department") come to the conclusion that any action is called for? It may be a matter of responding to some triggering event. More generally, it will result from judgement, based on an assessment of the situation, combined with experience. These are typical elements of what we call "information".

However, information does not come about by itself, but rather through some received "signal" or "message". Whatever the case, it is a question of conception (of some state and/or changes of a state in the "world") and subsequent interpretation (that it is relevant).

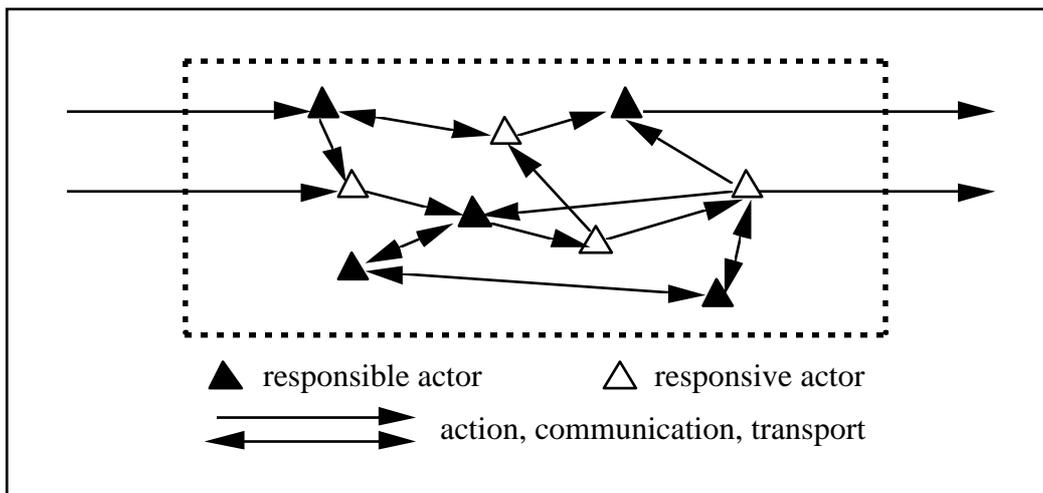


Figure 3.1-1: A world of interacting "actors" or "agents"

Action, therefore, appears to be motivated by "information" explicitly provided for that purpose, or extracted from some past experience. An "actor-based model" of the world may even be extended to include message-triggered actions of independently (but not autonomously) operating "machines" (or "devices"). They may be distinguished by calling a human (or group of humans) a responsible actor and a machine a responsive actor. Computers and computerised systems may be viewed as serving in the latter capacity (see figure 3.1-1).

The most characteristic aspect of this model is the interaction. What, precisely, is the role of the various actors in it? What is the nature of the information they act upon? What do they exchange when interacting? What characterises the systems assisting in acquiring and disseminating information? A quick answer to these questions is:

- that action is undertaken on the basis of some special knowledge - called information,
- that the latter is shared out by a special process of interaction - called communication,
- in the course of which (often symbolic) messages are exchanged - called sign carriers or representations, and
- that the set-up of the world under consideration and its information providing arrangements may be viewed as displaying a special form of cohesion - called system.

While intuitively appealing, this brief introduction implies acceptance and understanding of at least five difficult concepts: knowledge, information, communication, representation and system. An attempt will be made to relate these in a logically consistent fashion such that a

useful conceptual framework results. For this purpose, we have to begin with a number of fundamental underlying assumptions.

The most characteristic aspect of our field is that it exclusively deals with statements (data, messages, etc.) *about* the organisational world we are interested in. One's first intuition would be to say that such statements should represent "reality". A brief reflection will clarify why such an approach leads us into all sorts of traps.

The natural sciences, and to a lesser degree, the social sciences, as such, deal with reality by making predictions about the outcome of experiments, often in some probabilistic sense. In the information system field, on the other hand, there is the problem that many of its representations are subject to personal interpretation. No one doubts that one may record meaningful data regarding stocks, orders, tasks and academic achievements. However, numerous "facts" in society only hold true because they have been agreed, such as the value of money, the validity of a marriage or the legality of the government. As argued in [Sea69], these might be considered "institutional" facts, which only concern "reality" to the extent that they resulted from some societal negotiation. Social reality is essentially "constructed" [Sea95], and the same applies to that which makes up "information".

The above mentioned action triggering, for instance, always involves two parties. The sender of a message generally has the intention of "informing" specific or general receiver, but the person receiving the stream of symbols that make up that message may partially or wholly misunderstand its meaning. Sender and receiver may well have analogous ideas about the subject of the message, but there is no way of deciding whether these ideas (or "*conceptions*" as we shall call them) are in any way identical or even nearly the same. Similarly, someone seeking "information" from a personal source (say, by telephone) or a computerised service (say, some database), will not necessarily form the precise conception that the originator of the source intended.

The answer to these problems is not even to refer to "reality" in our framework, but to restrict our thinking to personal conceptions and interpersonal exchanges of representations of such conceptions. When agreement is reached as a result some exchange, we have clearly arrived at a jointly "constructed" view of a part of the "world".

It is this position that we call *Constructivism*. We will first introduce it intuitively, through the following assumptions:

Assumption [a]:

The "world" exists, independent of our own existence, or of our cognitive and intellectual capabilities.

Assumption [b]:

Human beings are able to observe and perceive "parts" or "aspects" of the "world" (which we will call domains) with their senses, thus forming perceptions in their minds. Perceptions can be considered as specific patterns, generally changing in time.

Assumption [c]:

Human beings are able to form conceptions in their minds, as a result of current or past perception, by means of various cognitive or intellectual processes, such as recognition, characterisation, abstraction, derivation, and/or inner reflection. The collection of (relatively) stable and (sufficiently) consistent conceptions in a person's mind is called his or her knowledge.

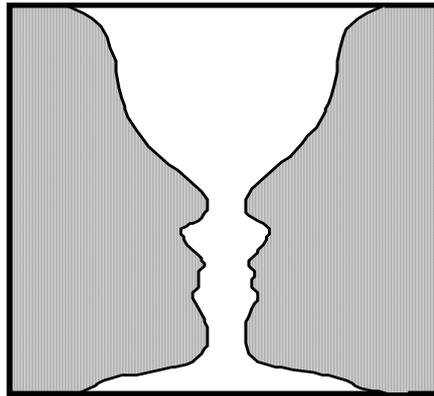


Figure 3.1-2: A wineglass or two faces?

Out of one specific perception, human beings can in principle construct any number of conceptions. Figure 3.1-2 shows an example. The assumption is that a person sees something which looks like this picture, and has no chance to investigate it further (e.g. by changing the viewing angle, etc.). In this case, the person will probably interpret what he or she sees (this specific pattern, a perception) as (at least) two possible conceptions (a wineglass or two faces). In most cases, a single human being is constructing out of one perception exactly one conception at a time. In case of various interpretations of a particular perception, human beings will usually try to resolve those interpretation conflicts by further investigating the domain and will try to get further perceptions thereof. It may be desired, but it is by no means guaranteed, that different people construct out of presumably one and the same perception exactly one and the same conception. Resolution of those interpretation conflicts between different people will usually be attempted by communication.

Assumption [d]:

A perceived domain may be conceived as composed of identifiable components, which we call things; they may overlap, contain each other or relate to each other in whatever empirical way.

Assumption [e]:

Some things are conceived as having a static existence (states), while others are conceived as changes of some state (transitions). Hence, a perceived domain may be conceived as having an existence in a temporal context.

Assumption [f]:

Some transitions may be conceived as being performed or brought about by some active things, called actors. Such a transition, called an action, is performed by that actor on passive things, called actands. A rational action by an actor is said to be in pursuit of a goal. It is possible that the cause of a transition cannot be attributed to a specific actor, but merely comes about because of, or is triggered by, some aspect of the preceding state.

The conception of action-by-an-actor-on-actands, with the implied quality of causation, is fundamental to any view of a "dynamic" domain. We may further distinguish: (a) human and hence responsible actors (or groups, organisations, societies of persons), and (b) non-human and hence non-responsible and often inanimate and merely responsive devices (in particular machines and computers).

Assumption [g]:

Persons use representations to communicate their conceptions. These conceptions are represented in some language on some medium.

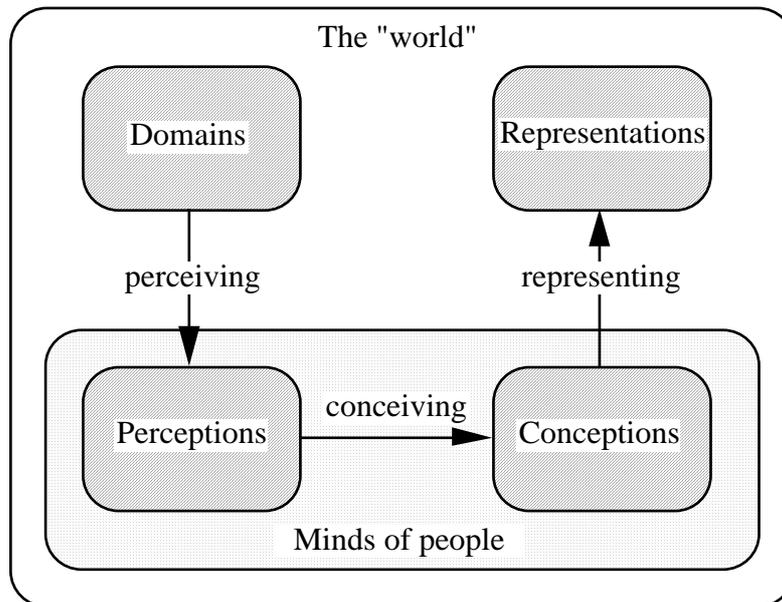


Figure 3.1-3: Perceiving, conceiving and representing domains

These assumptions are summarised and illustrated in figure 3.1-3. Not shown in this diagram is the overlap of the perceived domain with perception, conception and representation. The latter items may be domains being subject to perceiving, conceiving and representing actions themselves. This is obvious for representations, since they belong to the "physical world outside our minds" and may be therefore subject to these actions as any other domain outside our minds. However, perceptions and conceptions "in our minds" may certainly be subject to these actions as well, in particular when performed by neurophysiologists, psychologists, cognitive scientists, epistemologists, ontologists and similar scientists.

In representing conceptions, human beings are most often using symbols, such as letters, characters, character strings (labels, names, etc.), to refer to their conceptions. The use of such symbols is governed by the rules of languages. A language may be formal or informal, character-string-oriented (one-dimensional), graphical (two-dimensional) or even n-dimensional ($n > 2$), etc. A representation medium may be suitable for permanently storing these representations (e.g. sheets of paper, books, magnetic disks or tapes), or unsuitable for this (e.g. the air carrying sound waves).

Representations are used for communicating conceptions among different persons. That is, one person is forming representations of his or her conceptions, and another person is perceiving these representations and is constructing conceptions thereof, which in turn may be represented as well. The formation of representations by the one human being and the processes of perceiving and conceiving these representations by the other human being may occur at (roughly) the same time (direct, immediate communication), or at different times (indirect, delayed communication via a permanent storage medium, such as a book or a magnetic disk).

For a constructivist, existence in any real sense is unimportant. A person has only access to his or her own conceptions, and we cannot even be sure that our conceptions and views of a domain correspond to "reality" as such. On the other hand, representation is a vehicle for communicating a (personal) conception. By it, a group of people can try to arrive at a

commonly shared understanding of representations through discourse. When a group agrees on the meaning of a particular representation, we will call its interpretation a *shared conception*. It is then assumed that there is a unique domain it refers to, i.e. an inter-subjective reality.

Given these fundamental assumptions, we will now address the question as to what structures conceptions might have. We realise of course that this is a difficult question, since we do not have a straightforward, direct access to the conceptions in our minds. However, we assume that it is very well possible to come to reasonable answers to this question, by means of "*internal mental inspections*", and/or by means of analysing the "*deep structure*" [Cho68] or the "*conceptual essence*" [ISO82] of representations. The latter kind of analysis requires of course "*internal mental inspections*" as well. Pursuing this question by such inspections or analysis, and specifying the results thereof, have in fact a long tradition and relate to the fields of *epistemology* and *ontology*¹, respectively.

We shall refer to the result of our quest as "*our conceptual framework*". Of course, we do not pursue the aim of developing a universal ontological theory. Rather we restrict ourselves to the presentation of an ontological view suitable for describing and explaining the most important phenomena and concepts in the information system field only.

One major problem regarding such an ontological view has to do with the large variety of different conceptions or views prevailing in the information system field. We found however that this large variety of views can, in principle, be harmonised. We achieve this by structuring our ontological view in several layers, beginning with a fundamental layer specifying a fairly universal and generic view. From this fundamental layer, we derive a number of other layers, specifying more specialised views which are convenient for discussing information systems.

This layered approach stems from a basic principle underlying our framework, that is, it is based on a minimal set of ontological assumptions. In other words, we aim at a maximally integrated ("tightly coupled") set of concepts for describing and explaining the most important phenomena and concepts in the information system field. This minimality principle distinguishes our approach from several alternative approaches in which the requirement of minimality is not pursued, resulting in a large and "loosely coupled" set of concepts.

Whenever one attempts to build a conceptual framework "from scratch", one has an inherent "bootstrapping problem". In our case, we try to overcome this problem

- (1) by postulating a number of concepts deemed useful for constructing conceptions (sections 3.2 and 3.3),
- (2) by returning to the notion of conception and their construction and representation, to discuss these issues in more precise terms using the introduced concepts (sections 3.4 and 3.5), and
- (3) by introducing various specialisations of these considerations, to arrive finally at the notion of information system (sections 3.6 to 3.10).

¹The term 'ontology' is used in different meanings in the literature. Here and throughout the report we shall use this term - in accordance with our constructivist position - as meaning that department of metaphysics concerned with "theories" or "views" of how human beings *think of (i.e. "conceive") the "world"*. A particular theory or view will be called "ontological theory", "ontological view" or "ontological position". It is based on an unquestioned (but reasonably described) metaphysical position, or "*Weltanschauung*". Whereas an ontological view contains and correlates ontological categories or generic concepts representing specific components, aspects and features in a precise way, a "*Weltanschauung*" is an "inter-subjective" common understanding, expressed in informal terms.

3.2 The Starting Point of Our Conceptual Framework

The above assumptions are not necessarily "true". However, we do not know an absolute, objective truth, in the first place. What we can and will do is to use the concepts mentioned in a coherently argued view of our problem domain.

To begin with, we should remember our ultimate purpose - to achieve a coherent view of all aspects of information and communication in an organisational context. The means of acquiring, maintaining and communicating information are associated with physical aspects of the world, such as spoken or written messages, computers, networks and the like. Therefore, before we will discuss information and communication in more detail, we will look for ways in which we can construct conceptions of domains, in general.

Assumption [d] implies that when an overall conception is formed about a domain, it may be viewed as decomposable. Both the whole and the parts in such a view are indentifiable "things" to the person concerned. One might say that

things are conceptions.

Can we make such a statement and make sense? At this moment, it certainly is impossible to "open" a person's mind and "look" into it! On the other hand, it is very well possible to answer that question what the ingredients of conceptions are by *indirect* means, such as self-inspection or empirical tests based on communication. Giving answers to this question is in fact what cognitive scientists, ontologists and modelling methodologists have been doing for ages.

When communication with another person is intended about (aspects of) a domain, the first person will represent his/her conceptions, say, by speech or in writing. The corresponding utterances may be perceived and responded to in the same manner (Assumption [g]). Without making assumptions about any "existence" of what conceptions refer to, the constructivist position implies that persons can and do exchange views *about* their conceptions - couched in terms of "things". One might, therefore, also say that

conceptions are made up of things.

The normal personal way of perceiving and conceiving some domain is a matter of subconscious (and probably imperfect) generalisation. When one makes a conscious ontological analysis, including the inspection of numerous examples, a more precise generalisation may be achieved. And by an exchange of views, one may arrive at some jointly accepted ontological view. While the conceptions, as such, are not accessible, indirectly they are available for discussion and hence operationalisable.

Now, a proper ontological view should be capable of modelling any "part" or "aspect" of the "world", including the notions of 'perceiving' and 'conceiving', themselves. Upon further analysis, we find that these are actions, and similarly, that 'perceptions' and 'conceptions' are actands (see section 3.4). If, remaining within our constructivist view, we generalise as far as possible, we find that the actions 'perceiving' and 'conceiving', and the actands 'perception' and 'conception' are things, as well.

If one were to make a very shallow (and in fact mistaken) analysis, one might come to the conclusion that there is a definitional loop (things are conceptions, vs. conceptions are things). This would be a complete mis-understanding of the matters involved! In the first case, we are developing an ontological view by making well-educated assumptions about the "ingredients" of conceptions. In the second case we are using the resulting (ontological) view to model the

notion of ‘conception’ as the output actand of a ‘conceiving’ action. Thus, there is no definitional loop at all!

Let us now come to more concrete considerations and have a look at the following simple example of some sentences in natural language:

- (a) Ludwig Wittgenstein is a person.
- (b) Philosophical Investigations is a book.
- (c) Ludwig Wittgenstein is the author of Philosophical Investigations.
- (d) Philosophical Investigations is written by Ludwig Wittgenstein.

The conceptions represented by these sentences, i.e. their "interpretation", is of course straight forward in this simple example (provided one knows about this author and this book). We could for instance say that the string of characters ‘Ludwig Wittgenstein’ refers to a "*thing*", which we are characterising in specific ways. The same holds for the string of characters ‘Philosophical Investigations’. The strings of characters ‘is a person’, ‘is a book’, ‘is the author of’, and ‘is written by’, characterise these "things" in specific ways. These characterising elements can be viewed as special "things" too, which we will call "*predicators*" [Car56, KL73]. The complete sentences (a - d) refer to "*relationships*" where these "things" are involved. Each one of the simple sentences (a) and (b) refers to a "*unary* relationship". The sentences (c) and (d) refer to one and the same "*binary* relationship", that is, (c) and (d) are *two* different representations of *one and the same* conception.

With these preliminary considerations in mind, we can now commence with our definitions. We start out with the most general concept of all, "thing".

Definition^{1 2} E1: Thing³

A **thing** is any part of a conception of a domain (being itself a "part" or "aspect" of the "world"). The set of all things under consideration is the conception of that domain. •

Examples:

- A thing called ‘Ludwig Wittgenstein’;
- A thing called ‘is the author of’;
- A thing referred to as ‘Ludwig Wittgenstein is the author of Philosophical Investigations’;
- A thing called ‘HMS Atlantis’;
- A thing referred to by the phrase ‘The convoy of ships consisting of HMS Atlantis, HMS Gondwana, and HMS Utopia’.

This definition is of course too general to be of any practical value. To be able to structure our conceptions in a meaningful way, we need to introduce a number of special things. Firstly, to distinguish the things we wish to characterise in some way, from the things that characterises them, we introduce the special concept "predicator".

Definition E2: Predicator⁴, Predicated thing

¹ The following conventions are used in the definitions:

Term in bold: A concept defined here

Term underlined: A concept defined elsewhere (mostly but not necessarily in a previous definition)

² **Note:** Some forward references are used, since these "definitions" are only informal and explanatory. In a formal approach (see chapter 4), this is of course not possible and is avoided by introducing primitives. For example, the notion of thing will be an undefined primitive in chapter 4.

³ **Warning:** The term ‘Object’ is sometimes used in this meaning and sometimes in various other meanings in the literature (e.g. for what we will call an ‘Entity’)!

⁴ **Alternative terms:** ‘Role’, ‘Qualifier’

A **predicator** is a thing, used to characterise or qualify other things, and assumed as being "atomic", "undividable" or "elementary". •

Notes:

- Predicators allow us to assign any kind of quality, characteristics or involvement to a thing, in a general sense. It allows furthermore and in particular to construct "relationships" between things.
- Predicators are usually denoted as verb phrases in western natural languages.

Examples of predicators:

The things called 'is a person', 'is a book', 'is the author of', 'is written by'.

A **predicated thing** is a thing being characterised or qualified by at least one predicator. •

Note:

The intersection of the set of predicators and the set of predicated things is not necessarily empty, but will usually be so.

Since conceptions are usually not just sets of unrelated things, we must introduce the concept "relationship". It is expressible in terms of the concepts predicator and predicated thing.

Definition E3: Relationship

A **relationship** is a special thing composed of one or several predicated thing(s), each one associated with one predicator characterising the role of that predicated thing within that relationship. •

Note:

In other, more formal terms: A relationship consists of a non-empty set of pairs, whereby each pair consists of a predicated thing and a predicator. The cardinality of this set of predicated-thing-predicator pairs is usually quite small (often two). According to this cardinality, the relationships are called unary, binary, ternary, etc.

Examples:

- A representation of a unary relationship:
{<predicated thing, predicator>}:
Ludwig Wittgenstein is a person.
- Two representations of one and the same binary relationship:
{<predicated thing 1, predicator 1>, <predicated thing 2, predicator 2>}:
Ludwig Wittgenstein is the author of Philosophical Investigations.
Philosophical Investigations is written by Ludwig Wittgenstein.
- A representation of a ternary relationship:
A particular department orders a particular article from a particular supplier.
- Since a relationship is also a thing, it may be characterised by predicators itself, i.e. may be related to other things:
A particular student attends a particular course (a binary relationship);
A particular mark is granted for this attendance
(a binary relationship between a thing and the first relationship).

Since a thing is any conception of a domain, a non-empty set may be also viewed as a thing, consisting of a number of other things. In this case, there exists a special relationship, the omnipresent "set membership". The concepts "elementary thing" and "composite thing" can be defined on this basis.

Definition E4: Set membership, Elementary thing, Composite thing

A **set membership** is a special binary relationship between a thing (the set), characterised by the special predicator called 'has-element', and another thing, characterised by the special predicator called 'is-element-of'. •

Note:

Specific constraints must be added to avoid paradoxes. For instance, a thing being a set of things, may not be contained in that set itself. We dispense here with further details and refer to chapter 4 and to the appropriate literature on set theory.

An **elementary thing** is a thing, not being a relationship and not being characterised by the special predicator called 'has-element'. •

A **composite thing** is a thing, not being an elementary thing. •

Notes:

- A relationship is a composite thing.
- A set, "collection" or "aggregate" of things is a composite thing.
- An "aggregate" is usually considered not just as a *set* of "*loose*" components, but as a *composite thing* containing also relationships between these components (such as the spacial position of the components, etc.).
- A predicator is assumed to be always an elementary thing.

Examples:

- A convoy of ships contains a number of particular ships.
- A particular ship contains a number of ship components plus a number of relationships between these components indicating their spatial position.

On these grounds, we can now introduce the widely used notion of "entity".

Definition E5: Entity¹

An **entity** is a predicated thing as well as an elementary thing. •

Examples:

- The entity called 'Ludwig Wittgenstein' (characterised e.g. by the predicators called 'is a person' and 'is the author of').
- The entity called 'Philosophical Investigations' (characterised e.g. by the predicators called 'is a book' and 'is written by').

Notes:

- The reason to define an entity as a predicated thing is to distinguish it from a predicator.
- Whether or not some predicated thing is considered to be elementary (i.e. an entity) is dependent on the context in which that predicated thing is used. That is, the notion of elementariness depends on the level of "granularity" chosen in a particular situation. Whenever this situation changes, the elementariness or non-elementariness of a predicated thing may change as well.

Example:

A captain of ship may view his ship as a composite thing, containing a number of ship components plus a number of relationships between these components indicating their spatial positioning. An admiral, on the other hand, leading a convoy, may not be concerned with ship components and thus may view any single ship of his convoy as an entity.

We are building up our conceptual framework basically from an extensional point of view. Whenever we speak of a thing, a relationship, an entity, etc., we mean an "instance", not a "type". Sometimes however, it is necessary to apply an intensional point of view, and to use the notion of type of things. An intensional definition of the notion of type relies on that of characterisation (e.g. predicates).

¹**Warning:** This term is sometimes used in different meanings in the literature!

The most elementary characterisation of a predicated thing is by some of its predicates or the relationships it is part of. A composite thing, such as a relationship or a set of entities or relationships, can be characterised by its components (as well as by the predicate called 'has-element'). In the case of a composite and predicated thing (predicated also by other predicates than the one called 'has-element'), both kinds of characterisations are possible. Furthermore, simple characterisations can be combined to form more complex ones by logical junctions and/or quantifications. We dispense here with further details and refer to the relevant literature, e.g. on propositional and predicate logic.

Related to the notion of type are the notions of population and instance.

Definition E6: Type¹, Population, Instance

A **type** of things is a specific characterisation (e.g. a predicate) applying to all things of that type. •

A **population** of a type of things is a set of things, each one fulfilling the characterisation determining that type. •

An **instance** of a type of things is an element of a population of that type. •

Note:

These notions apply to predicated things as well as to composite things.

Examples of characterisations:

- is a person (the simplest case of a characterisation, i.e. one predicate).
- is the author of Philosophical Investigations (a simple characterisation, being part of a binary relationship).
- is the author of Philosophical Investigations OR is the author of Meaning and Necessity (a characterisation, being an or-junction of two simple characterisations).

Examples of types:

- The type of things fulfilling the characterisation 'is a person'.
- The type of things fulfilling the characterisation 'is the author of (any book)'.
- The type of things fulfilling the characterisation 'is the author of Philosophical Investigations'.
- The type of relationships containing the predicates 'is the author of' and 'is written by'.

Examples of populations:

- The set of all persons under consideration.
- The set of all book authors under consideration.
- The set of things characterised by the predicate 'is the author of Philosophical Investigations' ({Ludwig Wittgenstein}).

We have now all the ingredients to define the dynamic vs. the static aspects of conceived domains. Let us begin with the most fundamental concepts in that respect, the concept pair "transition" and "state".

Definition E7: Transition², State, Pre-state, Post-state

A **transition** is a special binary relationship between two (partially or totally) different composite things, called the pre-state and the post-state of that transition, whereby at least one thing is element of the pre-state, but not of the post-state, or vice versa. •

A **state** is a composite thing, involved as pre-state or as post-state in some transition. No

¹**Synonym:** 'Category'

²**Warning:** This term is sometimes used in different meanings in the literature!

element of a state may be a transition, itself. •

The **pre-state** of a transition is the state valid before that transition, and is characterised by the special predicator ‘before’. •

The **post-state** of a transition is the state valid after that transition, and is characterised by the special predicator ‘after’. •

Notes:

- The pre-state and the post-state of a transition differ in practice usually not with respect to the involved predicated things, but with respect to the relationships where these predicated things are contained in.
- For denoting transitions, we will henceforth use the shorthand notation:
transition: pre-state => post-state

Examples:

- $s_1 = \{\text{student-not-enrolled-for-course, ...}\}$, $s_2 = \{\text{student-taking-course, ...}\}$,
 $s_3 = \{\text{student-having-completed-course, ...}\}$ are three different states;
enrolment: $t_1: s_1 \Rightarrow s_2$ and completion: $t_2: s_2 \Rightarrow s_3$ are transitions;
note that the unspecified state elements (...) need not have changed.
- $s_1 = \{\text{blackboard-clean}\}$, $s_2 = \{\text{blackboard-written-on}\}$ are two different states;
 $s_1 \Rightarrow s_2$ and $t_2: s_2 \Rightarrow s_1$ are transitions,
whereby the post-state of the second transition is identical to the pre-state of the first transition.

Transitions can be related to each other, to form "state-transition structures".

Definition E8: State-transition structure¹

Given are the transitions $t_x: s_1 \Rightarrow s_2$ and $t_y: s_3 \Rightarrow s_4$. The following basic **state-transition structures** exist in this case:

(1) Sequence:

$\text{sequ}(t_x, t_y)$ is a sequence of transitions if s_3 is a subset of s_2 .

The resulting state-transition structure has s_1 as pre-state and s_4 as post-state.

Longer sequences are defined as follows:

$\text{sequ}(t_x, t_y, t_z)$ follows from $\text{sequ}(t_x, t_y)$ and $\text{sequ}(t_y, t_z)$

(2) Choice:

$\text{choice}(t_x, t_y)$ is a choice of transitions if the intersection of s_1 and s_3 is not empty.

The result is either transition t_x or t_y , but not both.

(3) Concurrency:

$\text{concur}(t_x, t_y)$ are concurrent transitions if the intersection of s_1 and s_3 is empty.

The result is $(s_1 \text{ union } s_3) \Rightarrow (s_2 \text{ union } s_4)$. •

Because of the choice construction there exist state-transition structures which have a non-deterministic post-state and thus are not transitions. This may be useful for a prescriptive description of the behaviour of a set of things. However, if we restrict ourselves to ‘backward-reasoning’, i.e. describing the behaviour of a set of things in the past, it is sufficient to restrict ourselves to state-transition structures with a unique pre-state and unique post-state. Those state-transition structures can be viewed as transitions themselves, called composite transitions. They have figuratively speaking a "coarser granularity" than their component transitions.

¹Synonym: ‘Process’

Definition E9: Composite transition

A **composite transition** is a state-transition structure with a unique pre-state and a unique post-state. •

Note that by the recursive definition of a state-transition structure, very complex transitions can be formed.

From an initial state of a state-transition structure, a well-defined number of valid states are reachable by valid transitions within that structure.

Our notion of transition implies neither the notion of *time* nor the notion of *actual* changes. It covers only the *potential* changes of conceived domains. A domain is conceived to potentially move from one state to another and back again, i.e. a transition may be reversed and repeated.

It is therefore important to distinguish transitions from their "occurrences". There may be more than one transition occurrence belonging to one and the same transition. For example, a blackboard (see the above examples of transitions) may actually change more than once from the status 'clean' to the status 'written-on', and vice versa. These transition occurrences belong to the transition of the blackboard from 'clean' to 'written-on', or vice versa, respectively. An actual transition occurrence, must be considered unique and cannot be repeated or reversed.

Definition E10: Transition occurrence¹

A **transition occurrence** is a specific occurrence of a transition. A set of transition occurrences is subject to strict partial ordering. •

Notes:

- For denoting transition occurrences, we will use the shorthand notation:
occ(transition): pre-state -> post-state
- A more formal way of expressing this is as follows: Given two transition occurrences, where the pre-state of the second one is identical to the post-state of the first one, say:

occ(t_b): $s_1 \rightarrow s_2$ and occ(t_a): $s_2 \rightarrow s_3$,

we declare them to be subject to strict partial ordering:

occ(t_b) < occ(t_a).

(pronounce: occ(t_b) occurs before occ(t_a); alternatively: occ(t_a) occurs after occ(t_b)).

The meaning of strict partial ordering is that, given three transition occurrences occ(t_1), occ(t_2) and occ(t_3), the following characteristics apply:

- if occ(t_1) < occ(t_2) and occ(t_2) < occ(t_3) then occ(t_1) < occ(t_3) (transitive)
- if occ(t_1) < occ(t_2), then occ(t_2) not < occ(t_1) (asymmetric)
- occ_i(t_j) not < occ_i(t_j) for every occurrence i of transition j (irreflexive)

Usually it is said that a transition occurs at a certain point in time. In fact, through the special predicators 'before' and 'after', relative time is introduced. One could introduce absolute time, e.g. in the form of time points on a time axis, as a fundamental concept. An alternative is to derive the concept of time points on a time axis from a sequence of transitions of a special entity, a "clock". The choice depends on the philosophical question whether time implies change or the other way around. For our framework we have chosen to derive a time axis

¹ **Warning:** The term 'Event' is sometimes used in this meaning and sometimes in various other meanings in the literature!

consisting of time points from the strict order of transition occurrences of a clock.

Definition E11: Relative time, Absolute time

The strict partial order imposed on the sets of all transition occurrences is called **relative time**. •

Absolute time may be determined by a clock that issues (assumedly) regular pulses (transition occurrences of the clock, or clock events). An absolute time value may be assigned to some transition occurrence, by comparing that transition occurrence with the successive (absolute-time-determining) clock events. A strictly ordered set of time points can be defined on the basis of these clock events, called time axis. •

Finally, we need the notion of "rule".

Definition E12: Rule¹

A **rule** determines a set of permissible states and transitions in a specific context. In other terms, a rule governs a non-empty set of types of things by determining their permissible populations. •

Notes:

- We use here the term 'rule' in the most generic sense, covering the widest possible spectrum, from the presumed rules governing the functioning of the "physical world" (very strong "laws of nature"), via rules of agreed behaviour in society (e.g. not always enforceable legal arrangements and business guidelines), to semantically or syntactically permissible and representations in some language (legal expressions). Rules are more or less strict or enforceable.
- Instead of specifying what is permissible, it may be easier in practice to formulate rules specifying what is forbidden, i.e. what the constraints are.
- In the above definitions, as well as in the definitions still to come, a few, but certainly not all possible generic constraints are mentioned. While we could introduce more constraints concerning things on a generic level, it is obvious that we would hardly be able to capture all possible constraints applying to any specific domains.

Examples:

The law of gravity; the laws of electromagnetism; the logical axioms that restrict statements to remain meaningful (e.g. true and false being mutually exclusive, a person being either active or inactive); mutually agreed rules of what may be entered into or changed inside a database, or passed over a network (semantic integrity, protocols, etc.).

¹**Note:** Sometimes, the term 'Law' is used instead [e.g. Bun77, Bun79].

3.3 Actors, Actions and Actands

With the fundamental concepts in section 3.2, we are in principle able to specify any conceptions of domains. However, for the purpose of treating information and communication in general, and within organisations in particular, it is useful to introduce a few further specialisations of the fundamental concepts. In particular, it is useful to consider explicitly what "causes" transitions (i.e. "actions" performed by "actors"), and the "goals" of actions.

"Action" by itself, may be considered as a kind of transition, one in which an "actor" is involved, who brings about (or causes) the transition, with possible involvement of consumable and/or non-consumable passive components, called "actands". Otherwise the same assertions as for transitions are valid. The equivalent of a composite transition is called a "composite action", and that of a transition occurrence is called an "action occurrence". If several actors are performing one action together in a co-ordinated way for trying to achieve a common goal, this action will be called "co-action".

Definition E13: Actor

An **actor** is a special thing conceived as being "responsible" or "responsive" and as being able to "cause" transitions, and is therefore part of their pre-states, and, if not "destroyed" or "consumed" by the transitions, also part of their post-states. •

Notes:

- One actor can in general cause more than one transition, i.e. perform more than one action.
- Actors can be classified in many different ways. For our purposes, a useful classification is to distinguish between actors having the normal capabilities of human beings, which we will call human actors, and all other actors, such as other living beings or devices.

Definition E14: Action¹, Composite action, Action occurrence, Co-action²

An **action** is a transition involving a non-empty set of actors in its pre-state, and, if not "destroyed" or "consumed" by the action, in its post-state as well, and involving a non-empty or empty set of other things (actands) as part of its pre-state, and having a non-empty or empty set of other things (actands) in its post-state. •

Examples:

- Stock-taking (action) by a warehouse-clerk (actor) checking current stock, and producing a stock-inventory;
- Issuing a stock item (action) by a stock supervisor (actor), resulting in a change of stock level;
- Writing (action) a report by an author (actor);
- Expressing (action) a conception by a person (actor), in the form of a the representation.

A **composite action** is a composite transition with the same conditions as applying for the notion of action. •

An **action occurrence** is a transition occurrence with the same conditions as applying for the notion of action. •

A **co-action** is a special action performed by at least two actors in a co-ordinated way,

¹ **Alternative terms:** 'Activity', 'Act'

² **Alternative term:** 'Co-ordinated action'

pursuing a common goal. •

Note:

A co-action may be viewed as a *composite* action or as a *simple* (non-composite) one, depending on the desired level of "granularity".

Example:

Two people are together lifting and shifting a heavy piece of furniture, with the goal to move it from one place to another.

Definition E15: Actand¹, Input actand, Output actand, Resource

An **actand** is a thing involved in the pre-state or post-state of an action, not considered as an actor for that action. •

Note:

An actand can, however, be functioning as an actor in another action, that is, the intersection of the set of all actors and the set of all actands may be non-empty in a specific context.

An **input actand** is a part of the pre-state of an action, excluding the actors. •

An **output actand** is a part of the post-state of an action, excluding the actors. •

The pre-state of an action, i.e. the union of the set of actors and the set of input actands of that action, is called its **resources**. •

Examples:

- Stock-taking (action) by a warehouse-clerk (actor) checking current stock (input actand), and producing a stock-inventory (output actand);
- Issuing a stock item (action) by a stock supervisor (actor), resulting in a change of stock level (input as well as output actand);
- Writing (action) a report (output actand) by an author (actor);
- Expressing (action) a conception (input actand) by a person (actor), in the form of a the representation (output actand).

Notes:

- Actands may be consumed (e.g. raw material, money), generated (e.g. end product, money), or used (e.g. machines, data) by actors through actions.
- An action must have at least one input actand or one output actand, i.e. the union of the set of input actands and the set of output actands of an action must be non-empty.
- The output actands of an action may depend on the context or situation in which that action is performed. We therefore introduce another concept, the "action context".

Definition E16: Action context

The **action context** of an action is a special, optional part of the pre-state of that action, qualifying the context or situation in which that action is performed, and determining or modifying at least one of its output actands. •

Example:

A pupil is supposed to perform the action of solving a mathematical task. In a school situation, in front

¹ **Note:** This is an artificial term (constructed in analogy to e.g. 'Operand').

Alternative term: 'Object of action'

of his teacher, say, he fails, i.e. there is no output actand at all. In a home situation on the other hand, in front of his parents, he solves the task properly.

By defining action as a transition, one describes what happens, but leaves out why that is the case. Especially in the organisational context that we ultimately wish to describe, the notion of purposefulness of action requires that we find a way of linking the notion of action with that of "goal". We consider any goal to be defined *a priori*, i.e. it must be part of the pre-state of the action in question. If there is a goal defined for an action, the actors of that action are said to be "goal-pursuing". Only if an action occurrence is successful, the goal specified (intensionally) as an input actand is equivalent to the post-state of that action occurrence.

Definition E17: Goal, Goal-pursuing actor

The **goal** of an action is a special input actand of that action, pursued by the actors of that action and stating the desired output state intensionally. •

A **goal-pursuing actor** is an actor performing an action, who deliberately aims at a specific goal when involved in that action. •

Note:

This definition covers both the case of a human actor consciously attempting to achieve the intended outcome and a device which is programmed or otherwise triggered to perform a task which has been set up in a purposeful way.

The following, only roughly sketched sample case may illustrate most of the concepts we have introduced so far. A full and more precise description would obviously require much more detail.

Concepts:	Some examples of instances of concepts:
Domain	A road transport enterprise
Predicator	is-truck, is-truck-driver, is-bill-of-lading ...
Predicated thing	staff members, truck drivers, planners, trucks, goods (to be transported), employment terms, the enterprise itself, customers, the market ...
Relationship	data of personnel, qualities/quantities of goods, bills of lading, transportation of cargo to destination ...
Entity	staff members, truck drivers, planners, trucks ...
Type	the notion of staff member ...
Population	all staff members, all truck drivers ...
Instance	one particular staff member, ...
Transition	conclusion of contract, delivery of goods, receipt of payment ...
State	warehouse contents, truck load, delivery-plan data, orders ...
Pre-state	cargo before delivery, schedule before execution ...
Post-state	delivered cargo, executed schedule ...
Transition occurrence	specific occurrence of one of the transitions mentioned ...
Relative time	planning stage of day, start of day, delivery sequence ...
Absolute time	today, a particular time of today ...
Rule	social security regulation, truck maintenance date, civil law, the highway code, transport licensing rules, mechanical and motor vehicle engineering principles, contracts (when concluded) ...
Actor	truck drivers, planners, trucks, fax machines, telephones, computers ...
Action	placing orders, receiving orders, agreeing contract, transporting goods, exchanging messages, adjusting schedules ...
Composite action	delivery run ...
Action occurrence	placing of specific order ...

Co-action	truck driver and truck transporting goods ...
Actand	customer, cargo, delivery plan, truck-maintenance plan ...
Input actand	orders (for delivery schedule), delivery schedule (for delivery run) ...
Output actand	goods (from delivery), truck (from maintenance) ...
Action context	the enterprise and its transport ordering customers ...
Goal	delivery schedule, contract terms, truck-maintenance plan ...
Goal-pursuing actor	truck drivers, planners (in connection with delivery schedules) ...

We shall see that many further fundamental concepts one might wish to introduce may be expressed in terms of the above.

3.4 Constructing and Representing Conceptions

One of the issues our constructivist approach concentrates on is how one can represent one's conceptions of some domains. Our field of interest, information systems, is concerned with precisely the same objective - representing conceptions of some organisational domain so as to provide a basis for decision making and control. In our case, such representations must be acceptable to more than one person. How do we arrive at an agreed position and how can we express ourselves in that connection?

Examples of components in the domain of stock analysis are the "stock on hand" and the "stock level". Since the stock analyst might not personally inspect the warehouse shelves, the conclusion that there is an "out of stock" condition would be drawn on the basis of a message regarding the latest stock level. As a result of earlier understandings, the perception and conception of that message triggers the interpretation that a re-order notice is to be issued. All the persons involved in this action constitute a larger domain, which may be called "stock control domain".

The earlier discussion of things in connection with their conceived, agreed or assumed "existence" is of importance, because the thinking and communication processes involved constitute the foundation of societal agreement. While accepted (or at least recognised) by a group of persons, any common understanding is based upon subjective individual mental processes. Exchange of representations of these, and negotiation regarding their meaning lead to an "inter-subjective" (mutually agreed) view of "reality" or abstraction thereof. We assume, therefore, that we can henceforth reason about these things, even though what is available to us (personally) is only what is in our minds. The mental processes as such will be considered later. Meanwhile, accepting our constructivist view, it is no longer necessary to argue about any inherent "reality" of what constitutes, or rather, might be behind such conceptions.

We still need to consider the character and form of what is communicated. What actually is communicated are representations of our conceptions. Later, we shall see that this involves a number of special actions, such as constructing conceptions, forming and uttering representations thereof, and interpreting representations.

Meanwhile, we must revisit our context for using conceptions and representations - the "organisation" we participate in and its "environment", i.e. the society. The term "society" as used here, is a collection of individuals, groupings and institutions in some coherent part of the "social world". It is joined by a common language, cultural traditions and similar links. In order to be capable of coherence a mechanism must exist about which we assume:

Assumption [h]:

For a society, it is of prime interest that any domain is represented such that agreement about it emerges. Without claim to presumed "existence" in any "real-world" sense of the term, the resulting representations everyone agrees with, will be called "inter-subjective reality".

We must now analyse the domain of that "inter-subjective reality" in more detail. The first aspect to be considered is the assumed process of how we go about viewing it, and the way in which we arrive at what we accept as valid.

How do we actually deal with our perceptions and the resulting conceptions? Our constructivist view accepts that the conceptions in the mind of a person result from "observation, experience or inner reflection" (Assumption [c]). These actions are usefully grouped into those of getting various patterns into the mind via the senses ("perceiving") and those that lead to the end result ("conceiving"). Having achieved a conception, one may then construct an external representation ("representing", see Assumption [g]) as a basis for communication. These actions were illustrated already in figure 3.1-3.

The establishment of inter-subjectivity requires an exchange of messages. A domain or a part thereof is perceived as a pattern, internally leading to a conception and represented externally by a representation of that conception. It is that representation that may be "communicated". In other words, one person has replaced the original "part of the world" to something that is suitable for communicating about it vis-a-vis some other person.

Now, if someone's representation, in turn, is perceived and processed by some other person along the same lines, the message exchange may lead to stable conceptions for all parties involved. The final external representation of one and the same conception in many peoples' minds may then be equated to its "inter-subjective reality". Because we adhere to the constructivist approach we need not be concerned with the detailed nature of the mental aspects involved. We merely assume that domains can be inter-subjectively represented by a mapping to commonly agreed representations.

Perceiving, conceiving and representing are special actions. Similarly, domain, perception, conception and representation are special actands. They will all be defined more explicitly at a later stage. For the moment, we shall restrict ourselves to the following recapitulation.

Assumption [i]:

Stable conceptions may be formed in a person's mind - as a result of observing the state of some part of the conceived world or changes of such a state, or as a result of communication with other persons regarding it. Any such conception may be called knowledge about the world. The assumed existence of the (mental) conception is a metaphor for the experience of persons when entertaining (hopefully) meaningful communication.

Let us now define the concepts relevant in the current context, domain, human actor, perception, conception, representation, and the various actions involved, in terms of the concepts we defined in the previous sections 3.2 and 3.3. Note that all the above concepts (thing etc.: E1 to E17) were introduced as conceptions, in line with our constructivist position. Thus, when we are going to explain conceptions and related issues in terms of the above concepts, we are in fact explaining conceptions in terms of conceptions. This is, however, not

a genuine difficulty as it might seem at the first glance, provided we are careful enough about the level of discourse.

Definition E18: Domain¹, Domain component², Domain environment

A **domain** comprises any "part" or "aspect" of the "world" under consideration. •

Examples:

- An enterprise (viewed as part of society) composed of departments.
- A department (of that enterprise) composed of employees.
- The recently engaged employees (a domain usually overlapping a number of departments).

A **domain component** is any "part" or "aspect" of that domain. •

Examples:

- An employee of a department.
- An employee out of the set of the recently engaged employees.
- A department of an enterprise.

A **domain environment** is the "world" without that domain. •

Note:

Neither a domain, nor any of its components, nor its environment can be determined a priori. They can be determined only a posteriori, after perceiving and conceiving that domain.

Definition E19: Human actor, Perception³, Perceiving action, Perceiver

A **human actor** is a responsible actor with the capabilities and liabilities of a normal human being, in particular capable of performing perceiving actions, conceiving actions and representing actions ⁴. •

A **perception** is a special actand resulting from an action whereby a human actor observes a domain with his senses, and forms a specific (static, non-time-varying, or dynamic, time-varying) pattern of visual, auditory or other sensations of it in his mind. •

A **perceiving action** is a special action of a human actor having a domain as input actand and a perception as output actand. •

A **perceiver** is a human actor involved in a perceiving action. •

¹ **Warning:** The term 'Domain' as well as the terms 'Field', 'Area' and 'Universe of discourse' are sometimes used in this meaning and sometimes in various other meanings in the literature!

² **Synonym:** 'Domain element'

³ **Warning:** The term 'Perception' is sometimes used in this meaning and sometimes for what we call 'Conception'!

⁴ See definition E23.

Definition E20: Conception¹, Conceiving action, Conceiver, Conceiving context

A **conception** is a special actand resulting from an action whereby a human actor aims at interpreting a perception in his mind, possibly in a specific action context. •

Notes:

- Hence, a conception, as used in further analyses, is treated as a specialisation of actand (being itself a specialisation of thing). To the constructivist, this does not mean that it "exists", only that in so far as one can talk about it, the specific description for doing so is accepted, as such.
- The interpretation of a perception may be trivial, resulting in a conception identical or similar to that perception, but usually it is non-trivial.

A **conceiving action** is a special action of a human actor having a perception and possibly some action context as input actand(s) and a conception as output actand. •

A **conceiver** is a human actor involved in a conceiving action. •

A **conceiving context** is the action context of a conceiving action. •

Example:

In figure 3.1-2, one possible conception is "wineglass", another one is "two faces". Which interpretation of that pattern is chosen may depend on the context or situation.

For the sake of convenience, we finally define a special composite action, an interpreting action, the corresponding actor, an interpreter, and an interpreting context:

Definition E21: Interpreting action², Interpreter, Interpreting context

An **interpreting action** is the sequence of a perceiving action performed on a domain, resulting in a perception of that domain, followed by a conceiving action performed on that perception, resulting in a conception. •

An **interpreter** is a human actor performing an interpreting action. •

An **interpreting context** is the action context of an interpreting action. •

For representing conceptions, we need a "language". We will use the term "language" only in the sense of "symbolic language". A (symbolic) language allows to express "sentences" or, more general, "symbolic constructs". The set of "symbols" used for any representations in a language is called its "alphabet".

Definition E22: Symbol, Alphabet, Symbolic construct³, Language

A **symbol** is a special entity used as an undividable element of a representation⁴ in a language. •

An **alphabet** of a language is a non-empty and finite set of symbols. •

A **symbolic construct** is a non-empty and finite "arrangement" of symbols taken from an

¹ **Warning:** The term 'Perception' is sometimes used instead [e.g. Rus76]!

² **Synonym:** 'Interpreting process'

³ **Alternative term:** 'Symbolic form'

⁴ See Definition E23.

alphabet. In the one-dimensional case, an arrangement is just a sequence of symbols (a "sentence"). In the n-dimensional case ($n > 1$), it may be any arrangement of its constituting symbols in the n-dimensional space. Provided one considers the elements of arrangement (such as sequence) belonging to the alphabet, a symbolic construct is a non-empty and finite set of symbols. •

A **language** is a non-empty set of permissible symbolic constructs. The permissible symbolic constructs in a language are determined either extensionally by enumeration or intensionally by a set of rules. The rules of a language may be syntactic ("grammar") as well as semantic ("semantic rules"). •

Notes:

- These definitions hold for any kind of symbolic language of any dimension.
- Languages can be classified according to many criteria, for example, as lexical or graphical, as (more or less) formal or informal, or as (more or less) universal or specific-purpose-oriented.

Examples:

- natural language (mostly leading to informal expression only)
- the language of mathematical logic
- graphical languages (may be used loosely or rigorously)
- icons and other pictorial means

Definition E23: Representation¹, Representing action, Representer, Representing context

A **representation** is a special actand describing some conception(s) in a language, resulting from an action whereby a human actor aims at describing his conception(s), possibly in a specific action context. •

Example:

The entity called 'Ludwig Wittgenstein' is a conception. A representation of it is the string of characters 'Ludwig Wittgenstein'. Another one is the expression 'The author of Philosophical Investigations'.

A **representing action** is a special action of a human actor having a conception and possibly some action context as input actand(s) and a representation as output actand. •

A **representer** is a human actor involved in a representing action. •

A **representing context** is the action context of a representing action. •

To be able to talk a bit more thoroughly about the action of representing conceptions, we have to introduce the concepts "label" and "reference":

¹ **Alternative terms:** 'Sign token', 'Sign carrier', 'Signifiant', 'Representamen'

Note: Representations can be short and simple (as in the above examples), or large and complex. The terms 'Sign token', 'Sign carrier' etc. are often used for short representations only. We prefer the term 'Representation' because of its more general connotation.

Warning: The term 'Sign' is sometimes used as meaning 'Representation', 'Sign token', etc., and sometimes as meaning a quaterary 'Sign relationship' between a domain, a conception, a representation and an interpreter/representer (or a ternary or binary projection thereof) (see below).

Definition E24: Label¹, Reference²

A **label** is a special entity being an elementary representation and used for referring to some conception in an elementary way. •

Note:

We can distinguish between various kinds of labels, in particular between labels used for referring to entities (entity labels), and labels used for referring to predicates (predicator labels).

Examples:

- The string of characters 'Ludwig Wittgenstein' (an entity label)
- The string of characters 'is a person' (a predicator label)

A **reference** is a special binary relationship between a conception and a representation used to refer to that conception. •

Examples:

- Reference between an entity and a label:
A specific entity is referred to by the entity label 'Ludwig Wittgenstein'.
- Reference between an entity and an expression (composed of several labels):
A specific entity is referred to by the expression 'The author of Philosophical Investigations'.
- Reference between a relationship and a sentence:
A specific binary relationship is referred to by the sentence 'Ludwig Wittgenstein is the author of Philosophical Investigations',
or by the sentence 'Philosophical Investigations is written by Ludwig Wittgenstein'.

Notes:

- A reference may be just a private matter of an individual person, or commonly used by several or a large number of persons, established inter-subjectively by means of communication among these persons. We call the latter ones common references. The above examples certainly belong to the latter ones.
- A conception may be represented by zero, one or several representation(s). A representation may refer to zero, one or several conception(s).
- References may be simple or complex. The simplest case of a reference is the relationship between an elementary thing and a label referring to it. These simple references are important in so far as they are used within more complex references. As in the general case, an elementary thing may be represented by zero, one or several labels, the latter being a synonym situation. A label may refer to zero, one or several elementary things, the latter being a homonym situation. Thus, a label does not necessarily identify an elementary thing.
- In general communication, and in an organisational context in particular, any homonym situation is undesirable, because it usually hampers or even causes failure of the communication. Homonym situations should therefore be avoided in the first place or clarified appropriately. A synonym situation is not so dangerous, but may cause deterioration of the communication as well, due to the unnecessarily large number of labels used. If synonyms exist, it is good practice to declare one of them as

¹ **Synonym:** 'Elementary sign token'

Warning: The term 'Name' is sometimes used in this meaning and sometimes in various other meanings in the literature!

² **Note:** A 'Reference' could be also called a (binary) 'Sign relationship'.

the preferred synonym.

Examples:

- The label ‘Queen Elisabeth’ does certainly not refer to one entity only. It refers to at least two persons, to at least one ship, and may constitute a reference to other entities as well.
- Whether or not the label ‘Ludwig Wittgenstein’ refers to one entity only (in the minds of people around the world), is unknown to us. Suppose there are several persons called ‘Ludwig Wittgenstein’ the expression ‘Ludwig Wittgenstein, the philosopher who wrote the book Philosophical Investigations’ will probably clarify that homonym situation.

Note:

The references used may exist for a long time already or may have to be newly established by the representing action. The creation of new references is usually called a baptising action and may be considered as a part of a representing action.

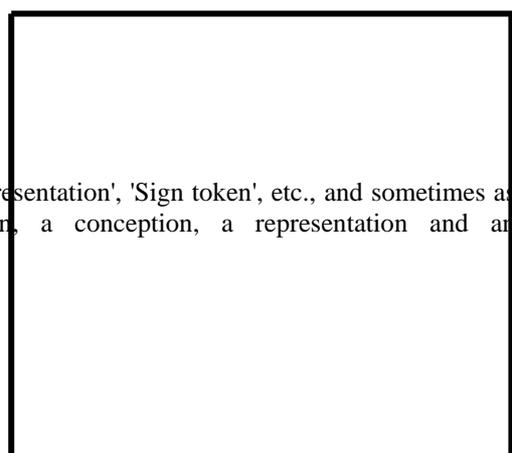
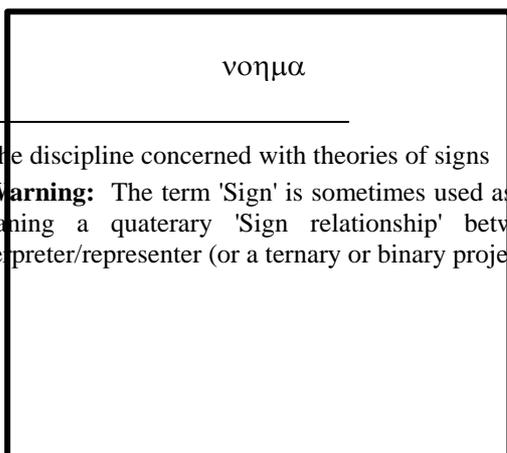
Examples of baptising actions:

- Baptising a child (the label usually exists already, only the reference is established);
- Inventing a new entity label for referring to a new or an already existing entity;
- Using an existing entity label for a new entity (usually resulting in a homonym situation);

We may repeat that the established terminology does not express what the "world" is - in the sense of any "existence" - or how it is made up. But when we talk about it, it will be in these terms. The question might be raised whether the concept of a conception is needed at all. One obviously cannot test it directly, because it is in the practically inaccessible world of a person's mind. What one does have access to are the statements (some representation) a person makes about it. This situation is exactly the same as that of the physicist who tries to understand some phenomenon of the "physical reality" by means of a theory. The theory explaining it - on the basis of which predictions may be made and tests conducted to corroborate (or falsify) it - is also a conceived abstraction of the phenomenon's characterisation. Thus, *a physicist's theory is merely a special case of a conception* (represented e.g. as a bunch of mathematical formulae).

By now we have established that representations are produced from perceptions and conceptions. It may be worth our while to look briefly at the historical development of the ideas, which relate closely to the discipline of semiotics¹ from which we have borrowed a number of ideas.

As mentioned before, another name for the symbols and symbolic constructs used for representing conceptions is "sign" ². There is an extensive tradition, dating back to ancient Greek philosophy (e.g. Aristotle, 384-322 BC), of theorising about the nature of signs, their meanings and their relation to things one observes. A traditional way of distinguishing and illustrating different categories and their relationships is to use the so-called "semiotic" or "representational" triangle, also known as Ogden's representational triangle (see figure 3.4-1).



¹ The discipline concerned with theories of signs

² **Warning:** The term 'Sign' is sometimes used as meaning 'Representation', 'Sign token', etc., and sometimes as meaning a quaterary 'Sign relationship' between a domain, a conception, a representation and an interpreter/representer (or a ternary or binary projection thereof).

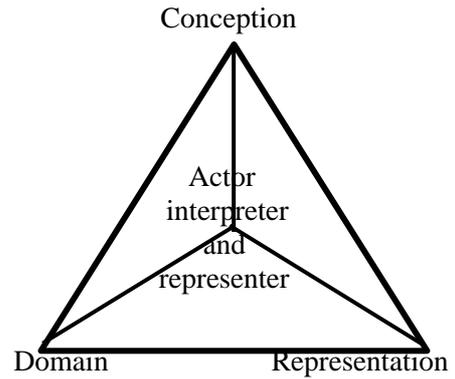
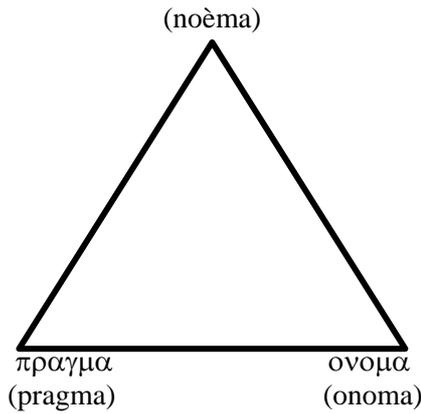


Figure 3.4-1: Classical or Ogden's representational triangle

Figure 3.4-2: The FRISCO semiotic tetrahedron

The classical triangle relates an "object" (πραγμα, "pragma"), i.e. some domain, a "concept" (νοῆμα, "noèma"), i.e. the thought or conception that stands for that domain, and its "name" (ονομα, "onoma"), i.e. the label by which one refers to it.

Later, C.S. Peirce (1839-1914) [HW31, Moo+82, Mor45] stressed the important role the "interpreting body" or "interpretant" is playing in the conceptualisation process. He emphasised that there are at least three aspects:

- some physical representation (the "representamen")
- something to which this refers or alludes (the "sign object")
- somebody able to interpret a sign object and to construct a representamen (the "interpreter")
- some mental effort produced by a sign object on the interpreter (the the "interpretant")

Hence, Peirce extended the three classical (Aristotle/Ogden) categories by an additional "interpreter". He rephrased the classical definition of the sign concept ("aliquid stat pro aliquo") to :

Sign = something which stands to somebody for something in some respect or capacity.

This view may best be expressed by a tetrahedron, which is shown in Figure 3.4-2 as a projection in the plane of the classical triangle. In fact, the centre stands for some actor who either represents something to someone else or, alternatively, interprets what resulted from perception (observation) of some domain.

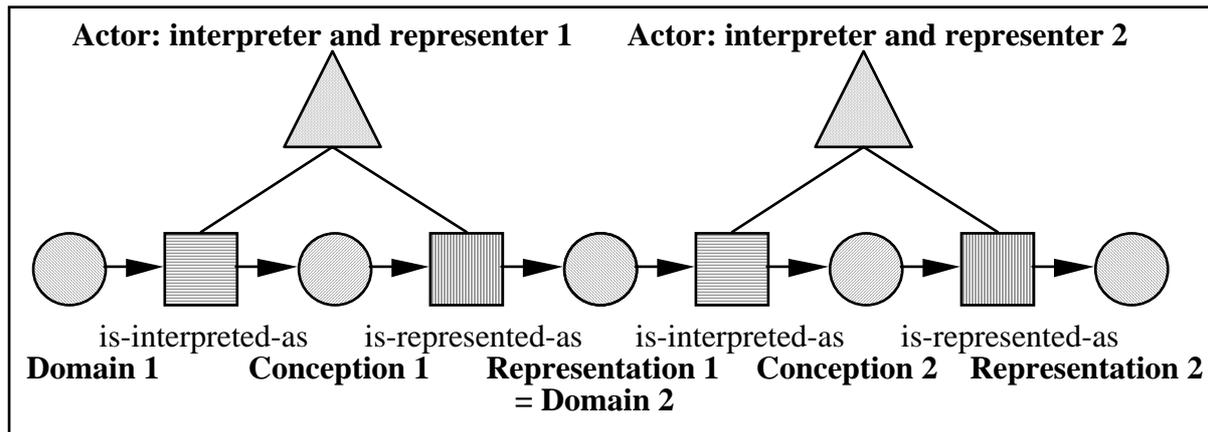
Note that our conceptual framework concerning domains, conceptions, representations, and interpreters and representers can be mapped easily to the views of Aristotle and Peirce. However, for our purpose, a different terminology is preferred. The mapping between the terms used is as follows:

Aristotle's terms	Peirce's terms	Our preferred terms
Pragma	sign object	domain
Noema	-	conception
Onoma	representamen	representation
-	interpretant	interpreter and representer

The semiotic triangles help illustrating the two philosophical positions mentioned in the beginning of this chapter 3:

3.5 Communicating Conceptions: Restrictions Imposed by the Constructivist View

We may utilise an extension of figure 3.4-3 to describe the actions involved when two human actors communicate. The first human actor utters a representation (say, speaks a sentence, writes a note, etc.). That representation will then serve as a domain for the second human actor, who interpretes it, that is to say, constructs a personal conception for subsequent use. The second human actor utters some representation on his own accord, which may then be interpreted by the first human actor, and so on (see figure 3.5-1).



Legend:

- Triangles: Actors
- Rectangles: Actions
- Circles: Actands
- Lines: Actors perform actions
- Arrows: Actands are input or output of actions

Figure 3.5-1: Communication between two human actors

The main thrust of this argument is that the two persons and any representations are in the "physical, hence easily accessible part of the world", whereas the conceptions are in the "mental, hardly accessible world". From the objectivist point of view, the physical exchange between the persons is well understood and one ignores any discrepancy between the two conceptions. The constructivist position is that this lack of manifest agreement regarding the conceptions in fact means that the entire schema remains speculative. Whilst gradual convergence of the representation-domain pairs that are exchanged will be interpreted as a stabilisation of knowledge, one merely assumes that the agreement is meaningful.

The actions and interactions depicted in figure 3.5-1 are a first step in describing some more explicit detail of what was called "communication" in assumption [g]. It is evident that this refers to a complex process. A fuller discussion is deferred until section 3.8. In anticipation, however, we may already conjecture that the result will be a change in each participant's conceptions. Now, if the individual conceptions have actually become stable they may well have converged to a situation where they may be said jointly to constitute a "shared" set of conceptions. It was in that sense that we speak of meaningful "agreement".

In traditional semiotics, one distinguishes syntactics, semantics and pragmatics. These deal with the structures, meanings and usage of representations, respectively. There is furthermore a need for considering other aspects, the physical properties of representations. Closely

related, but more abstract in nature, are the empirical characteristics of representations, such as the pattern of symbols, the "entropy" and so on.

All these inhabit a world in which persons interact, agreeing or disagreeing about meanings, expressing beliefs, entering into commitments, etc. In other words, there is also a social angle to the use of representations. In order to understand which aspect or aspects one discusses it is important to distinguish them as applying at different semiotic levels.

Assumption [j]:

Persons view, experience and discuss representations at several semiotic levels, that is to say, they address, with varying emphasis: the physical and empirical aspects of a representation, the laws for expressing such a representation, its meaning, its use and effects, and the extent to which one can agree on and possibly derive commitments from whatever a statement about some thing might entail.

Definition E25: Semiotic level

The **semiotic level** of a representation is the aspect considered in representing it. The semiotic levels are: physical, empirical, syntactical, semantical, pragmatic, and social. •

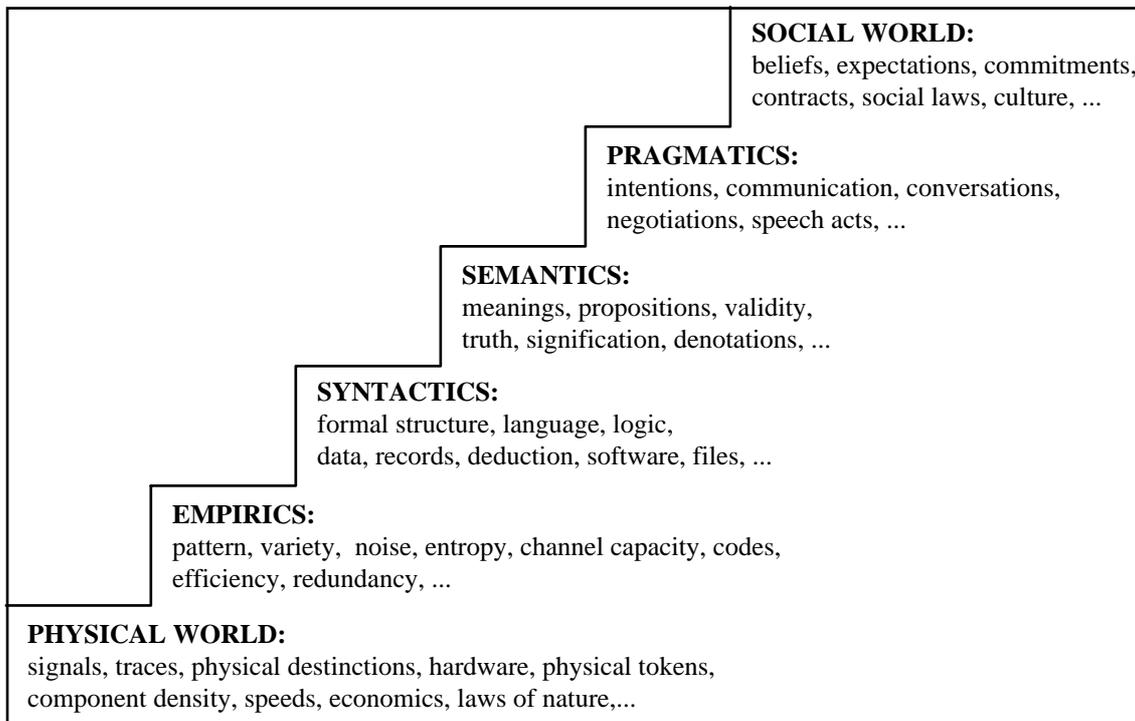


Figure 3.5-2: The "semiotic ladder" between the "physical" and the "social world"

The six semiotic levels that are of interest in the context of information systems, are listed in figure 3.5-2, in a format which is sometimes called the "semiotic ladder" [Sta73, Sta96a]. Alongside, a number of illustrative words are shown, which are examples of aspects of representations such as treated at the levels in question.

A specific example of something that might be considered at different levels is an application software development contract. The written document would be at the physical level, the textual pattern is an empirical matter, the legal/business language in which it is drawn up is at the syntactic level, its thrust is at the semantic level, it has resulted from negotiations at the pragmatic level, and constitutes a number of accepted commitments at the social level. If e-mail or facsimile messages have been exchanged in the course of contracting, similar aspects would be in the forefront: physical message bits, the syntax of the

network protocol, the significance of the message to sender and recipient, the pragmatic extent to which action would follow receipt, the legal validity of what the exchanging parties agree. And the software contracted to be developed has similar aspects: socially - its expected usability, pragmatically - its expected usefulness, semantically - its functionality, syntactically - its correctness and completeness, physically - its release diskettes and documentation.

All six semiotic levels are of importance in talking about information systems, as well as when actually designing and using them. The common confusion as to what various terms mean is due to the fact that one is often unaware of the level at which assertions apply, in particular in situations in which more than one level is involved.

Our definition of the notion of language is very general. It encompasses the traditional aspect of a syntax (an "alphabet" or set of accepted "words", combined with a set of rules which determine how one may use these), and implies the intention that it will be used to express something, hence that it has semantics. Because "expression" is an act towards others, there is a pragmatic and probably a social connotation, as well. Finally, the term "expressing" suggests that in actual use, there will be a physical medium (written text, spoken word, electronically transmitted bit stream etc.), possibly subject to specific empirical patterns. In other words, our notion of language is applicable at all semiotic levels as well.

A more detailed discussion of the semiotic levels will be presented in section 6.1.

3.6 Models

So far, we have seen how reasonable agreement may be reached in talking about the world. We have found that representations of conceptions often have a significance at more than one semiotic level.

When discussing the way in which we view the world and talk about it, we did not distinguish between formal and informal representations. In the context of organisations, an important aspect is that clear, precise and unambiguous conceptions of some perceptions of some organisational domains are established. These special conceptions are called models. Before defining the concepts "model" and "model denotation", let us state one more assumption.

Assumption [k]:

Human actors may have in their minds clear, precise and unambiguous conceptions, models, which they may express in the form of a model denotation, in some appropriate formal or semi-formal language.

Definition E26: Model¹, Model denotation²

A **model** is a purposely abstracted, clear, precise and unambiguous conception. •

A **model denotation** is a precise and unambiguous representation of a model, in some appropriate formal or semi-formal language. •

¹ **Warning:** This term is sometimes used for what we call 'Model denotation'.

² **Note:** The term 'denotation' has several meanings in the English language. We use it exclusively in the sense of 'precise and unambiguous representation'.

Examples:

An Entity-Relationship diagram (graphical representation) may be used to denote a model of the information-oriented aspects of an organisation. A Petri Net (graphics or algebraic expression) is often used to denote a model of the dynamics (or behaviour) of an information system. The documentation accompanying an application program (natural language + diagrams) provides a stylised description of its functionality, but usually does not constitute its model in any true sense. What is contained in the repository (or encyclopaedia) of an iCASE tool, if it was by means of such a tool that the application was designed, is a model, in that a complete mapping to an operational package may be made.

Notes:

- Our definitions of the concepts model and model denotation are chosen so as to be consistent with our constructivist views on conception and representation (see figure 3.6-1). The term model is intended to convey the feel of generic or specific precision. Effective model denotations often employ the formality of mathematics. In many cases, they are expressed in a restricted, systematic set of symbols (for instance, standardised graphics, reserved words from a more general natural language, and so on).
- A model may be denoted in many different ways, using different languages, but one particular model denotation is expressed in exactly one language.

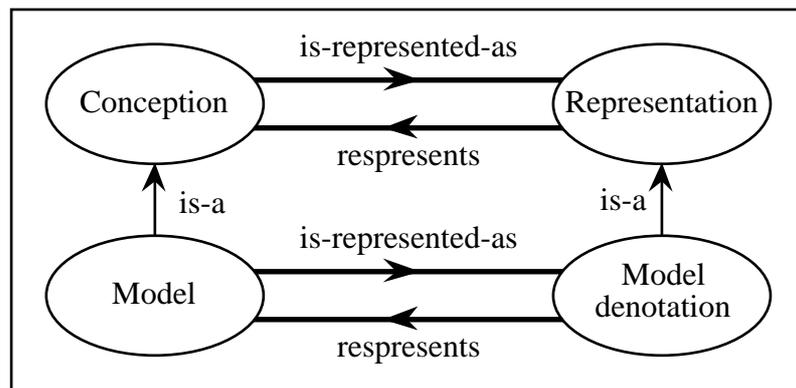


Figure 3.6-1: Models and model denotations

We define a modelling action as a special composite action comprising a perceiving, a conceiving as well as a representing action by a modeller:

Definition E27: Modelling action¹, Modeller

A **modelling action** is the sequence of a perceiving action performed on some domain, followed by a conceiving action on that perception, resulting in a model, and followed by a representing action on that model, resulting in a model denotation. •

A **modeller** is a human actor performing a modelling action. •

For practical reasons, a model is often split into two parts, an intensional part and an extensional part.

¹ **Synonym:** Modelling process

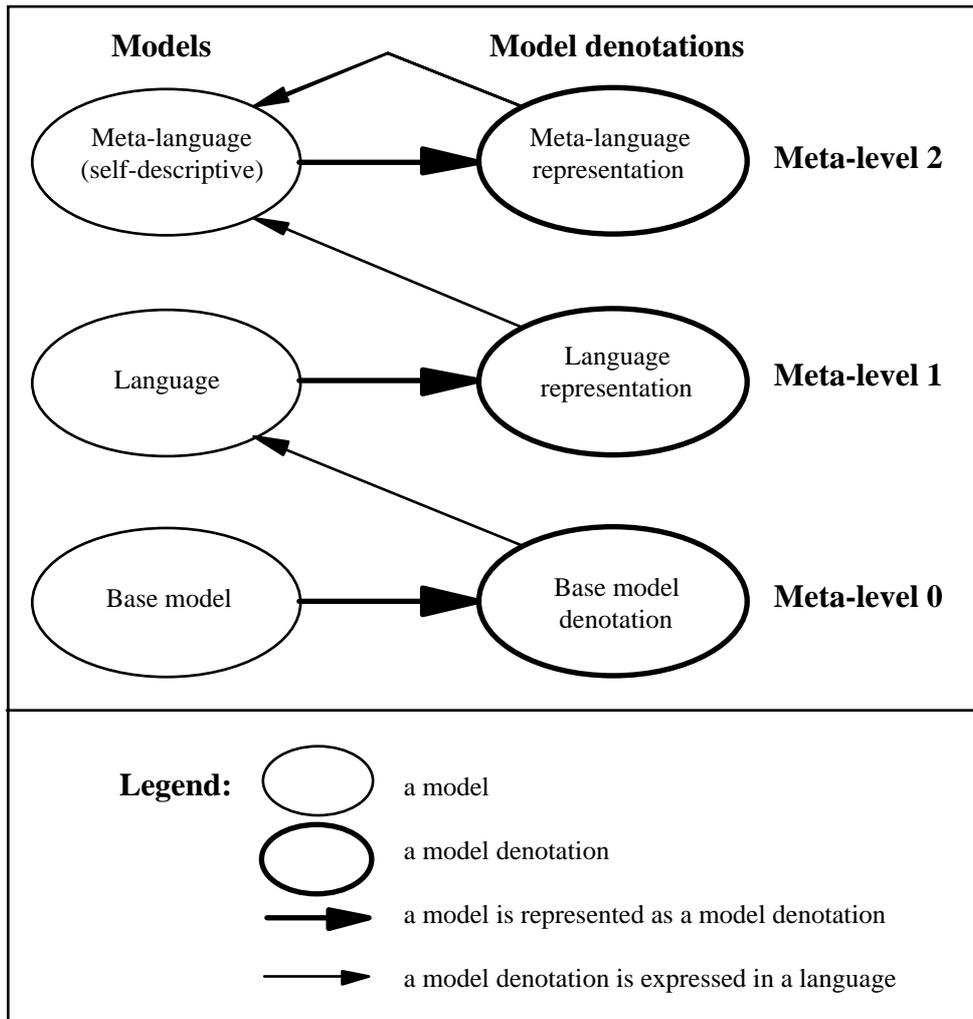


Figure 3.6-2: Models, model denotations and meta-levels

Definition E28: Intensional model, Extensional model

An **intensional model** is that part of a model comprising the possibilities and necessities of a domain only, i.e. the types and rules. •

An **extensional model** is that part of a model containing a specific population of the types in the corresponding intensional model, whereby this population must obey all rules determined in that intensional model. •

Example:

A database schema is an intensional model denotation. A database population is an extensional model denotation.

Notes:

- Each extensional model relates to exactly one intensional model, but for the latter, there may be many possible extensional models.
- An intensional model may be formulated merely for the purpose of stating all possibilities and necessities of a domain. An extensional model should always be used in conjunction with its corresponding intensional model. This is because changes of the actual state instances occur much more frequently than changes in the possibilities and necessities. Furthermore, extensional models are usually much larger and more detailed than intensional models. In case of large models, it is easier to guarantee the validity of an intensional model than that of an extensional model. It is

always necessary to check whether the actual states and transitions contained in an extensional model are consistent with what its intensional model prescribes. This is also the reason why - in developing models - the intensional model is built before attempts are made to populate it.

As said before, a model denotation is expressed in a language. The language, in turn, is described in what is called the meta-language. That meta-language will be described in the meta-meta-language, and so on. In this way, a meta-level hierarchy is formed, which begins with the original model (being normally, but not necessarily, something which itself is not a language) and its model denotation, and is open-ended towards any higher levels. We refer to this origin as the base model and the base model denotation, respectively. The meta-level hierarchy continues upwards until, at some level, a *self-descriptive* language is used, i.e. a language being sufficiently expressive to be used to formulate its own rules. For practical reasons, this is done mostly at level 2, but might also be done at level 1 already (see figure 3.6-2).

Example:

Base model:	A particular model consisting of states and transitions;
Base model denotation:	A graphical representation of this model (a state-transition diagram);
Language:	The language of bipartite directed graphs;
Language representation:	A particular specification of this language (expressed on terms of the meta-language);
Meta-language:	A first-order logic language;
Meta-language representation:	A particular specification of this language (expressed in terms of the same meta-language which is thus self-descriptive).

Each language to be used for denoting models has some conceptual foundation, consisting of a set of basic concepts and a set of rules determining the set of possible models denotable in that language. This conceptual foundation of a language may be viewed as a model too, which we call a meta-model.

Definition E29: Meta-model

A **meta-model** is a model of the conceptual foundation of a language, consisting of a set of basic concepts, and a set of rules determining the set of possible models denotable in that language. •

Notes:

- In other words, a meta-model determines a specific conceptual or ontological view of the "world", a specific classification of the concepts dealt with, and the general rules every model denoted in the language has to obey.
- A meta-model is always intensional in nature since it determines - as the basis of a language - the possibilities and necessities which are present when denoting any (intensional or extensional) model in that language.
- The concept pair extensional / intensional model can therefore be applied across meta-levels as well: A base model, viewed as a whole, is always a population of the meta-model of the language in which the base model is denoted. A meta-model is always a population of the corresponding meta-meta-model, etc.
- In this way, a hierarchy of extensional / intensional models is formed. It is only identical with the meta-level hierarchy if the base model is not split at all (e.g. in semantic network approaches), or if the base model is viewed as a whole, that is to say, if the splitting of the base model into an extensional and an intensional model is

ignored, for that matter.

In order to be able to determine whether or not two model denotations refer to models of exactly the same domain, one has to establish the notion of equivalence of models. Two model denotations (a) and (b) are said to be equivalent, if and only if the one model denotation (a) can be transformed into a model denotation (b), and this model denotation (b) can be transformed back into the original model denotation (a) without any loss, by means of appropriate transformation rules, and if and only if the same holds also for the inverse pair of transformations. This statement applies regardless of whether the models are denoted in the same or in different languages, and regardless of the meta-level of the models.

Whether equivalence of models and model denotations can be established with confidence in practice, depends both on the syntactical correctness and the semantical validity of the transformation rules. Correctness and validity, in turn, depend on the quality of the representation of the language(s) involved, and in particular on the quality of the meta-model(s).

Languages for denoting models have various qualities. Important ones are the degrees of expressiveness, arbitrariness and suitability. *Expressiveness* concerns the number and kinds of conceptions that may be expressed. *Arbitrariness* states whether there is only *one* way of modelling a certain conception (deterministic language, arbitrariness zero), or more. *Suitability* is a relative quality. It would be high, in the context of expert usage, for a very specific special-purpose language (but probably with a low overall expressiveness). It would be low for those same experts when it provides fairly elegant formulations for general purposes, but is clumsy in dealing with some specific problem domain. The choice of language is a matter of trade off vis-a-vis these aspects.

The different qualities manifest themselves in the corresponding meta-models. Straight comparisons are not only difficult because of differences in the kinds of concepts permitted and the extent to which specific aspects may be modelled, but also as a result of terminological variation.

Nevertheless, it is possible to provide a suitable ordering and transformation scheme for any meta-models. A specific approach in this direction will be described in section 6.3.

3.7 Systems

The notion of system is not uniquely defined in the literature. Many definitions can be found, such as:

- A collection of interrelated parts characterised by a boundary with respect to its environment [Iiv83];

or simply:

- A set of objects with a set of relations [Lan71].

These definitions followed on from the "classical" general system theory [Ber50, Bou56, Kli69], the essence of which was the interplay of known or presumed components. In that view, a system is an abstract construct that might be contemplated in its own right. As such, it is described as a formal expression for that set of components, together with a way in which they are connected. From the properties of the elements, one may derive the overall behaviour and characteristics of the system. This kind of description also permits an enumeration of all allowable states and (often) also of any transition that might occur. Analytically, therefore, it is useful to describe a system as a set of known components and to add an explicit or implicit states-and-transitions description.

This classical view leaves a number of questions unanswered, in particular some that are relevant in the early stages of analysis of a complex domain. The various persons involved may have very different conceptions of it. While all conceive some coherence among a choice of elements from the domain, individually, their selections may be different, different elements may be assigned significance or dominance and they may recognise entirely different overall characteristics of what they see as a whole. In [Lin90a], therefore, it is argued that a system is not an absolute or objective phenomenon. According to Checkland [Che81], there must be an observer and describer who conceives and thinks about the world or some part of it as a system. Through some process of communication and negotiation (as introduced in section 3.5 and further developed in section 3.8), various personal views may converge. We feel that these considerations are best expressed in the following assumption and subsequent analysis:

Assumption [I]:

When viewing some domain, human actors often conceive coherence in that domain or assert the presence of such coherence. The resulting model of that domain, made up of coherent elements, is called a system. Anything outside that domain will be considered belonging to the system environment.

Accordingly, we found that the notion of a system is best defined in the context of how we perceive a domain, forming conceptions and ultimately achieving knowledge about it.

Definition E30: System¹, System denotation, System component², System environment, System viewer, System representer

A **system** is a special model, whereby all the things contained in that model (all the system components) are *transitively coherent*, i.e. all of them are directly or indirectly related to each other form a coherent whole. A system is conceived as having assigned to it, as a whole, a specific characterisation (the so-called "*systemic properties*"). •

Notes:

¹ **Warning:** This term is used in various different meanings in the literature [e.g. Ber50, Bou56, Kli69, Ack71, Bun79, Che81]!

² **Alternative term:** System element

- The requirement of transitive coherence implies that the model must be a set of transitively coherent *relationships*, since other things cannot be coherent. Two relationships R1 and R2 are *directly* coherent if and only if they contain at least one predicated thing in common. Two relationships R1 and R3 are *transitively* coherent if and only if they are directly coherent or there exists a relationship R2 such that R1 and R2 as well as R2 and R3 are (directly or transitively) coherent. A set of transitively coherent relationships may for instance be represented as a fully connected graph.
- Since a system is a special model and a model is a special conception, all the considerations about conceptions and models apply here as well (see figure 3.7-1).

A **system denotation** is a precise and unambiguous representation of a system. •

A **system component** is a non-empty set of things being contained in that system. •

The **system environment** of a system is the set of all things not being contained in that system. •

A **system viewer** is a human actor perceiving and conceiving a domain as a system. A system viewer recognises the system, by its distinction from the system environment, by its coherence, and because of its systemic properties. •

A **system representer** is a human actor representing a system in some language. •

Examples:

When describing the physical world, we consider the complex "earth-plus-atmosphere" a system, and likewise for the "solar system". An "enterprise", which is a coherent grouping of resources (people, capital goods, offices, finances, etc.), is also viewed a system in our sense. Any piece of machinery functions as a system, for instance a computer, a network, and so on.

Note:

Our conceptual framework can be viewed as a system of concepts, since it is transitively coherent.

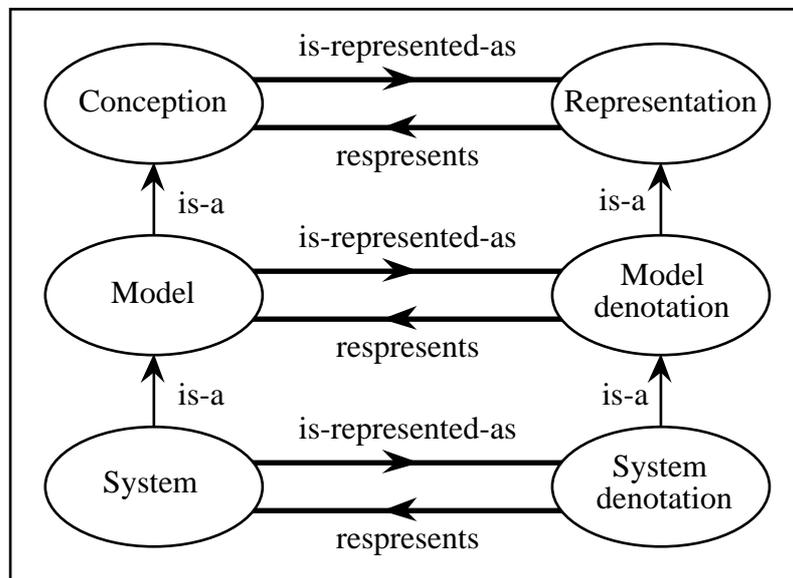


Figure 3.7-1: System and system denotation

The essential elements of the definition of system may be related graphically by recourse to the semiotic tetrahedron (see figure 3.7-2).

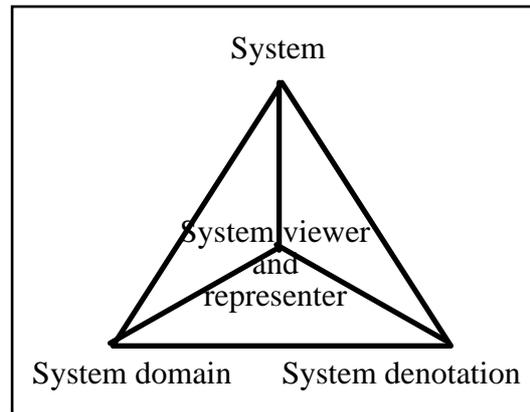


Figure 3.7-2: The FRISCO semiotic tetrahedron adapted to the notion of system

The classical components-and-interactions and states-and-transitions descriptions leave open the question what the systemic property is (other than the constructor's assertion that this collection is considered to be the system studied). Domains are especially interesting if they possess at least one true systemic property (that is not merely the sum of the component properties). Thus, the analytical formulation is only applicable once the system and its systemic properties have been recognised as such. In the domain of information usage the following are typical examples of system descriptions:

Examples:

- A computer, i.e. a number of hardware and software components, has as systemic property the capability of rendering data storage and processing services. Should any component fail, then the performance will deteriorate and often, the machine would malfunction altogether.
- An enterprise, i.e. a number of co-operating resources, has as systemic property the ability to act as one whole, by inter-communication of staff and departments. Should any of the inter-personal or inter-departmental links be severed, then much if not all of the organisation's effectiveness is lost.

Agreement between two human actors as to what constitutes a particular system is a matter of exchanging representations (along the lines of figure 3.5-1). Thus, one may say that the conceptions of two human actors about the same domain may not be the same at the start, but we assume that communication will ultimately help establishing a useful joint system and system denotation.

Systems may be classified according to several characteristics:

Definition E31: Dynamic system, Static system, Active system, Passive System, Open system, Closed System

A **dynamic system** is conceived as capable of undergoing change, i.e. some of the system components are transitions. •

A system that is not dynamic is called a **static system**. •

An **active system** is conceived as capable of doing something, i.e. some of the system components are actors performing actions on actands. •

A system is called a **passive system** if it is not active. •

An **open system** is conceived as one which may respond to external messages or triggers, •

i.e. there may be transitions within the system due external causes coming from the system environment. •

A system is called a **closed system** if the system environment cannot cause any transitions within the system. •

Clearly, some of these qualities are exclusive. For instance, a closed system might be both dynamic and active, but it would not be conceived as communicating with the environment and, therefore, be of theoretical interest only. An active system would not be static, for action implies changes of something (except, possibly in the somewhat hypothetical system that would be conceived as doing something to its environment, without any internal change). An open system cannot be static, by definition, its openness implies capability of change.

Later on, we shall see that organisational systems and information systems are in general open, active and dynamic. However, a mere administrative system for example is open and dynamic, but not active, i.e. any changes must be administered from outside.

Any part of a system may be a system in its own right and then be called a *sub-system* of the original system (the *super-system*). One or more of the systemic properties of the sub-system may undoubtedly constitute the sub-system's contribution to at least one of the systemic properties of the super-system. However, the system viewer in question may conceive other characteristics which do not necessarily contribute explicitly. We define "sub-system" as follows:

Definition E32: Sub-system

Any **sub-system** S' of a larger system S is a system, itself. The set of all sub-system components of S' is a proper subset of the set of all system components of S. •

Examples:

- The central processor (execution of instructions) and the local bus (data interchange) are sub-systems of a computer (capability of rendering data storage and processing services), but only constitute (part of) the enabling technology.
- The sales department (order acquisition and processing) and the accounts department (bookkeeping and budgeting) are essential to the enterprise they serve, but do not reflect the overall objectives (e.g. constituting a good investment to the shareholders, rendering some public service).
- The engine of a motor vehicle provides propulsion, an essential contribution to the overall objective of comfortable transport; some characteristics of the engine may contribute negatively, however (e.g. its noise must be dampened by insulating material under the hood), others may be irrelevant (e.g. the beauty of its design may only be a matter of pride to its designer, without contributing to the vehicle's quality, as such).

What constitutes a sub-system, therefore, is a matter for the system viewer in question. In the three examples, the situations are straightforward, but in many information system design cases, there may be a wide choice of options. These may be resolved by the aforementioned process of communication between the parties involved, which would concentrate on relevant contributory features.

In the road transport enterprise case discussed at the end of section 3.3, the following would be representative examples of some of the concepts introduced since:

Concepts:	Some examples of instances of concepts:
Domain	A road transport enterprise

Conception	any of the things mentioned already at the end of section 3.3
Language	order conventions, company circular conventions, truckers jargon ...
Representation	order message ...
Semiotic level	bill of lading (contractual = social, transportation guarantee = pragmatic, stipulations = semantic, statements = syntactic, written lines = empirical, electronic record = physical) ...
Model	day planning model, staff development plan ...
System	the enterprise as a whole, the long term and day planning systems ...
Sub-system	any department, the garage, today's crew, credit/debit control ...

3.8 Organisations, Information and Communication

In anticipation of a more precise definition, let us remember that we viewed an organisation as a delimited social structure within the context of which actions are undertaken - on the basis of information - because they are deemed useful or necessary. We now observe that such a structure displays *systemic* characteristics. There are numerous *active* components (people, machines, etc.) that appear to belong together and that interact in meaningful ways. At the highest level, there usually are a few clear objectives that lead to a very specific overall behaviour (goal-seeking, on the basis of organisation-wide co-operation). Any view of an organisation satisfies the conditions of our definition of a system. This also applies to the more limited contexts of specific parts within them, such as the production department, the personnel, financial and material management, the internal control mechanism, and so on. These may be viewed as *sub-systems*.

Now, the most characteristic aspect of these systems and sub-systems is that the coherence and interaction is based on a capability of exchanging meaningful representations, which we associate with the intuitive notion of information. Typically, that information may be characterised as "knowledge-for-decision-making-and-action". And in many instances, it is acquired by one actor as a result of communication with another actor. All this may be summed up as follows.

Assumption [m]:

Organisations are characterised by (usually goal-oriented) arrangements between a number of parties, whereby (mostly co-operative and co-ordinated) action is enabled through (often systematic) communication, which entails exchange of information, involving meaningfully composed messages.

These arrangements concern actions on any kind of actand (material or informational object on which the action is or may be performed). The essence of these arrangements is that actor-actor interaction takes place, namely through communication, often deliberately oriented towards a common goal. Even if a goal is not explicit and/or the action is not truly co-operative, some form of interaction may exist. An explicitly or tacitly agreed organised situation may then be said to exist. Actands may be other actors, devices, material goods or abstractions, such as money or data about any of these things. All of these may be characterised as actands.

The concept knowledge can be characterised as "a relatively stable and consistent collection of conceptions". We will speak of (personal) knowledge, meaning the conceptions of a single person, and of shared knowledge, referring to the union of common conceptions of all persons

considered, in particular in the context of an organisation (organisational knowledge). Some knowledge may be acquired by people through perceiving and conceiving domains which are not representations, and through inner reflections about their conceptions. Some knowledge may be acquired by people through receiving and interpreting messages representing knowledge of other people, i.e. through message transfer and communication. Messages may be transient (e.g. spoken sentences) or available in a permanently recorded form (e.g. books, memos, notes, databases, etc.). We will define information as that kind of knowledge increment people acquire through message transfer and communication.

Let us now review the various characteristics we associate with the concepts information and communication. The two are related since information concerns knowing (or getting to know) something, and communication is the prime process of providing it. However, what is the nature of that "something"?

Our constructivist approach implies that we consider in particular how we can exchange representations of conceptions and try to reach agreement about them. Hence, if information means having or acquiring knowledge, all aspects of expressing conceptions apply. However, if information would imply providing all that might be "known" about a thing, one might easily receive an excess of it. The concept is also associated with a sense of utility, especially with usefulness for decision making and action. Thus, providing *useful* information should be restricted to really *meaningful* elements (by filtering out the irrelevant, i.e. by abstraction). Useful information in support of an action concerning X may even entail knowledge about some Y, rather than about X as such, if that would be relevant for making the decision in question. For instance, the decision to invest in a particular kind of productive capacity may require information about the future market for its product rather than past production costs.

Useful knowledge involves a variety of properties of (and/or concerning) the thing one seeks or wishes to provide information about. Typical characteristics of information are:

- the domain to which the thing is conceived to belong,
- the thing's identifier (i.e. a unique representation of it, e.g. a unique label),
- the thing's state(s) and the period(s) of validity thereof,
- the thing's (expected) behaviour and influencability and time dependent changes thereof,
- the thing's "environment" (i.e. other things that affect it).

Considering them in terms of the semiotic levels, one observes that they are all at the syntactic level. In figure 3.5-2, formal structure, language and data, among others, are mentioned as examples. The characteristics mentioned here are in the same categories.

On the other hand, the utility and action (intention) connections mean that - implicit or explicit - expression will (have to) be given to aspects that link information to the other semiotic levels:

- associated semantic values (meaning of what is represented)
- associated pragmatic values (intentions of information conveyer, acceptance by receiver)
- associated social values (implied or explicit agreements, etc.)
- required physical representation (spoken or written word, computerised recording, etc.)

The modelling needed when eliciting, sending or receiving information, in general, leads to representations that are comprehensible to more than one party. Whilst information, as a concept, intuitively seems to be associated with a single actor (who contemplates action on the basis of it, or wishes to inform another actor), in most cases, it is the subject of communication in which several actors are involved.

The following definitions fit the characteristics that we have summed up. Let us begin with the notion of "knowledge":

Definition E33: Knowledge¹

Knowledge is a relatively stable and sufficiently consistent set of conceptions possessed by single human actors. •

Notes:

- The phrase 'relatively stable' implies that the state of the relevant collection of conceptions is not subject to major or frequent transitions, whereas the phrase 'sufficiently consistent' implies that practically speaking only a few, if any, rules are violated that apply to the domain covered by the knowledge in question. These aspects, although fuzzy, are quite relevant for making judgement and action based on knowledge reasonably successful. To be useful, this concept also has to encompass further aspects such as intention, will, etc.
- *Since conceptions are strictly personal, so is knowledge. Knowledge can, however, be shared by people through message transfer and communication.*
("Shared knowledge", see below)

Before we can define communication, we need the concepts "data" as well as "message" and related notions.

Definition E34: Data

The term **data** denotes any set of representations of knowledge, expressed in a language. •

Notes:

- In other words, data are meaningful symbolic constructs (expressed in a language), which can be qualified as "knowledge-bearing".
- Some non-sense or some corrupted sequences of symbols do *not* qualify to be called data.

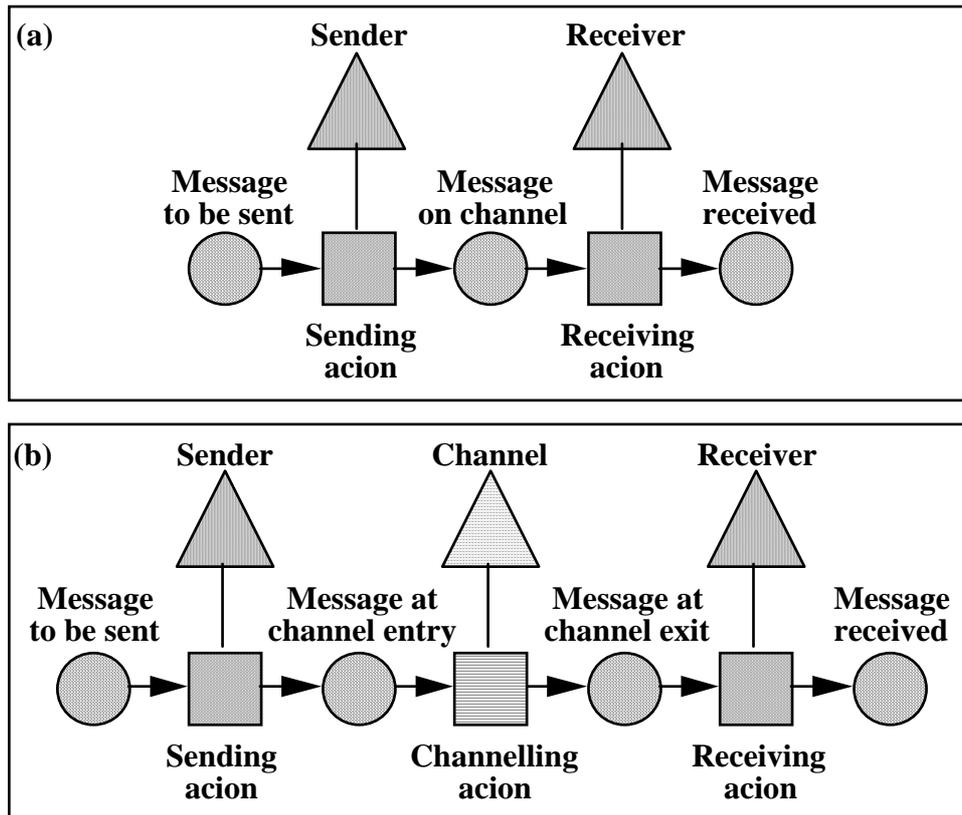
Definition E35: Message², Message transfer, Sender, Receiver

A **message** is data, transmitted by one actor (the sender) via a channel (a medium), and intended for a non-empty set of other actors (the receivers). •

A **message transfer** is a sequence of actions, the sending action by the sender and the receiving actions by the receivers, whereby the input actand of the sending action is the message to be sent, whereby the output actand of the sending action, being (in the simplest case) equal to the input actand of the receiving action, is the message on the channel, and whereby the output actand of the receiving action is the message received. •

¹ **Warning:** This term is used in various different meanings in the literature!

² **Warning:** This term is used in various different meanings in the literature!



Legend:

- | | | | |
|-------------------|---------|---------|--|
| Triangles: | Actors | Lines: | Actors perform actions |
| Rectangles: | Actions | Arrows: | Actands are input or output of actions |
| Circles or ovals: | Actands | | |

Figure 3.8-1: Message transfer: (a) simple case; (b) extended case with channel

Notes:

- A channel is any medium of transfer, e.g. the air carrying sound, sheets of paper in a book, an electronic network, etc. If the channel has no memory (such as in the case of the air carrying sound), the message is necessarily *transient*, otherwise (such as in the case of sheets of paper in a book or magnetic disks in an electronic network) it may be sufficiently *permanent*.
- This definition covers only the simple case of *undistorted* message transfer, i.e. it abstracts from the problem that data may be distorted or corrupted when being transported on the channel, in particular if a technical device is used as channel. To cover these issues as well, one needs to introduce a third actor, a *channel*, and a third action, a *channelling action*, whose input and output actands (a message at the channel entry and at the channel exit, respectively), may or may not be identical (see figure 3.8-1).
- We can distinguish various aspects all of which are relevant in our context. First of all there is the aspect of the intention of the sender, i.e. why does he represent something at all and send it (presumably but not necessarily some of his knowledge). Then there is the aspect of the reliability of the channel, i.e. whether or not the message remains undistorted. Furthermore there is the aspect of the intentions of the receivers, e.g. why do they receive and interpret the message at all, does the message (in their interpretation) contain any knowledge new to them ("information"; see below), and how will they use that knowledge, whether new to them or not.

Examples:

- Speeches, letters or e-mail messages addressing one or several people;
- Books ("sent" by the author(s) and "received" by the readers);
- Databases ("sent" by the database providers and updaters and "received" by the database retrievers).

A **sender** is an actor sending a message. •

A **receiver** is an actor receiving a message. •

The notion of message transfer and related concepts are illustrated in figure 3.8-1.

Receiving a message will result in a perception that may (and generally does) change the receiver's conceptions. The same may happen when a person refers to a source of data (knowledge representations), such as a book, a database, or a catalogue, i.e. when the message has been made available without an immediate specific addressee. We call the *increment* of knowledge, brought about by receiving a message, the "information" associated with the message transfer, defining it as follows:

Definition E36: Information¹

Information is the knowledge increment brought about by a receiving action in a message transfer, i.e. it is the difference between the conceptions interpreted from a received message and the knowledge before the receiving action. •

Notes:

- People speak about "sending information" to someone, or "informing" somebody. It means that the sender has merely the *intention* to bring about a knowledge increment in the receivers.
- An important aspect of information is how it is *used* by a receiver. It may be used in a variety of ways. Normally, the intention of the sender is to provide or share useful knowledge such that rational decisions² can be made by the receivers in connection with some contemplated actions.
- The intentions of the sender may of course be not co-operative at all. He could for example consciously lie to the receivers, to try to avoid or trigger certain actions of them. Such cases are better called "false information" or "dis-information".

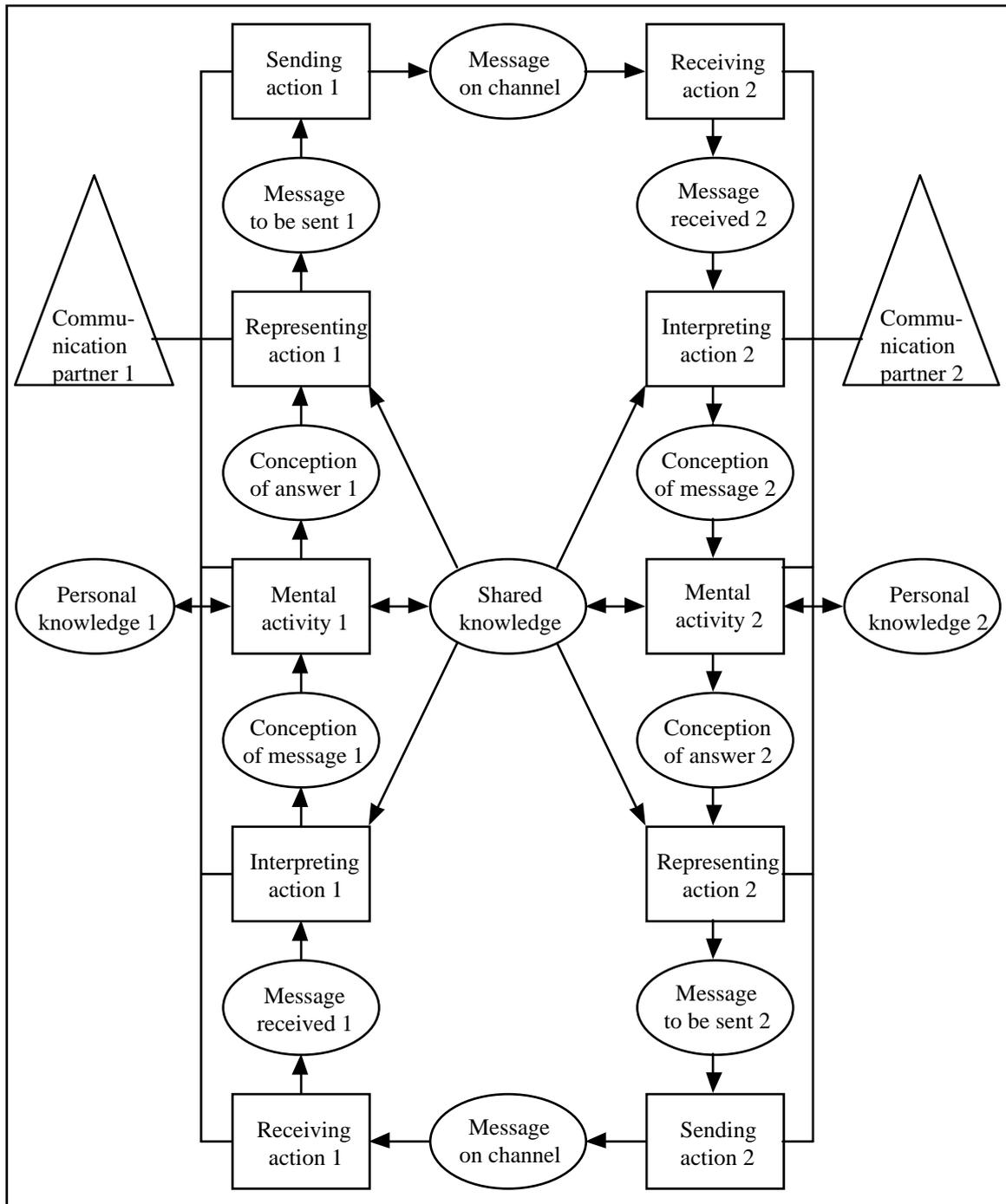
Examples:

- reference to a road map (a representation of geographic knowledge) may help decide whether there is need to turn off a road (decision to act - the decision is triggered by the information) or to continue (decision not to act - the information provides reassurance that one is on the right road);
- a road sign may indicate that the contemplated exit is temporarily blocked - reading it helps form the conception that the situation differs from what is represented on the map (supplementary information);
- a blinking signal advises that an unexpected situation is ahead (warning information);
- reference to an encyclopaedia, a database, the news, a company memo, a telephone call or an e-mail message from a friend - all may provide information helping to decide on action or inaction, as appropriate - possibly for reassurance or enjoyment, possibly for future action.

Information also stands for anything that helps achieving a more stable view of "reality", as experienced in an individual person's conceptions. It will be further enhanced by purposeful interaction between persons, along the lines of figure 3.5-1, when the representations exchanged represent their respective views about that "reality".

¹ **Warning:** This term is used in various different meanings in the literature [e.g. Lan63, Sta73, Lan77, FL89, Fuc92, FRE92, FHO95]!

² The term 'decision' is to be understood here as in common language, meaning that a choice is or may be made.



Legend: Same as in figure 3.8-1.

The actand called 'shared knowledge' comprises knowledge about the language(s) used, and a certain amount of knowledge about the domain, the context and the goals of the communication.

Figure 3.8-2: Communication between two communication partners

The interactive process referred to presupposes a capacity indeed to interact. For this to be effective, a number of conditions must be met such that the parties understand each other, i.e. form conceptions and exchange representations in a reproducible and stabilising manner. When such a situation prevails we shall speak of "communication".

Human actors are communicating if they exchange representations, such that one of them (in the role of the sender) is representing his conceptions in a language, and the other ones (in the

role of the receivers) are interpreting these representations, i.e. form conceptions thereof, and possibly answering (taking over the role of sender) by representing their own conceptions in the same or another language, which can in turn be interpreted by the first actor (taking over the role of receiver), and so on. For successful communication, it is not only necessary to use languages which are understood by the respective receivers. Equally important is that the communicating partners all possess some initial knowledge about the domain, the context and the goal of the communication and that this knowledge - whilst personal for each - sufficiently corresponds to that in the others. If such initial "commonly shared knowledge" is indeed present, a process may take place as illustrated in figure 3.8-2, which may be defined informally, as follows:

Definition E37: Communication¹

Communication is an exchange of messages, i.e. a sequence of mutual and alternating message transfers between at least two human actors, called communication partners, whereby these messages represent some knowledge and are expressed in languages understood by all communication partners, and whereby some amount of knowledge about the domain of communication and about the action context and the goal of the communication is made present in all communication partners. •

Notes:

- On a coarser level of "granularity", communication can be defined as a special *co-action*. In this case one would view communication as *one* (composite) action performed by more than one actor in a co-ordinated way.
- The way in which communication is defined here and illustrated in figure 3.8-2 corresponds to the reasoning applied in speech act theory [e.g. Sea69].

Examples:

- advising in face-to-face contact or by telephone, exchanging views in a meeting, mailing a report and receiving comments about it, adding data to a database and extracting data from it, publishing an article or a book and exchanging views about it with the readers ...
- sending a copy, repeating advice and rehashing arguments in a meeting, in principle, would not qualify as communication, because these actions do not contribute to knowledge as such; however, the exchanges involved may serve as reassurance ...

The two parties involved in the exchange of figure 3.5-1 were viewed as one actor uttering a representation and the other actor interpreting it. A full cycle of communication activity - from establishing a first contact to final agreement about the information to be shared and termination of the contact - takes a number of steps that vary depending on the nature of the contact and the formality of the representations employed. Clearly, a telephone conversation will drift more than a formal reporting action that follows a prescribed protocol. Yet, in all cases, there will be alterations of sending actions and of receiving/interpreting actions.

The initial "commonly shared knowledge" needed in order to enable effective and hopefully purposeful communication is a specialisation of a wider body of knowledge, namely the sets of (personal) conceptions simultaneously possessed by a collection of persons. While they cannot actually know what is in the minds of other people, the result of repeated communication may be that their conceptions about some domain stabilise to such an extent that they conceive the situation as one where their knowledge about it is the same. We define "shared knowledge" as follows:

Definition E38: Shared knowledge

¹ **Warning:** This term is sometimes used also for what we call 'Message transfer'!

Shared knowledge is that knowledge of the individuals in a group of human actors, which they assume to be identical (or at least similar) to that of the others, as resulting from the negotiation process implicit in some communication. •

Notes:

- Thus, shared knowledge is a subset of (personal) knowledge.
- When communication has resulted in shared knowledge regarding some domain, those sharing that knowledge may be said to have achieved "inter-subjectively negotiated conceptions" concerning an "inter-subjective reality".
- *Once shared knowledge has come about, the naïve realist and the constructivist will speak about it in identical terms.*

We will now take a closer look at the notion of "organisation". Before defining an organisation as a system, we make the following assumption:

Assumption [n]:

An organisation is a grouping of actors, together with a collection of actands, such that (a) a common goal is pursued or some other characteristic coherence is displayed, and (b) interactions occur that are based on communication. Information is used in an organisation in the context of its functioning (actions), both internally and with respect to its environment, the society at large. Because of this social involvement, norms are meaningful directives for individuals in the organisation and the organisation as a whole.

Definition E39: Organisational system, Norm

An **organisational system** is a special kind of system, being normally dynamic, active and open, and comprising the conception of how an organisation is composed (i.e. of specific actors and actands) and how it operates (i.e. performing specific actions in pursuit of organisational goals, guided by organisational rules and informed by internal and external communication), where its systemic properties are that it responds to (certain kinds of) changes caused by the system environment and, itself, causes (certain kinds of) changes in the system environment. •

Norms are socially agreed rules affecting and to a large extent directing the actions within an organisational system. •

Examples:

One very general norm of society is that its rules (laws in the legal sense) must be obeyed. Other norms may be formal codes of good practice. There may also exist norms that are only informally stated or tacitly accepted throughout the organisation.

An organisational system is an abstraction of how an organisation appears to manifest itself in the "real world". The groupings of actors and actands constitute the components that form a whole, and there are the systemic properties of (a) having an organisational goal or some other factor of coherence, and (b) operating on the basis of "action-enabled-by-information". These are two qualifications that are neither possessed by the environment, nor by components making up the organisational system.

The practical purpose of describing an organisation as a system may be to prepare for organisational change and/or for planning computerised support systems. As said before, an actor may be a human actor (or grouping of human actors), but also a device. Actions of a

device do not come about autonomously, but because they have been set up (programmed) and triggered by some external force or effect. For example, a computer will respond to its user pressing a key, or to encountering a particular programmed condition. Without further speculation about psychological aspects, behaviour of human actors may be assumed to be autonomous, but based upon consciously or unconsciously formed conceptions. In our constructivist approach, such conceptions must either have been available all along or resulted from personal perception or inter-person communication.

Organisations do not exist in isolation. In fact, they are usually dynamic, open and active systems. Their involvement has both pragmatic and social significance, as must have been clear from the discussion above.

In this section, we have seen how organisations may be viewed as information-using arrangements, where information is defined as knowledge increment imparted by communication through message transfers and used for actions. Information clearly is significant at the pragmatic level (providing a basis for action) and the social level (because of publicly or tacitly accepted norms). Information is represented by and/or derived from collections of data, which in turn have physical and empirical properties and are subject to syntactical and/or semantical rules, thus having significance at the physical, empirical, syntactical and semantical level.

3.9 Information Systems

The actual use of information by and within an organisation displays systemic characteristics.

Assumption [o]:

Organisational information usage derives from its operational practice, which may be present in the form of its organisational system (by the "business model"); within it, one finds embedded what may be described as a complex net of information flows in support of its actions; some of the information flows may be arranged within a computerised information sub-system.

Now, practical computerised systems will deal with restricted parts of the overall information flows and usage in the organisation only. Thus, they may be considered specialisations of the more general information-providing arrangements one comes across as part of organisational actions. The full extent of the informational issues can also be viewed as a system with elements such as messages, message senders/receivers, message media and any traditional equipment (telephones, writing pads), as well as advanced aids (computers, networks). The systemic property, however, does not co-incide with, nor is it even part of that of the organisational system. In this instance, it is the capability of providing information. Such a system - an information system in the most general sense - may be defined as follows.

Definition E40: Information system¹, Information system denotation

An **information system** is a sub-system of an organisational system, comprising the conception of how the communication- and information-oriented aspects of an

¹ **Warning:** This term is used in various different meanings in the literature [e.g. Lan63, Sta73, Lan77, Ver87, Oll+88]!

Synonym: Information System in the Broader sense (Acronym: ISB)

organisation are composed (e.g. of specific communicating, information-providing and/or information-seeking actors, and of specific information-oriented actands) and how these operate, thus describing the (explicit and/or implicit) communication-oriented and information-providing actions and arrangements existing within that organisation. •

An **information system denotation** is a precise and unambiguous representation of an information system. •

The organisational system as well as the information system are instances of the general type system. Any specific information-providing arrangements are clearly part of all that is comprised by the organisational system. They form a system, themselves, but the systemic properties are so different that one cannot say that an information system is a specialised instance of the organisational system it is embedded in. One can only say that a system-sub-system relationship exists.

On the other hand, one may be more explicit in describing the various kinds of informational arrangements. For instance, our definition of an information system implies consideration of the full extent of information flows and usage, as one whole.

Any actual instance of an information system (in the broader sense) comprises all informal and formal informational actions and all knowledge- and data-processing actions within the organisation in question.

If one considers only the formally defined information-providing actions set up and/or planned in an organisation, one likewise observes a systemic arrangement (a set of interacting elements with an overall purpose, i.e. a systemic property). Since these elements are restricted to what is or may be computerised (or at least so described), an appropriate name is "computerised information sub-system" (CISS).

Definition E41: Computerised information sub-system¹

A **computerised information sub-system** is a sub-system of an information system, whereby all actions within that sub-system are performed by one or several computer(s). •

The systemic properties of information systems in the "broader" and the "narrower" sense are not the same. An actual ISB would include a description of informal information flows, which any CISS designed to fit into it would not. Yet, the nature of their information services is of the one kind.

To what extent the detailed specification expressed by a CISS denotation may indeed be viewed as a sub-system denotation of the more extended, but also more globally stated information system denotation, is a matter of taste or definition. At any rate, if the same level

¹ **Acronym:** CISS (Computerised Information Sub-System)

Synonym: Information System in the Narrower sense (Acronym: ISN)

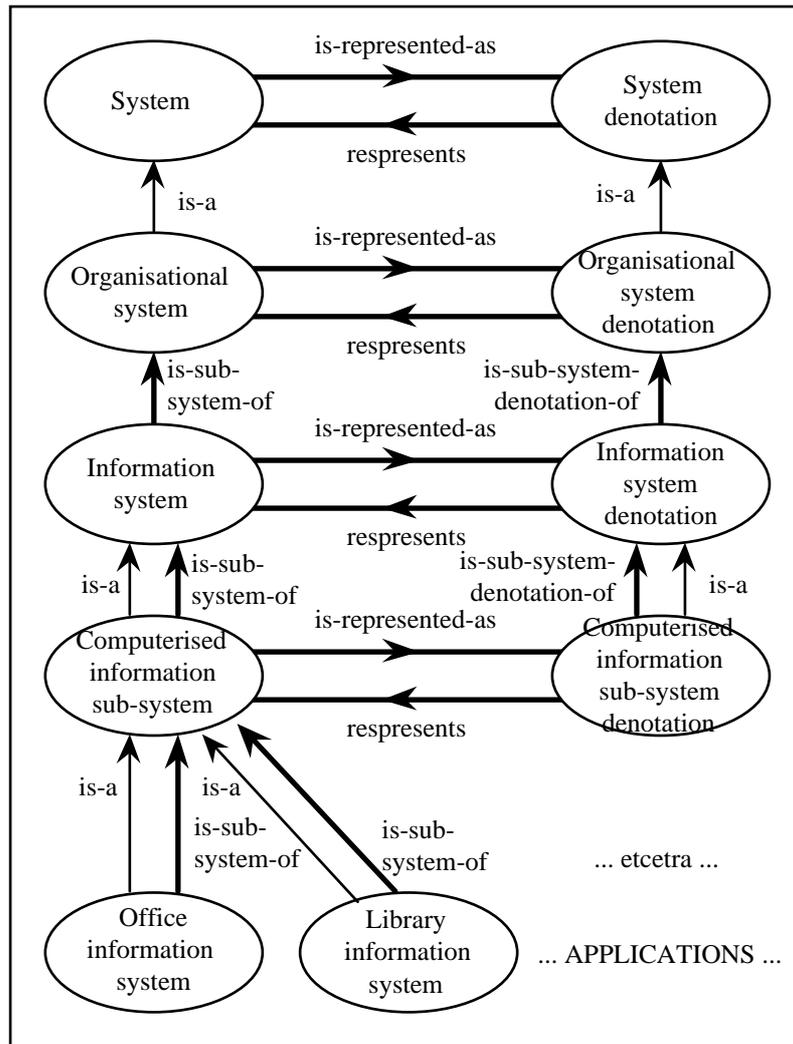


Figure 3.9-1: Organisational and information systems and their denotations

of detail is practised in both, it is justified. The CISS then constitutes a system having the various actual computerised applications as sub-systems. One may therefore conclude that the following specialisation ("is-a" sub-type) and containment ("is-sub-system-of") relationships hold:

- An organisational system is a system.
- An information system is a system; a specific information system will also be a sub-system of some specific organisational system.
- A CISS is an information system; it is also a sub-system of some specific information system (and, hence, also a sub-system of an organisational system).
- A computerised application is a system; it is also a sub-system of some specific CISS (and, hence, also of the corresponding information system and organisational system).

These relationships, which are consistent with our definitions of system and sub-system, are illustrated in figure 3.9-1.

Because two kinds of relationship are intermixed, a brief word of how their mutual relationship should be understood.

The levels of abstraction in figure 3.7-1, from high to low, stand for increasing specialisation, for example representation (most general) - model denotation (more specific representation) -

system denotation (most specific). The "is-a" relationship expresses the fact that something is a sub-type of some given type.

The sub-system definition states that there is an "is-a" relationship of the concept sub-system with respect to the concept system. However, because of the "systemic-property" characterisation, it does not necessarily follow that a specific instance of a system and one of its sub-systems are of the same type. For instance, consider a motor vehicle and its engine. Both clearly constitute systems, one with the systemic property "transportation-capability", the other with "propulsion-capability". One would call that motor vehicle an instance of the type "motor vehicle". On the other hand, the engine would not be considered an instance of a the type "motor vehicle" at all, even if it is part of one. Thus, the "is-sub-system-of" relationship merely expresses containment.

The following considerations may further clarify these points. The information system represents a partial aspect of what is covered by the organisational system. Hence, one may view it at best as a sub-system of the organisational system. From an "informational" point of view, the information system is all-embracing, and possesses all the characteristics of a dynamic, active and open system. On the other hand, the CISS, which covers only what is (or may be) computerised excludes many informally described aspects. Thus, the CISS, at best is a sub-system of the information system, even if it may also be viewed as a (more restricted) information system.

Actual software-systems ("applications"), forming sub-systems of the over-all CISS, are also systems, as well as information systems. Note that the representations of an organisational system and the information systems embedded in it, are expressed by a system denotation in some appropriate language.

A full information system denotation may not always be formulated explicitly, but may be implied in the combined "informational" aspects mentioned for the organisational system. Typical partial examples are a financial plan, a human resources plan, a logistic plan and the like. These constitute important organisational abstractions, i.e. information about the organisation and its environment, suitable for deciding appropriate courses of action. Ideally, all partial model interrelationships should be studied, although a comprehensive model may be difficult to compile and impossible to keep up to date. Yet, in principle it is the basis for any detailed planning, design and evolution of computerised systems.

Someone analysing the "business activity" may restrict himself to information flows and usage in rather broad terms. Both a professional information system designer and the casual user of some high-level computerised tool might be thinking of the logic ultimately to be displayed by the application that will result from what they formulate. This would also hold for a knowledge engineer, who attempts laying down the strict rules for some support system. A software engineer, on the other hand, is definitely concerned with the specification of the software system that is to perform some given (simple or complex) task. Finally, the actual constructor of a computer-based system and the casual tool user - once his choice of parameters has resulted in an operational system - associate the terminology with the computer-based application, itself.

In the remaining section 3.10 we will introduce some further considerations about computerised information sub-systems, partially using our conceptual framework and partially going beyond it.

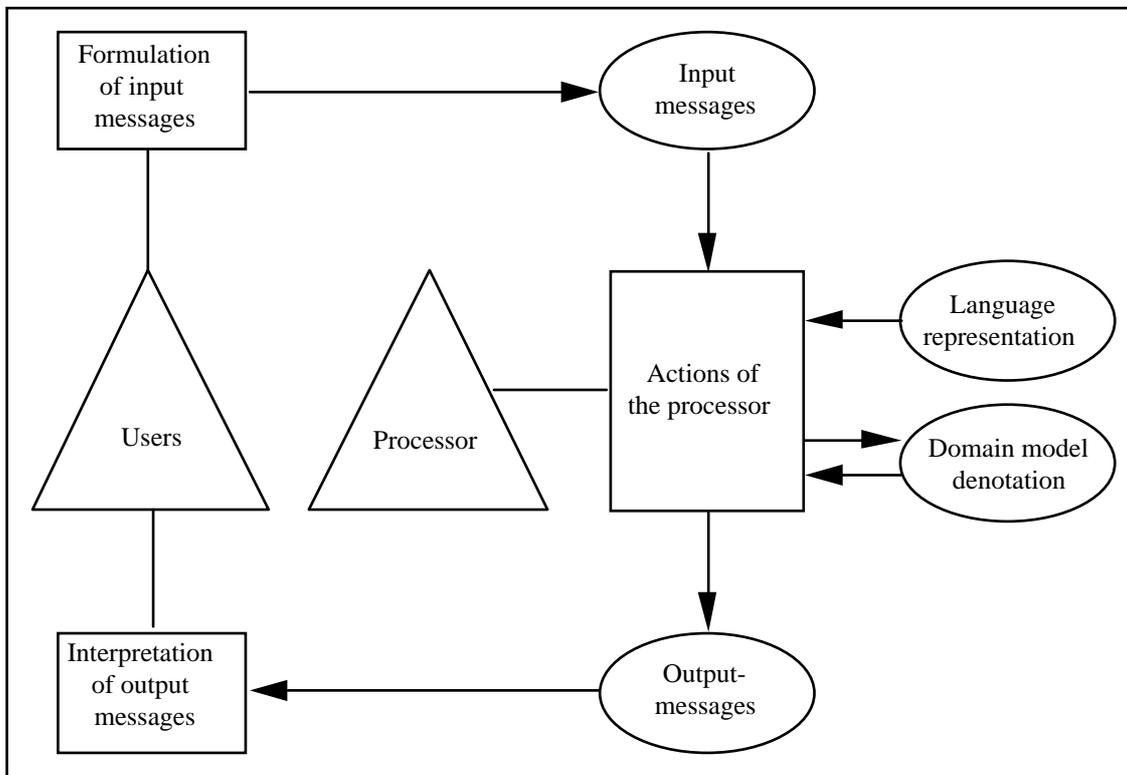
3.10 Computerised Information Sub-systems

Firstly, let us consider the components of a computerised information sub-system (CISS) in some more detail.

We assume that in a CISS, there is *conceptually* only one actor, the "computerised information processor" (henceforth processor, for short). The processor is a device which receives input messages concerning the domain of communication in the context of the organisation (henceforth domain, for short), and produces output messages about the same domain.

In practice, the processor may well be and often is *physically distributed*, such that several co-operating computers perform the necessary actions.

A CISS might be considered to be a communication partner for many "users" who want to communicate about one and the same domain (e.g. the "responsible agents" connected to a single "responsive agent" in figure 3.1-1). However, since we have reserved the notion of communication partner to *human actors*, we prefer to say that a CISS is used as a (somewhat sophisticated) channel for the message transfers between the various users.



Legend:

- Triangles: Actors
- Rectangles: Actions
- Ovals: Actands
- Lines: Actors perform actions
- Arrows: Actands are input or output of actions

Figure 3.10-1: A representation of a CISS and its users

A user specifies input messages to the processor and receives specific output messages from the processor. A user is often a human actor, but may be also some artefact, such as a sensor that presents signals corresponding to its measurements as input to the processor, or a physical device that performs actions on the environment in accordance with the output of the

processor, e.g. a robot.

In order to communicate about the domain, a denotation of a model of that domain is needed. That is, the CISS must contain a domain model denotation (which therefore is an actand of the CISS). This domain model denotation may be built up and maintained by specific input messages to the processor, parts of it may be retrieved or modifications may be effected by other input messages. A domain model denotation is normally composed of an intensional model and an extensional model, for instance a "database schema" (containing types and rules) and a "database" (containing a population of these types obeying these rules).

The processor input and output messages are formulated in some language, or possibly several languages. They may be more or less universal or specific to the kind of domain. A language representation is required by the processor to interpret input messages and to formulate output messages correctly. A top view of an CISS and its users is illustrated in figure 3.10-1.

We will now take a closer look at the computerised information processor and its actions. These may be decomposed as follows:

- An *input processor* is responsible for checking incoming messages for syntax errors. For this, the definition of the processor language is needed. If there is an error, a "syntax error" message is issued and no further processing will be performed. Otherwise, the input process will distinguish requests for building up and maintaining the domain model denotation ("building requests") and request for retrieving parts of the domain model denotation and/or deriving specific output ("retrieval requests")
- A *retrieval processor* is responsible for retrieving parts of the domain model denotation and for derivations thereof. This process not only has access to the domain model denotation, but also to the language representation. Output are "retrieval messages", such as answers to retrieval requests.
- A *building processor* is responsible for building up and maintaining a logically consistent domain model denotation. The building processor analyses the building requests and finds out whether or not the requested changes to the domain model denotation would result in a model which is still logically consistent and in accordance with the meta-model of the language used. In order to achieve this, both the meta-model and, in general, the domain model denotation itself are needed. Access to the domain model denotation is already realised by the retrieval processor and need not be repeated. Therefore, the building processor merely issues an auxiliary retrieval request to the retrieval processor, using the answer to decide the consistency question. In case the result is negative, the building request is rejected as a whole and the domain model denotation remains unchanged. Otherwise, it will be modified according to the building request. An appropriate output building message will be issued, stating whether the building request was accepted or not, and - in case it was not - why not. Other output building messages are possible, such as specific reactions to specific changes of the domain model denotation (e.g. in a signalling system).

There are numerous possible criteria for classifying information systems and sub-systems thereof. We will list a number of possible classification criteria. We will also mention examples of specific subclasses of information systems or sub-systems thereof, fulfilling one or another classification criterion.

(a) Addressee-specific vs. addressee-unspecific communication

Addressee-specific communication means that a communication partner addresses messages directly to a well-determined set of other communication partners. Typical examples are:

- talking directly to people,
- talking via the telephone to people,
- sending ordinary mail (snail mail) to specific mail addresses,
- sending electronic mail (e-mail) to specific e-mail addresses,
- sending fax via the telephone.

Addressee-unspecific communication means that there is no well-determined set of addressees. Typical examples are:

- writing and publishing a book or an article,
- radio and television broadcasting.

In an information system, both, addressee-specific as well as -unspecific communication can occur.

(b) Transient vs. permanently stored messages

A transient message may be kept stored in the minds of the (human) sender or receiver, but it is not permanently stored on an artificial medium. A permanently stored message will be kept in an artificial storage medium for later use or re-use at least for a sufficient period of time. Those storage media are for example:

- a paper filing cabinet,
- a book shelf,
- a computer storage medium (magnetic disk, etc.).

In an information system, both kinds of messages can occur. If the messages represent important or repeatedly needed information, they will be permanently stored.

In the case of a CISS, the set of messages permanently stored as data, are called the database of the CISS. A database may be physically centralised, say, on the magnetic disks of the organisation's computer centre, or may be physically distributed over various interconnected computers, such that conceptually there is still one database, or may be physically distributed over various unconnected (e.g. personal) computers, such that the database is also conceptually distributed.

The communication supported by a CISS can be addressee-specific as well as -unspecific. For example, e-mail networks are CISSs supporting addressee-specific communication. They usually do not have a centralised database. If desired, a communication partner, i.e. a user, can store messages in his own personal database. A CISS supporting only addressee-specific communication is sometimes called a "communication system".

The database of a CISS supporting addressee-unspecific communication represents a body of knowledge which can be used or re-used for supporting all sorts of information-related functions within the organisation. There are users who keep that information up-to-date by updating the database whenever necessary. There are (the same or other) users who request the CISS to retrieve information from the database, or to provide information which is derived from the database in some specific way.

Such a database-oriented CISS, often called a database system or database management system, is usually implemented by taking a CISS shell (e.g. a generalised database management system) and filling it with organisation-specific data.

(c) Kinds of organisations

The kind of organisation, whether it is a hospital, a bank, a government, a trading company, a factory, a school, or a university, may have a strong influence on the kinds of information to be acquired and used, on the kinds of languages to be used for representing knowledge and formulating messages, etc. We speak of hospital information systems, banking information systems, etc. One can expect that for a hospital information system, information about patients, diagnoses, therapies, etc. is relevant, while for a banking information system information about bank customers, accounts, share markets, etc., is required.

The kind of organisation may also have an influence on the choice of the CISS shells.

How far these influences extend, cannot be stated in general terms, but must be investigated case by case.

(d) **Kinds of functions within an organisation**

An organisational system can be divided into sub-systems according to the various functions within the organisation. The information system of the organisation can be sub-divided accordingly. We can have, for example [Daf86]:

- an *input handling information sub-system*,
- a *production-oriented information sub-system*,
- an *output handling information sub-system*,
- a *maintenance-oriented information sub-system*,
- an *adaptive information sub-system*, and
- a *managerial information sub-system*.

These sub-systems can be further sub-divided, and in large organisations they usually are. Take for example managerial information sub-systems. Other terms used for them or for sub-systems thereof are: management information systems (MIS), decision support systems (DSS), management planning systems, etc. A possible sub-division is done according to the different managerial levels. Usually there are three of them:

- The *top managers* are responsible for strategic decisions and do high-level planning and organising.
- The *middle managers* are responsible for tactical decisions and do more detailed planning and organising, as well as directing and controlling lower-level managers.
- The *lower-level managers* are responsible for operational decisions and do directing and controlling of co-workers.

The kind of information needed by a manager to perform his tasks depends on his managerial level. While a lower-level manager needs detailed information about the on-going operation, a middle manager needs less detailed operational information and more information about organisational and planning issues. A top manager finally needs "high-level" information to help him doing strategic planning and taking strategic decisions.

It is often the case that the information needed on a higher managerial level can be derived from the information needed on a lower managerial level, that is, the former information is "condensed" or "aggregated" out of the latter one. But this is not always so. Think for example of a top manager who requires information about global market trends. This information is neither derived from the one on lower levels, nor necessarily needed there.

(e) **Degree of probabilistic behaviour**

Organisational systems are in general probabilistic. This holds for their information systems as

well. Information as such can be expressed in a reproducible and reliable manner, but the acquisition and use of information by human or other probabilistic actors is nevertheless probabilistic in nature. For example, human actors may make mistakes when acquiring or transferring information, they may mis-interpret or mis-use information, etc.

In the case of a CISS, however, we normally can expect that it behaves deterministically, unless there is a stochastic flaw in the underlying hardware. But it could be, that also a CISS might behave probabilistically, on purpose. Think for example of an expert who performs tasks using partially heuristics or random try-and-error methods. We now want to develop a CISS which is able to mimic the behaviour and methods of the expert to some degree, for the purpose of helping non-experts or semi-experts to perform these tasks in the same professional way as the expert would do. Those CISSs are usually called expert systems. If we now want to mimic a random try-and-error method in the CISS, we have to apply, for example, a random noise generator in conjunction with the computer, to achieve this probabilistic behaviour of the expert system.

(f) Reactive vs. autonomous behaviour

Like organisation systems, also their information systems are in general partially autonomous/reactive.

Most CISSs are however just reactive. You put in some request, and they will answer. This is in particular the case for plain retrieval systems, where facts about the relevant domain of communication are stored in a database. You can ask questions about the database, such that the answers are subsets of the database, or derivations thereof.

A little bit more sophisticated are signal systems. Here, the CISS gives a message (a signal) to the users whenever the database gets into a specific state due to an update. This looks like autonomous behaviour, but in fact it is not. Take as example a raw material stock keeping administration. The database contains the facts, how many units of specific raw materials are actually on stock, and how many units of these raw materials are used on average per week. The signal system gives a signal, whenever the actual number of units on stock of some raw material becomes smaller than, say, twice the average frequency of use in units per week, for the same raw material. This is not an autonomous behaviour, it is just a reaction to a specific update of the actual units of raw material on stock.

There are however also partially autonomous/reactive CISSs. Report systems, for example, produce regular (daily, monthly, yearly) reports about the current contents of the database, or about derivations thereof (e.g. statistics). This behaviour can be called autonomous, provided we call the built-in clock an autonomous device.

(g) Kinds of derivation

There are basically three kinds of derivation:

- by arithmetic operation,
- by the cardinality operation, and
- by logical deductions (inferences).

Arithmetic operations can naturally be applied to numerical data only, for all sorts of calculations and for the production of statistics. The results of arithmetic operations can be again subject to those operations.

The latter feature is for example applied in CISSs, which are used for forecasting. Those systems are called forecasting systems, dynamic planning systems, or dynamic simulation systems. The idea is to store numerical data (quantified assertions) about the current (and may be also the past) state of the domain of communication in the database. Furthermore, the CISS contains derivation rules which make use of arithmetic operations, and which specify how a state of the domain of communication at a point of time is derived from a state which is present a small time interval earlier. The iterative application of these derivation rules to the original database or the subsequent derivations will result in a set of curves representing a forecast of the future development of the set of quantified assertions of the domain of communication.

These derivation rules for forecasting systems can become quite complex. In fact, they are often a set of difference equations approximating differential equations which in turn model the possible future behaviour of a domain of communication. Those techniques have been not only applied in management information systems and decision support systems, but for instance also to predict the future of the planet earth and the human race (Club of Rome [e.g. Mea72]).

The cardinality operation is applied to a set of things and results in the cardinality of this set. Whenever we want to know how many items of a specific sort are stored in our database, we use this operation. The result of this operation, being an integer, can be subject to arithmetic operations.

Logical deductions can be applied to all sorts of databases. Here is a very simple example: Suppose you have a database containing a genealogy of your family, which is normally a set of father-child and mother-child relationships. If you want to know who is the sibling (brother or sister), uncle or aunt, nephew or niece of whom, you need derivation rules like:

- $(p1 \text{ is father of } p3 \wedge p1 \text{ is father of } p4 \wedge p2 \text{ is mother of } p3 \wedge p2 \text{ is mother of } p4) \Rightarrow (p3, p4 \text{ are siblings}).$

In simple databases, there are only relationships which are true propositions about the domain of communication. There are information systems where more sophisticated databases are needed, and where in particular the various logical junctions are used within the database itself. Think of a hospital information system where there is a database representing the diagnoses performed on patients by various physicians. For example, for the patient John three diagnoses were made by three different physicians:

- \neg John has anaemia,
- John has anaemia \vee beriberi \vee cancer,
- John has cancer \Rightarrow John has anaemia \vee beriberi.

By means of general logical deduction techniques we can find out whether questions like "Does John have cancer?" can be answered at all and if they can, what the answer is, and whether such databases are logically consistent or not.

CISSs with the distinct feature to perform all sorts of logical deduction are often called deductive database systems. Many expert systems also belong to that class of CISSs.

(h) Kinds of changes and degree of changeability

Organisations evolve. They have to adapt to their changing environment to survive. Laws relevant to the organisation may change. Markets may change, etc. The information system of an organisation has to adapt and change accordingly. We can identify three major categories of changes:

- (1) Change of the individual things in the *extensional* model of the domain of communication. For example, in a hospital information system, a new diagnosis is made of a patient, replacing an old one. Those changes do not affect the hospital rule that new diagnoses of patients must be recorded, replacing old ones.
- (2) Change of types of things and rules in the *intensional* model of the domain of communication. For example, the hospital management decides that in the future all diagnoses of patients must be performed not by single physicians, but by groups of physicians, and that all diagnoses must be time-stamped and kept for at least 10 years for reference and comparison.
- (3) Change of the *languages* used to represent information and to formulate messages. For example, the hospital management finds that due to the above changes (2), new sorts of retrieval request occur, asking for example for the time periods when a patient had a certain disease. It is found that it is cumbersome to formulate those requests with the currently used retrieval language. Thus the management decides to provide a new retrieval language which is better suited.

Changes of sort (1) usually occur daily, and are handled easily by most currently existing CISS shells. Changes of sort (2) usually occur less frequently, and are a problem for existing CISS shells. In those cases, often a new CISS has to be implemented, replacing the old one. This replacement has usually the odd side effect to interrupt the work in and the functioning of the organisation. Changes of sort (3) are even more difficult to handle with existing CISS shells. Usually, the shell itself has to be replaced by a new one.

We call the sort of changes (2), including (1), evolution of *first* order, and the sort of changes (3), evolution of *second* order.

3.11 Summary of Assumptions and Definitions

A coherent set of definitions has been presented, that relate the most important concepts used in connection with organisational information usage and communication. They cover computerised representation and manipulation for supporting these activities, being based on an analysis of the nature of such organisations and the human actors that populate them. It is accepted that a number of basic assumptions are made that cannot be further questioned.

A line of arguments is laid down that is felt to be useful in human interaction. If understood by the stakeholders in information system development and application, they will then have a better understanding of each other's concerns and the levels at which these are addressed.

An informal compilation of our conceptual framework, comprising all assumptions and explanatory definitions, follows underneath.

ASSUMPTIONS:

Assumption [a]:

The "world" exists, independent of our own existence, or of our cognitive and intellectual capabilities.

Assumption [b]:

Human beings are able to observe and perceive "parts" or "aspects" of the "world" (which we will call domains) with their senses, thus forming perceptions in their minds. Perceptions can be considered as specific patterns, generally changing in time.

Assumption [c]:

Human beings are able to form conceptions in their minds, as a result of current or past perception, by means of various cognitive or intellectual processes, such as recognition, characterisation, abstraction, derivation, and/or inner reflection. The collection of (relatively) stable and (sufficiently) consistent conceptions in a person's mind is called his or her knowledge.

Assumption [d]:

A perceived domain may be conceived as composed of identifiable components, which we call things; they may overlap, contain each other or relate to each other in whatever empirical way.

Assumption [e]:

Some things are conceived as having a static existence (states), while others are conceived as changes of some state (transitions). Hence, a perceived domain may be conceived as having an existence in a temporal context.

Assumption [f]:

Some transitions may be conceived as being performed or brought about by some active things, called actors. Such a transition, called an action, is performed by that actor on passive things, called actands. A rational action by an actor is said to be in pursuit of a goal. It is possible that the cause of a transition cannot be attributed to a specific actor, but merely comes about because of, or is triggered by, some aspect of the preceding state.

Assumption [g]:

Persons use representations to communicate their conceptions. These conceptions are represented in some language on some medium.

Assumption [h]:

For a society, it is of prime interest that any domain is represented such that agreement about it emerges. Without claim to presumed "existence" in any "real-world" sense of the term, the resulting representations everyone agrees with, will be called "inter-subjective reality".

Assumption [i]:

Stable conceptions may be formed in a person's mind - as a result of observing the state of some part of the conceived world or changes of such a state, or as a result of communication with other persons regarding it. Any such conception may be called knowledge about the world. The assumed existence of the (mental) conception is a metaphor for the experience of persons when entertaining (hopefully) meaningful communication.

Assumption [j]:

Persons view, experience and discuss representations at several semiotic levels, that is to say, they address, with varying emphasis: the physical and empirical aspects of a representation, the laws for expressing such a representation, its meaning, its use and effects, and the extent to which one can agree on and possibly derive commitments from whatever a statement about some thing might entail.

Assumption [k]:

Human actors may have in their minds clear, precise and unambiguous conceptions, models, which they may express in the form of a model denotation, in some appropriate formal or semi-formal language.

Assumption [l]:

When viewing some domain, human actors often conceive coherence in that domain or assert the presence of such coherence. The resulting model of that domain, made up of coherent elements, is called a system. Anything outside that domain will be considered belonging to the system environment.

Assumption [m]:

Organisations are characterised by (usually goal-oriented) arrangements between a number of parties, whereby (mostly co-operative and co-ordinated) action is enabled through (often systematic) communication, which entails exchange of information, involving meaningfully composed messages.

Assumption [n]:

An organisation is a grouping of actors, together with a collection of actands, such that (a) a common goal is pursued or some other characteristic coherence is displayed, and (b) interactions occur that are based on communication. Information is used in an organisation in the context of its functioning (actions), both internally and with respect to its environment, the society at large. Because of this social involvement, norms are meaningful directives for individuals in the organisation and the organisation as a whole.

Assumption [o]:

Organisational information usage derives from its operational practice, which may be present in the form of its organisational system (by the "business model"); within it, one finds embedded what may be described as a complex net of information flows in support of its actions; some of the information flows may be arranged within a computerised information sub-system.

DEFINITIONS:

Definition E1: Thing

A **thing** is any part of a conception of a domain (being itself a "part" or "aspect" of the "world"). The set of all things under consideration is the conception of that domain. •

Definition E2: Predicator, Predicated thing

A **predicator** is a thing, used to characterise or qualify other things, and assumed as being "atomic", "undividable" or "elementary". •

A **predicated thing** is a thing being characterised or qualified by at least one predicator. •

Definition E3: Relationship

A **relationship** is a special thing composed of one or several predicated thing(s), each one associated with one predicator characterising the role of that predicated thing within that relationship. •

Definition E4: Set membership, Elementary thing, Composite thing

A **set membership** is a special binary relationship between a thing (the set), characterised by the special predicator called 'has-element', and another thing, characterised by the special predicator called 'is-element-of'. •

An **elementary thing** is a thing, not being a relationship and not being characterised by the special predicator called 'has-element'. •

A **composite thing** is a thing, not being an elementary thing. •

Definition E5: Entity

An **entity** is a predicated thing as well as an elementary thing. •

Definition E6: Type, Population, Instance

A **type** of things is a specific characterisation (e.g. a predicate) applying to all things of that type. •

A **population** of a type of things is a set of things, each one fulfilling the characterisation determining that type. •

An **instance** of a type of things is an element of a population of that type. •

Definition E7: Transition, State, Pre-state, Post-state

A **transition** is a special binary relationship between two (partially or totally) different composite things, called the pre-state and the post-state of that transition, whereby at least one thing is element of the pre-state, but not of the post-state, or vice versa. •

A **state** is a composite thing, involved as pre-state or as post-state in some transition. No element of a state may be a transition, itself. •

The **pre-state** of a transition is the state valid before that transition, and is characterised by the special predicator 'before'. •

The **post-state** of a transition is the state valid after that transition, and is characterised by the special predicator 'after'. •

Shorthand notation:

transition: pre-state => post-state

Definition E8: State-transition structure

Given are the transitions $t_x: s_1 \Rightarrow s_2$ and $t_y: s_3 \Rightarrow s_4$. The following basic **state-**

transition structures exist in this case:

(1) Sequence:

sequ (t_x, t_y) is a sequence of transitions if s_3 is a subset of s_2 .

The resulting state-transition structure has s_1 as pre-state and s_4 as post-state.

Longer sequences are defined as follows:

sequ (t_x, t_y, t_z) follows from sequ (t_x, t_y) and sequ (t_y, t_z)

(2) Choice:

choice (t_x, t_y) is a choice of transitions if the intersection of s_1 and s_3 is not empty.

The result is either transition t_x or t_y , but not both.

(3) Concurrency:

concur (t_x, t_y) are concurrent transitions if the intersection of s_1 and s_3 is empty.

The result is (s_1 union s_3) => (s_2 union s_4). •

Definition E9: Composite transition

A **composite transition** is a state-transition structure with a unique pre-state and a unique post-state. •

Definition E10: Transition occurrence

A **transition occurrence** is a specific occurrence of a transition. A set of transition occurrences is subject to strict partial ordering. •

Definition E11: Relative time, Absolute time

The strict partial order imposed on the sets of all transition occurrences is called **relative time**. •

Absolute time may be determined by a clock that issues (assumedly) regular pulses (transition occurrences of the clock, or clock events). An absolute time value may be assigned to some transition occurrence, by comparing that transition occurrence with the successive (absolute-time-determining) clock events. A strictly ordered set of time points can be defined on the basis of these clock events, called time axis. •

Definition E12: Rule

A **rule** determines a set of permissible states and transitions in a specific context. In other terms, a rule governs a non-empty set of types of things by determining their permissible populations. •

Definition E13: Actor

An **actor** is a special thing conceived as being "responsible" or "responsive" and as being able to "cause" transitions, and is therefore part of their pre-states, and, if not "destroyed" or "consumed" by the transitions, also part of their post-states. •

Definition E14: Action, Composite action, Action occurrence, Co-action

An **action** is a transition involving a non-empty set of actors in its pre-state, and, if not "destroyed" or "consumed" by the action, in its post-state as well, and involving a non-empty or empty set of other things (actands) as part of its pre-state, and having a non-empty or empty set of other things (actands) in its post-state. •

A **composite action** is a composite transition with the same conditions as applying for the notion of action. •

An **action occurrence** is a transition occurrence with the same conditions as applying for

the notion of action. •

A **co-action** is a special action performed by more than one actor in a co-ordinated way, pursuing a common goal. •

Definition E15: Actand, Input actand, Output actand, Resource

An **actand** is a thing involved in the pre-state or post-state of an action, not considered as an actor for that action. •

An **input actand** is a part of the pre-state of an action, excluding the actors. •

An **output actand** is a part of the post-state of an action, excluding the actors. •

The pre-state of an action, i.e. the union of the set of actors and the set of input actands of that action, is called its **resources**. •

Definition E16: Action context

The **action context** of an action is a special, optional part of the pre-state of that action, qualifying the context or situation in which that action is performed, and determining or modifying at least one of its output actands. •

Definition E17: Goal, Goal-pursuing actor

The **goal** of an action is a special input actand of that action, pursued by the actors of that action and stating the desired output state intensionally. •

A **goal-pursuing actor** is an actor performing an action, who deliberately aims at a specific goal when involved in that action. •

Definition E18: Domain, Domain component, Domain environment

A **domain** comprises any "part" or "aspect" of the "world" under consideration. •

A **domain component** is any "part" or "aspect" of that domain. •

A **domain environment** is the "world" without that domain. •

Definition E19: Human actor, Perception, Perceiving action, Perceiver

A **human actor** is a responsible actor with the capabilities and liabilities of a normal human being, in particular capable of performing perceiving actions, conceiving actions and representing actions. •

A **perception** is a special actand resulting from an action whereby a human actor observes a domain with his senses, and forms a specific (static, non-time-varying, or dynamic, time-varying) pattern of visual, auditory or other sensations of it in his mind. •

A **perceiving action** is a special action of a human actor having a domain as input actand and a perception as output actand. •

A **perceiver** is a human actor involved in a perceiving action. •

Definition E20: Conception, Conceiving action, Conceiver, Conceiving context

A **conception** is a special actand resulting from an action whereby a human actor aims at interpreting a perception in his mind, possibly in a specific action context. •

A **conceiving action** is a special action of a human actor having a perception and possibly some action context as input actand(s) and a conception as output actand. •

A **conceiver** is a human actor involved in a conceiving action. •

A **conceiving context** is the action context of a conceiving action. •

Definition E21: Interpreting action, Interpreter, Interpreting context

An **interpreting action** is the sequence of a perceiving action performed on a domain, resulting in a perception of that domain, followed by a conceiving action performed on that perception, resulting in a conception. •

An **interpreter** is a human actor performing an interpreting action. •

An **interpreting context** is the action context of an interpreting action. •

Definition E22: Symbol, Alphabet, Symbolic construct, Language

A **symbol** is a special entity used as an undividable element of a representation in a language. •

An **alphabet** of a language is a non-empty and finite set of symbols. •

A **symbolic construct** is a non-empty and finite "arrangement" of symbols taken from an alphabet. In the one-dimensional case, an arrangement is just a sequence of symbols (a "sentence"). In the n-dimensional case ($n > 1$), it may be any arrangement of its constituting symbols in the n-dimensional space. Provided one considers the elements of arrangement (such as sequence) belonging to the alphabet, a symbolic construct is a non-empty and finite set of symbols. •

A **language** is a non-empty set of permissible symbolic constructs. The permissible symbolic constructs in a language are determined either extensionally by enumeration or intensionally by a set of rules. The rules of a language may be syntactic ("grammar") as well as semantic ("semantic rules"). •

Definition E23: Representation, Representing action, Representer, Representing context

A **representation** is a special actand describing some conception(s) in a language, resulting from an action whereby a human actor aims at describing his conception(s), possibly in a specific action context. •

A **representing action** is a special action of a human actor having a conception and possibly some action context as input actand(s) and a representation as output actand. •

A **representer** is a human actor involved in a representing action. •

A **representing context** is the action context of a representing action. •

Definition E24: Label, Reference

A **label** is a special entity being an elementary representation and used for referring to some conception in an elementary way. •

A **reference** is a special binary relationship between a conception and a representation used to refer to that conception. •

Definition E25: Semiotic level

The **semiotic level** of a representation is the aspect considered in representing it. The semiotic levels are: physical, empirical, syntactical, semantical, pragmatic, and social. •

Definition E26: Model, Model denotation

A **model** is a purposely abstracted, clear, precise and unambiguous conception. •

A **model denotation** is a precise and unambiguous representation of a model, in some appropriate formal or semi-formal language. •

Definition E27: Modelling action, Modeller

A **modelling action** is the sequence of a perceiving action performed on some domain, followed by a conceiving action on that perception, resulting in a model, and followed by a representing action on that model, resulting in a model denotation. •

A **modeller** is a human actor performing a modelling action. •

Definition E28: Intensional model, Extensional model

An **intensional model** is that part of a model comprising the possibilities and necessities of a domain only, i.e. the types and rules. •

An **extensional model** is that part of a model containing a specific population of the types in the corresponding intensional model, whereby this population must obey all rules determined in that intensional model. •

Definition E29: Meta-model

A **meta-model** is a model of the conceptual foundation of a language, consisting of a set of basic concepts, and a set of rules determining the set of possible models denotable in that language. •

Definition E30: System, System denotation, System component, System environment, System viewer, System representer

A **system** is a special model, whereby all the things contained in that model (all the system components) are transitively coherent, i.e. all of them are directly or indirectly related to each other form a *coherent* whole. A system is conceived as having assigned to it, as a whole, a specific characterisation (the so-called "*systemic properties*"). •

A **system denotation** is a precise and unambiguous representation of a system. •

A **system component** is a non-empty set of things being contained in that system. •

The **system environment** of a system is the set of all things not being contained in that system. •

A **system viewer** is a human actor perceiving and conceiving a domain as a system. A system viewer recognises the system, by its distinction from the system environment, by its coherence, and because of its systemic properties. •

A **system representer** is a human actor representing a system in some language. •

Definition E31: Dynamic system, Static system, Active system, Passive System, Open system, Closed System

A **dynamic system** is conceived as capable of undergoing change, i.e. some of the system components are transitions. •

A system that is not dynamic is called a **static system**. •

An **active system** is conceived as capable of doing something, i.e. some of the system components are actors performing actions on actands. •

A system is called a **passive system** if it is not active. •

An **open system** is conceived as one which may respond to external messages or triggers, i.e. there may be transitions within the system due external causes coming from the system environment. •

A system is called a **closed system** if the system environment cannot cause any transitions within the system. •

Definition E32: Sub-system

Any **sub-system** S' of a larger system S is a system, itself. The set of all sub-system components of S' is a proper sub-set of the set of all system components of S. •

Definition E33: Knowledge

Knowledge is a relatively stable and sufficiently consistent set of conceptions possessed by single human actors. •

Definition E34: Data

The term **data** denotes any set of representations of knowledge, expressed in a language. •

Definition E35: Message, Message transfer, Sender, Receiver

A **message** is data, transmitted by one actor (the sender) via a channel (a medium), and intended for a non-empty set of other actors (the receivers). •

A **message transfer** is a sequence of actions, the sending action by the sender and the receiving actions by the receivers, whereby the input actand of the sending action is the message to be sent, whereby the output actand of the sending action, being (in the simplest case) equal to the input actand of the receiving action, is the message on the channel, and whereby the output actand of the receiving action is the message received. •

A **sender** is an actor sending a message. •

A **receiver** is an actor receiving a message. •

Definition E36: Information

Information is the personal knowledge increment brought about by a receiving action in a message transfer, i.e. it is the difference between the conceptions interpreted from a received message and the personal knowledge before the receiving action. •

Definition E37: Communication

Communication is an exchange of messages, i.e. a sequence of mutual and alternating message transfers between at least two human actors, called communication partners, whereby these messages represent some knowledge and are expressed in languages understood by all communication partners, and whereby some amount of knowledge about the domain of communication and about the action context and the goal of the communication is made present in all communication partners. •

Definition E38: Shared knowledge

Shared knowledge is that knowledge of the individuals in a group of human actors, which they assume to be identical (or at least similar) to that of the others, as resulting from the negotiation process implicit in some communication. •

Definition E39: Organisational system, Norm

An **organisational system** is a special kind of system, being normally dynamic, active and open, and comprising the conception of how an organisation is composed (i.e. of specific actors and actands) and how it operates (i.e. performing specific actions in pursuit of organisational goals, guided by organisational rules and informed by internal and external communication), where its systemic properties are that it responds to (certain kinds of) changes caused by the system environment and, itself, causes (certain kinds of) changes in the system environment. •

Norms are socially agreed rules affecting and to a large extent directing the actions within an organisational system. •

Definition E40: Information system, Information system denotation

An **information system** is a sub-system of an organisational system, comprising the conception of how the communication- and information-oriented aspects of an organisation are composed (e.g. of specific communicating, information-providing and/or information-seeking actors, and of specific information-oriented actands) and how these operate, thus describing the (explicit and/or implicit) communication-oriented and information-providing actions and arrangements existing within that organisation. •

An information system denotation is a precise and unambiguous representation of an information system. •

Definition E41: Computerised information sub-system (CISS)

A **computerised information sub-system** is a sub-system of an information system, whereby all actions within that sub-system are performed by one or several computer(s). •

**Was sich überhaupt sagen läßt, läßt sich klar sagen;
und wovon man nicht reden kann, darüber muß man schweigen.
(What can be said at all, can be said clearly;
and whereof one cannot speak thereof one must be silent.)**

Ludwig Wittgenstein, Tractatus logico-philosophicus

4 INFORMATION SYSTEM CONCEPTS: A Formal Approach

This chapter presents most aspects of our conceptual framework in a formalised way. Since the emphasis here is on formalisation rather than on explanation, it is certainly advisable to read chapter 3 before this chapter. The order of appearance of the concepts is similar to the order in chapter 3, but not identical.

The presentation of our conceptual framework is organised as follows: Firstly, in section 4.1 the fundamental layer of the framework is presented on which all other layers are built on. This fundamental layer can be viewed as our underlying ontological view. Since we are, when setting up such a conceptual framework, in fact doing a modelling exercise par excellence, we have to choose a particular ontological view. For a more detailed discussion of those considerations see chapter 3.

The fundamental layer starts from our ontological assumption that the "world" is conceived by humans as consisting of things. Entities as well as relationships as well as sets of things are considered as things. Things may be subject to change (transitions, considered as things themselves). Transitions may have several transition occurrences, and transitions can be composed into composite transitions. Furthermore, classifications of things lead to the definition of types of things.

In section 4.2, the triple action, actor, and actand is defined as a specialisation in terms of concepts of the fundamental layer.

By specialising these concepts further, in section 4.3, cognitive and semiotic concepts are introduced. The notions of perceiving, conceiving and representing are reflected in our formal framework, and defined as specialisations of actions.

In section 4.4, system concepts are defined as the last step towards the definitions of organisational and information systems.

These latter definitions are given in section 4.5, which concludes the presentation of our formal specification.

Section 4.6 provides a summary of all primitives, axioms, definitions and functions, and includes cross-references to chapter 3, such that the reader can easily see which aspects of our conceptual framework are formalised and which ones are not.

For the (partial) formalisation of our framework, we use a *first-order-logic language*. This language is associated to the following relational structure **A**:

A = <A, {R_i}_i ∈ I >, where

A is a non-empty set of elements, the relational domain of \mathbf{A}

I is a set such that for each $i \in I$, $\lambda(i)$ is a positive integer, the arity of the relation R_i on A , i.e. $R_i \subseteq A^{\lambda(i)}$.

Remark: A relation of arity 1 is a subset of A .

We define the language $L(\mathbf{A})$ by:

- (1) its alphabet:
 - (i) individual variables $v_0, v_1, v_2, \dots, v_n, \dots$
 - (ii) $\lambda(i)$ -ary relational symbol R_i for each $i \in I$
 - (iii) logical connectives $\neg, \wedge, \vee, \Rightarrow, \Leftarrow, \Leftrightarrow, |=$
 - (iv) universal quantifier \forall
 - (v) existential quantifier \exists
 - (vi) uniqueness quantifier $\exists!$
 - (vii) brackets $()$
- (2) words of L ; they are of the following three kinds:
 - (i) terms: $\text{Term}(L)$ is the set of all individual variables.
 - (ii) atomic formulae: $\text{Atom}(L)$ are the words of the form $R_i(t_1, \dots, t_{\lambda(i)})$, where $i \in I, t_1, \dots, t_{\lambda(i)} \in \text{Term}(L)$.
 - (iii) well-formed formulae $\mathbf{F}(\mathbf{A})$ is the smallest set containing all atomic formulae and closed by connectors and quantifiers:
If $\phi, \psi \in \mathbf{F}(\mathbf{A})$ then $(\phi \wedge \psi), (\phi \vee \psi), \neg \phi, \forall v_i [\phi], \exists v_i [\phi] \in \mathbf{F}(\mathbf{A})$

If L is a first-order-logic language, \mathbf{A} is said to be a realisation of L if $L = L(\mathbf{A})$.

The purpose of this first-order-logic language is to express any propositions about a given structure \mathbf{A} . Individual variables are interpreted as ranging over A . Terms, after the assignment of elements of A , are interpreted as elements of A as well. Symbols of relations are interpreted as corresponding relations. Each formula ϕ is interpreted with respect to the assignment of its free variables as a proposition about \mathbf{A} .

We have chosen for the following style of concept definitions: We first mention the denotation of the set of elements we wish to define. Then we define this set, by stating which elements belong to this set.

Note that the formalisation found in this chapter is not a goal in itself. The framework has merely been partially formalised to detect ambiguities and inconsistencies in setting it up. Note that we will specify only very few constraining axioms in a formal way. Some others will be mentioned informally. It has not been our intention to provide a complete formalisation with all details. Furthermore, it is often a difficult philosophical question as to whether or not a particular constraining axiom holds for all possible "worlds", not only for the "real world" but also for "systems of belief", "fantasy worlds", etc. To a large extent, we purposely left those questions open.

Nevertheless, our formalisation may help explaining existing or future languages for specifying information systems, and relating them to each other [Oei95].

We will illustrate the framework by accompanying the formalisation with a few examples which are taken from the following, very simple, case:

- Subjects are taught by lecturers and examined by examiners.
- Students attend subjects, and receive some exam grades.

- A class consists of a number of students.

4.1 The Fundamental Layer

The fundamental layer is based on our ontological assumptions that any "parts or aspects of the world", i.e. any domain, may be conceived as a thing or as consisting of things, and that there may be changes.

Primitive P1: Thing

The set of all things is denoted by \mathbf{Z}

Things may be related to other things in various ways. For relating things, we need the notion of relationship.

Definition D1: Relationship

The set of all relationships is denoted by \mathbf{R} .

$$\mathbf{R} = \{r \in \mathbf{Z} \mid r \subseteq \mathbf{U} \wedge \mathbf{U} = \{\langle q, p \rangle \mid q, p \in \mathbf{Z}\} \wedge 1 = |r| < \infty\}.$$

Notes:

- A relationship is defined here as a non-empty and finite set of pairs, whereby each pair u is composed of a thing q (the predicated thing) and a thing p (the predicator). The predicator p characterises the predicated thing q within the relationship r .
- The cardinality of the set of predicated-thing-predicator pairs forming a relationship is one or greater. If one, it is called a unary relationship. If two or greater, it is called binary or ternary, etc.

On these grounds, we can now define the notions of predicator and predicated thing.

Definition D2: Predicator

The set of all predicators is denoted by \mathbf{P} .

$$\mathbf{P} = \{p \in \mathbf{Z} \mid \exists u \in \mathbf{U}, q \in \mathbf{Z} [u = \langle q, p \rangle]\}.$$

Definition D3: Predicated thing

The set of all predicated things is denoted by \mathbf{Q} .

$$\mathbf{Q} = \{q \in \mathbf{Z} \mid \exists u \in \mathbf{U}, p \in \mathbf{P} [u = \langle q, p \rangle]\}.$$

Note:

The intersection $\mathbf{P} \cap \mathbf{Q}$ is not necessarily empty, but in many practical cases, it is empty.

Examples of things in our sample case are the following predicated things, predicators and relationships:

Predicated things:

The specific persons P1, P2, P3; the specific subject S1; the specific exam grade E1; the attendance of person P1 in subject S1.

Predicators:

To be untaught (characterising a person); to be idle (characterising a person); to attend or having attended a subject (characterising a person); to be attended by a person (characterising a subject); to teach a subject (characterising a person); to be taught by a person (characterising a subject); to examine a subject (characterising a person); to be examined by a person (characterising a subject); yielding an exam grade (characterising an attendance); to be yielded by an attendance (characterising an exam grade).

Unary relationships:

- A specific person P1 is untaught: $\{\langle P1, \text{untaught} \rangle\}$;
- A specific person P2 is idle: $\{\langle P2, \text{idle} \rangle\}$.

Binary Relationships:

- A specific person P1 is or was attending a specific subject S1:
{<P1, attending>, <S1, attended-by>};
- A specific person P2 teaches a specific subject S1: {<P2, teaching>, <S1, taught-by>};
- A specific person P3 examines a specific subject S1: {<P3, examining>, <S1, examined-by>};
- The attendance of person P1 in subject S1 yields or yielded an exam grade E1:
{<{<P1, attending>, <S1, attended-by>}, yielding>, <E1, yielded-by>}.

The notion of set membership is necessary in our conceptual framework, and will be introduced in terms of the above concepts. A set membership is a special binary relationship that is useful to determine which things are elementary and which ones are composite. For defining set membership, we need to introduce special predicators:

Primitives P2: Set membership predicators

'has-element' $\in \mathbf{P}$ and 'is-element-of' $\in \mathbf{P}$ are special predicators characterising a set and an element of a set in the context of a set membership, respectively. •

Definition D4: Set membership

The set of all set memberships is denoted by **SM** .

$\mathbf{SM} = \{sm \in \mathbf{R} \mid sm = \{<q_1, \text{has-element}>, <q_2, \text{is-element-of}>\} \wedge q_1, q_2 \in \mathbf{Q} \wedge q_1 \neq q_2\}$. •

Notes:

- We may use the usual abbreviations:
 - $sm = (q_2 \in q_1)$;
 - $\{q_2, \dots\} = q_1$; or $\{q_2, \dots\} \subseteq q_1$;
 - $\wp(q_1)$ denotes the set of all subsets of q_1 .
 - $\wp_m(q_1)$ denotes the set of all sub-multi-sets of q_1 .
- Constraints apply to set memberships, e.g. for avoiding paradoxes. For instance, a thing being a set of things, may not be an element of that set itself, nor an element of the transitive closure of that set. We will introduce this axiom (A2) below, but for the rest we dispense here with further details and refer to the appropriate literature on set theory.

On these grounds, we can define the notion of elementariness.

Definition D5: Elementary thing

The set of all elementary things is denoted by **EZ** .

$\mathbf{EZ} = \{ez \in \mathbf{Z} \setminus \mathbf{R} \mid \neg \exists u \in \mathbf{U} [u = \langle ez, \text{has-element} \rangle]\}$. •

A predicator is assumed to be always elementary, which is expressed by the following axiom:

Axiom A1:

$\forall p \in \mathbf{P} [p \in \mathbf{EZ}]$. •

An entity is, in a specific context, considered to be an elementary thing. This does not mean that the domain we conceive as an entity, cannot be conceived as being divided into smaller parts, i.e. consisting of several other things, in a different context. What is considered as one entity in a specific context, may be divided into several entities or things in another context. That is, the level of "granularity" is context-dependent.

We assume furthermore that an entity is always characterised by at least one predicator, i.e. an entity is always a predicated thing.

Definition D6: Entity

The set of all entities is denoted by \mathbf{E} .

$$\mathbf{E} = \mathbf{EZ} \cap \mathbf{Q}.$$

We can now define composite things simply as non-elementary:

Definition D7: Composite thing

The set of all composite things is denoted by \mathbf{CZ} .

$$\mathbf{CZ} = \mathbf{Z} \setminus \mathbf{EZ}.$$

Note that a relationship is a composite thing.

Examples of entities, composite things and set memberships in our sample case are:

Entities:

- The specific persons P1, P2, P3; the specific subject S1; the specific exam grade E1.

Composite things:

- The already mentioned relationships;
- A set of entities: A specific class of students C1 consisting of persons P1, ...: $C1 = \{P1, \dots\}$.

Set membership:

- A specific person P1 is element of the class of students C1: $P1 \in C1$.

As we said, number of constraints apply to set memberships, for avoiding paradoxes etc. We state now the axiom that a thing being a set of things, may not be an element of that set itself, nor an element of the transitive closure of that set. For that purpose, let us introduce the functions Elementof and Elementsof determining the elements or the transitive closure of elements of a set, respectively:

Function F1: Elementof

Let Elementof: $\mathbf{CZ} \in \wp(\mathbf{Z})$ be a function from composite things to sets of things,

where $\text{Elementof}(cz) = \{z \in \mathbf{Z} \mid \exists sm \in \mathbf{SM}$

$$[sm = \langle cz, \text{has-element} \rangle, \langle z, \text{is-element-of} \rangle]\}$$

Function F2: Elementsof

Let Elementsof: $\mathbf{CZ} \rightarrow \wp(\mathbf{Z})$ be a function from composite things to sets of things,

where $\text{Elementsof}(cz) = \emptyset$ if $cz \in \mathbf{EZ}$,

else $\text{Elementsof}(cz) = \text{Elementof}(cz) \cup \{\text{Elementsof}(z) \mid z \in \text{Elementof}(cz)\}$.

The axiom is now:

Axiom A2:

$$\forall cz \in \mathbf{CZ} [cz \notin \text{Elementsof}(cz)].$$

For convenience in later definitions we define now the following auxiliary functions on things:

The function Predthingin determines the set of predicated things being involved in a relationship.

Function F3: Predthingin

Let Predthingin: $\mathbf{R} \rightarrow \wp(\mathbf{Q})$ be a function from relationships to sets of predicated things,

where $\text{Predthingin}(r) = \{q \in \mathbf{Q} \mid \exists p \in \mathbf{P} [\langle q, p \rangle \in r]\}$.

We can also define a recursive variant of this function to retrieve the "nested" predicated things within relationships.

Function F4: Predthingsin

Let $\text{Predthingsin}: \mathbf{Q} \rightarrow \wp(\mathbf{Q})$ be a function from predicated things to sets of predicated things,
 where $\text{Predthingsin}(r) = \emptyset$ if $r \in \mathbf{E}$,
 else $\text{Predthingsin}(r) = \text{Predthingin}(r) \cup \{\text{Predthingsin}(q) \mid q \in \text{Predthingin}(r)\}$. •

Things may change. To be able to express changes, we need the notion of transition. A transition is a special binary relationship between two sets of things, whereby the first one is called the pre-state and characterised by the special predicator ‘before’, and the second one is called the post-state and characterised by the special predicator ‘after’. Pre- and post-state of a transition must not be identical, but some elements of them may be the same, thus may remain unchanged. A transition must have a unique pre-state and a unique post-state.

In most practical cases, the elements of pre- and post-state are relationships. The relationships in which a specific predicated thing is involved in, may change. The predicated thing is in those cases said to be in different states during its existence, due to transitions.

There are other cases as well. A thing being part of the pre-state may disappear, i.e. does not appear in the post-state, or a thing not existing in the pre-state, may appear in the post-state.

None of the elements of a state may be or contain a transition.

Primitives P3: Transition predicators

‘before’ $\in \mathbf{P}$ and ‘after’ $\in \mathbf{P}$ denote special predicators for defining transitions, whereby the first one characterises the pre-state of a transition, and the second one the post-state. •

Definition D8: Transition

The set of all transitions is denoted by \mathbf{T} .

$\mathbf{T} = \{t \in \mathbf{R} \mid \exists s_b, s_a \in \mathbf{CZ} [t = \{ \langle s_b, \text{before} \rangle, \langle s_a, \text{after} \rangle \} \wedge s_b \neq s_a]\}$. •

Note:

We will henceforth use the abbreviation for transitions:

$t: s_b \Rightarrow s_a$.

Definition D9: State

The set of all states is denoted by \mathbf{S} .

$\mathbf{S} = \{s \in \mathbf{CZ} \setminus \mathbf{T} \mid \exists t \in \mathbf{T} [s \in \text{Predthingin}(t)]\}$. •

The following functions are defined to retrieve the (pre-)state before and (post-)state after a transition, respectively:

Function F5: Prestateof

Let $\text{Prestateof}: \mathbf{T} \rightarrow \mathbf{S}$ be a function from transitions to states,

where $\text{Prestateof}(t: s_b \Rightarrow s_a) = s_b$ denotes the (pre-)state before the transition t . •

Function F6: Poststateof

Let $\text{Poststateof}: \mathbf{T} \rightarrow \mathbf{S}$ be a function from transitions to states,

where $\text{Poststateof}(t: s_b \Rightarrow s_a) = s_a$ denotes the (post-)state after the transition t . •

Example:

The transition t_1 of a lecturer P_2 from being in an ‘idle’ state to the state of ‘teaching’ is given by:

$t_1: s_1 \Rightarrow s_2$, where

$s_1 = \text{Prestateof}(t_1) = \{ \langle P_2, \text{idle} \rangle \}$, and $s_2 = \text{Poststateof}(t_1) = \{ \langle P_2, \text{teaching} \rangle \}$.

Transitions can be related to each other, to form state-transition structures, in the following way:

Definition D10: State-transition structure

If $t_x: s_1 \Rightarrow s_2$, $t_y: s_3 \Rightarrow s_4$ are transitions, then the following basic state-transition structures exist:

(1) **Sequence:**

$\text{sequ}(t_x, t_y)$ if $s_3 \subseteq s_2$.

The resulting state-transition structure has s_1 as pre-state and s_4 as post-state.

Longer sequences are defined as follows:

$\text{sequ}(t_x, t_y, t_z) \Leftarrow \text{sequ}(t_x, t_y) \wedge \text{sequ}(t_y, t_z)$.

(2) **Choice:**

$\text{choice}(t_x, t_y)$ if $s_1 \cap s_3 \neq \emptyset$. The result is either transition t_x or t_y , but not both.

(3) **Concurrency:**

$\text{concur}(t_x, t_y)$ if $s_1 \cap s_3 = \emptyset$. The result is $(s_1 \cup s_3) \Rightarrow (s_2 \cup s_4)$.

Let **ST** denote the set of all state-transition structures formed by these rules. •

A state-transition structure with a unique pre-state and a unique post-state, thus being a transition, is called a composite transition.

Definition D11: Composite transition

The set of all composite transitions is denoted by **CT**.

$$\mathbf{CT} = \mathbf{ST} \cap \mathbf{T} .$$
 •

Note that by the recursive definition of a state-transition structure, very complex transitions can be formed.

From an initial state of a state-transition structure, a well-defined number of valid states are reachable by valid transitions within that structure.

It is important to distinguish transitions from their occurrences. There may be more than one transition occurrences belonging to the same transition. For example, a specific lecturer P2 may change more than once from the status 'idle' to the status 'teaching'. These transition occurrences belong to the transition of P2 from 'idle' to 'teaching'.

Definition D12: Transition occurrence

(1) A transition $t: s_1 \Rightarrow s_2$ is enabled to occur in state s if $s_1 \subseteq s$.

(2) If a transition $t: s_1 \Rightarrow s_2$ occurs in state s ,

then s is changed to the new state $s' = (s \setminus s_1) \cup s_2$. •

Note:

The occurrence of a transition $t: s_1 \Rightarrow s_2$ in state s leading to state s' is denoted as:

$$\text{occ}(t): s \rightarrow s'.$$

The choice construction in state-transition structures allows to describe non-deterministic structures, i.e. state-transition structures whose post-state is not determined. This is indispensable for a *prescriptive* representation of the potential behaviour of a set of things, i.e. for an *intensional* model. In essence, an intensional model deals with transitions or transition types but not with transition occurrences. However, if we have to describe the *actual*

behaviour of a set of things in retrospect, i.e. as an *extensional* model, then we are concerned with transition occurrences. Hence, all choices are resolved when they occur and have a determined post-state. Note however that the post-state of a choice may be different for different occurrences of the choice.

Given are two transitions $t_x: s_1 \Rightarrow s_2$ and $t_y: s_2 \Rightarrow s_3$. If the transition occurrence $\text{occ}(t_x): s \rightarrow s'$ occurs before the other transition occurrence $\text{occ}(t_y): s' \rightarrow s''$, then this situation is denoted as $(\text{occ}(t_x): s \rightarrow s') < (\text{occ}(t_y): s' \rightarrow s'')$. The symbol $<$ denotes here an ordering of the set of transition occurrences. Because of the concurrency state-transition structure, $<$ is a strict partial order. It allows to define relative time.

Definition D13: Relative time

The strict partial order ($<$) imposed on the set of all transition occurrences is called relative time. •

Usually it is said that a transition occurs at a certain point in time. Through the predicators ‘before’ and ‘after’, relative time is introduced. Absolute time, i.e. time points on a time axis have not been introduced yet. One could introduce time points as a fundamental concept. An alternative is to derive the concept of time points and a time axis from a sequence of transitions of a special entity, a clock. The choice depends on the philosophical question whether time implies change or the other way around. In our ontological view we have chosen to derive a time axis consisting of time points from the strict order of clock transition occurrences.

Definition D14: Absolute time

An entity of the type ‘clock’ is subject to sequences of transitions (clock pulses or clock events). A strictly ordered set of time points can be defined on the basis of these clock pulses, called time axis. •

Note that we are building up our conceptual framework basically from an extensional point of view. Whenever we speak of a composite or predicated thing, a relationship, an entity, etc., we mean an "instance", not a "type". Sometimes however, it is necessary to apply an intensional point of view, and to use the notion of type. A type of things is a specific characterisation (e.g. a predicate or a well-formed formula) applying to all things of that type. Related to this notion are the notions of population and instance. A population of a type of things is a set of things, each one fulfilling the characterisation determining that type. An instance of a type of things is an element of a population of that type.

Definition D15: Type, Population, Instance

The set of all types of predicated or composite things is denoted by **TY**.

$$\mathbf{TY} \subseteq \wp(\mathbf{Q} \cup \mathbf{CZ}).$$
 •

Note:

The above definition is an extensional one. In practice, types are most often defined intensionally. Intensionally speaking, a type is a predicate (a Boolean function) on $\mathbf{Q} \cup \mathbf{CZ}$. If the predicate is true for a specific predicated or composite thing, it belongs to that type.

The set of all populations of types of predicated or composite things is denoted by **PO**.

$$\mathbf{PO} = \{po \in \wp(\mathbf{Q} \cup \mathbf{CZ}) \mid \exists ty \in \mathbf{TY} [po \subseteq ty]\}.$$
 •

The set of all instances of types of predicated or composite things is denoted by **IN**.

$$\mathbf{IN} = \{in \in \mathbf{Q} \cup \mathbf{CZ} \mid \exists ty \in \mathbf{TY} [in \in ty]\}.$$
 •

Examples of simple types with one instance each:

- Entity type 'person', characterised by the junction of predicators 'untaught' or 'attending' or 'idle' or 'teaching' or 'examining'; Entity instance 'P1';
- Entity type 'student', characterised by the predicator 'attending' (a sub-type of 'person'); Entity instance 'P1';
- Predicated-thing type 'attendance', characterised by the predicator 'yielding'; Predicated-thing instance {<P1, attending>, <S1, attended-by>};
- Relationship type 'attendance', characterised by the predicators 'attending' and 'attended-by' within these relationships; Relationship instance {<P1, attending>, <S1, attended-by>};
- Relationship type 'attendance of student P1' (a sub-type of 'attendance'); {<P1, attending>, <S1, attended-by>};
- Transition type 'lecturer idle to teaching', characterised by the pre-state predicator 'idle' and the post-state predicator 'teaching'; Transition instance {<P2, idle>} \Rightarrow {<P2, teaching>}.

The last but not the least concept on the fundamental layer is "rule". A rule specifies the set of permissible states (static rule) or transitions of states (dynamic rule) within a conception of a domain.

Definition D16: Rule

The set of all rules is denoted by \mathbf{F} .

$$\mathbf{F} \subseteq \wp(\mathbf{S}) \cup \wp(\mathbf{T}).$$

Note:

The above definition is an extensional one. In practice, rules are most often defined intensionally. Intensionally speaking, a rule is a predicate (a Boolean function) on $\mathbf{S} \cup \mathbf{T}$. If the predicate is true, the state or transition in question is permissible, otherwise not. Instead of specifying what is permissible, it is often easier to specify what is forbidden, i.e. what the constraints are.

Examples of some simple rules are:

- A student can be either untaught or attending a subject, not both at the same time;
- A lecturer can be either idle or teaching a subject, not both at the same time;
- A student gets at most one exam grade per attendance;
- A student may repeat a subject at most twice;
- A subject is examined after being taught.

4.2 The Layer of Actors, Actions, and Actands

In many cases of discussing information system issues, it is useful not only to talk in terms of things, relationships, states and transitions, but also in terms of actions, actors and actands. The latter concepts can be defined in terms of the former ones.

In order to define actions, actors, and actands, we first have to introduce a special predicator:

Primitives P4: Actor-characterising predicators

Let 'performing' $\in \mathbf{P}$ denote the predicator indicating the capability of a thing to bring about or perform a transition.

Let 'performed-by' $\in \mathbf{P}$ denote the predicator indicating that a transition is performed by a thing.

An action is now defined as a special transition which is characterised by the actor-characterising predicators introduced: An action is a (simple or composite) transition $t: s_1 \Rightarrow s_2$, if there is at least one thing q in the pre-state s_1 performing transition t .

Definition D17: Action

The set of all actions is denoted by \mathbf{N} .

$$\mathbf{N} = \{t \in \mathbf{T} \mid \exists q \in \mathbf{Q} \exists r \in \mathbf{R}$$

$$[r = \{\langle q, \text{performing} \rangle, \langle t, \text{performed-by} \rangle\} \wedge q \in \text{Prestateof}(t)]\}. \quad \bullet$$

Since actions are transitions, the rules for forming state-transition structures apply here as well. Thus there may be structures of actions, which we call processes, and there may be composite actions.

Definition D18: Composite action

The set of all composite actions is denoted by \mathbf{CN} .

$$\mathbf{CN} = \mathbf{ST} \cap \mathbf{N}. \quad \bullet$$

The considerations about occurrences of transitions apply to actions as well.

The following function retrieves the actors of an action:

Function F7: Actorof

Let Actorof: $\mathbf{N} \rightarrow \wp(\mathbf{Q})$ be a function determining the actors performing an action,

where Actorof(n) = $\{q \in \mathbf{Q} \mid$

$$q \in \text{Prestateof}(n) \wedge \exists r \in \mathbf{R} [r = \{\langle q, \text{performing} \rangle, \langle n, \text{performed-by} \rangle\}]\}. \quad \bullet$$

An actor can now be defined as follows: A thing q is an actor iff there is an action n in which the thing q is an element of the pre-state of that action n and characterised by the predicator 'performing'.

Definition D19: Actor

The set of all actors is denoted by \mathbf{O} .

$$\mathbf{O} = \{q \in \mathbf{Q} \mid \exists n \in \mathbf{N} [q \in \text{Actorof}(n)]\}. \quad \bullet$$

The following functions retrieve elements of the pre- or post-states of an action, which are not actors. These elements are called actands, input or output actands, respectively:

Function F8: Inputof

Let Inputof: $\mathbf{N} \rightarrow \wp(\mathbf{Z})$ be a function determining the input actands of an action,

where Inputof(n) = $\{z \in \text{Prestateof}(n) \mid \neg(z \in \text{Actorof}(n))\}$. \bullet

Function F9: Outputof

Let Outputof: $\mathbf{N} \rightarrow \wp(\mathbf{Z})$ be a function determining the output actands of an action,

where Outputof(n) = $\{z \in \text{Poststateof}(n) \mid \neg(z \in \text{Actorof}(n))\}$. \bullet

Actands are defined as follows: An element of the pre- or post-state of an action n is an actand iff it is not an actor of that action n .

Definition D20: Actand

The set of all actands is denoted by \mathbf{D} .

$$\mathbf{D} = \{z \in \mathbf{Z} \mid \exists n \in \mathbf{N} [z \in \text{Inputof}(n) \vee z \in \text{Outputof}(n)]\}. \quad \bullet$$

Example:

Consider a teaching action $N1$ of the actor (lecturer) $P2$ teaching student $P1$, i.e.

$N1 = (\{\langle P1, \text{untaught} \rangle, \langle P2, \text{performing} \rangle\}) \Rightarrow (\{\langle P1, \text{taught} \rangle, \langle P2, \text{performing} \rangle\})$;

The lecturer $P2$ is the actor of the teaching action $N1$, i.e.

Actorof($N1$) = $P2$;

The untaught student $P1$ is the input actand of the teaching action $N1$, i.e.

Inputof($N1$) = $\{\langle P1, \text{untaught} \rangle\}$;

The taught student P1 is the output actand of the teaching action N1, i.e.
 Outputof (N1) = {<P1, taught>}.

The elements of the pre-state of an action are also called its resources. A resource of an action is any element of its pre-state, either an actor or an input actand.

Definition D21: Resource

The set of all resources is denoted by **RS** .

$$\mathbf{RS} = \{z \in \mathbf{Z} \mid \exists n \in \mathbf{N} [z \in \text{Prestateof}(n)]\}. \quad \bullet$$

The output actands of an action may depend on the context or situation in which that action is performed. We therefore introduce another concept, the "action context" of an action. It is a special, optional input actand of that action, qualifying the context or situation in which that action is performed, and determining or modifying at least one of its output actands. For capturing this notion formally, we need a special predicator:

Primitive P5: Context-characterising predicator

Let 'is-context' $\in \mathbf{P}$ denote the predicator indicating the characteristic of being a specific context. •

Definition D22: Action context

The set of action contexts is denoted by **X** .

$$\mathbf{X} = \{d \in \mathbf{D} \mid \exists n \in \mathbf{N} \exists r \in \mathbf{R} [r = \{<d, \text{is-context}>\} \wedge d \in \text{Inputof}(n)]\}. \quad \bullet$$

By defining an action as a transition, one describes what happens, but leaves out why that is the case. We wish to describe the purposefulness of an action by linking the notion of action to the notion of "goal". For introducing that notion, we need the following special predicator:

Primitives P6: Goal-pursuing predicators

Let 'pursued-by' $\in \mathbf{P}$ denote the predicator indicating the pursuit of a goal of an actor performing an action, to achieve the desired output state of that action. •

Let 'pursuing' $\in \mathbf{P}$ denote the predicator indicating an actor pursuing a goal. •

A goal is a special input actand of an action, specifying (intensionally) what is pursued by the actor of that action, i.e. what the desired output state is.

Definition D23: Goal

The set of goals is denoted by **G** .

$$\mathbf{G} = \{d \in \mathbf{D} \mid \exists n \in \mathbf{N} \exists o \in \mathbf{O} \exists r \in \mathbf{R}$$

$$[r = \{<d, \text{pursued-by}>, <o, \text{pursuing}>\} \wedge d \in \text{Inputof}(n) \wedge o \in \text{Actorof}(n)]\}. \quad \bullet$$

Note that an action occurrence is called *successful* if the goal of the action specified (intensionally) in its pre-state is equivalent to the post-state of that action occurrence.

4.3 The Layer of Cognitive and Semiotic Concepts

In the previous layer the concepts of actions, actors, and actands have been defined. In this layer these concepts are further specialised to define cognitive and semiotic concepts which are relevant for explaining the "semiotic tetrahedron" and the process of modelling. The notions of perceiving, conceiving and representing which were explained already in our basic assumptions in an informal way (see section 3.1), are reflected in our formal framework, and defined as specialisations of actions. Note that this is not a definitional loop, but rather an explanatory loop only.

The main actors considered on this layer are humans or actors with equivalent or similar cognitive capabilities.

To introduce these notions formally, we rely on a few special predicators which are considered primitive and not further explained here (see chapter 3):

Primitive P7: Human-characterising predicator

'is-human' $\in \mathbf{P}$ denotes the predicator characterising any human being or any being with equivalent capabilities and liabilities. •

Primitive P8: Domain-characterising predicator

'is-domain' $\in \mathbf{P}$ denotes the predicator characterising any domain. •

Primitive P9: Perception-characterising predicator

'is-perception' $\in \mathbf{P}$ denotes the predicator characterising any perception. •

Primitive P10: Conception-characterising predicator

'is-conception' $\in \mathbf{P}$ denotes the predicator characterising any conception. •

Primitive P11: Representation-characterising predicator

'is-repres' $\in \mathbf{P}$ denotes the predicator characterising any representation. •

On these grounds, we define the notions of human actor, and of perceiving, conceiving and representing action with the corresponding actands and actors, straightforward:

Definition D24: Human actor

$\mathbf{OH} = \{o \in \mathbf{O} \mid \exists r \in \mathbf{R} [r = \{<o, \text{is-human}>\}]\}$ denotes the set of all human actors. •

A perceiving action is an action having a domain as its input and a perception as its output. A human actor performing a perceiving action is called a perceiver.

Definition D25: Domain, Perception, Perceiving action, Perceiver

$\mathbf{DD} = \{dd \in \mathbf{D} \mid \exists r \in \mathbf{R} [r = \{<dd, \text{is-domain}>\}]\}$ denotes the set of all domains. •

$\mathbf{DP} = \{dp \in \mathbf{D} \mid \exists r \in \mathbf{R} [r = \{<dp, \text{is-perception}>\}]\}$ denotes the set of all perceptions. •

$\mathbf{NP} = \{n \in \mathbf{N} \mid \exists! dd \in \mathbf{DD} \exists! dp \in \mathbf{DP} [dd = \text{Inputof}(n) \wedge dp = \text{Outputof}(n)]\}$
denotes the set of all perceiving actions. •

$\mathbf{OP} = \{o \in \mathbf{OH} \mid \exists n \in \mathbf{NP} [o \in \text{Actorof}(n)]\}$ denotes the set of all perceivers. •

A conceiving action is an action having a perception as its input and a conception as its output. A human actor performing a conceiving action is called a conceiver. The corresponding context is called conceiving context.

Definition D26: Conception, Conceiving action, Conceiver, Conceiving context

- DC** = $\{dc \in \mathbf{D} \mid \exists r \in \mathbf{R} [r = \{\langle dc, \text{is-conception} \rangle\}]\}$ denotes the set of all conceptions. •
- NC** = $\{n \in \mathbf{N} \mid \exists! dp \in \mathbf{DP} \exists! dc \in \mathbf{DC} [dp = \text{Inputof}(n) \wedge dc = \text{Outputof}(n)]\}$
denotes the set of all conceiving actions. •
- OC** = $\{o \in \mathbf{OH} \mid \exists n \in \mathbf{NC} [o \in \text{Actorof}(n)]\}$ denotes the set of all conceivers. •
- XC** = $\{x \in \mathbf{X} \mid \exists n \in \mathbf{NC} [x \in \text{Inputof}(n)]\}$ denotes the set of all conceiving contexts. •

For the sake of convenience, we define now a special composite action, an interpreting action. An interpreting action is the sequence of a perceiving action performed on a domain, resulting in a perception of that representation, followed by a conceiving action performed on that perception, resulting in a conception. An interpreter is a human actor performing an interpreting action. An interpreting context is the context of the action of interpreting a domain.

Definition D27: Interpreting action, Interpreter, Interpreting context

- NI** denotes the set of all interpreting actions.
- NI** = $\{n \in \mathbf{N} \mid \exists np \in \mathbf{NP} \exists nc \in \mathbf{NC} [n = \text{sequ}(np, nc) \wedge \text{Inputof}(n) = \text{Inputof}(np) \wedge \text{Outputof}(np) = \text{Inputof}(nc) \wedge \text{Outputof}(nc) = \text{Outputof}(n) \wedge \text{Actorof}(n) = \text{Actorof}(np) = \text{Actorof}(nc)]\}$. •
- OI** = $\{o \in \mathbf{OH} \mid \exists n \in \mathbf{NI} [o \in \text{Actorof}(n)]\}$ denotes the set of all interpreters. •
- XI** = $\{x \in \mathbf{X} \mid \exists n \in \mathbf{NI} [x \in \text{Inputof}(n)]\}$ denotes the set of all interpreting contexts. •

A language is required for representing a conception. We will use the term ‘language’ only in the sense of *symbolic* language.

A (symbolic) language allows to express "sentences" or, more general, "symbolic constructs". The set of symbols used in the language is called its alphabet. The permissible symbolic constructs in that language are determined either extensionally by enumeration or intensionally by a set of rules.

A symbol is an elementary representation, i.e. the set of all symbols is the intersection of the set of all entities (in this particular context) and the set of all representations.

An alphabet of a language is a non-empty and finite set of symbols.

A symbolic construct is a non-empty and finite arrangement of symbols taken from an alphabet. In the one-dimensional case, an arrangement is just a sequence of symbols (a "sentence"). In the n-dimensional case ($n > 1$), it may be any arrangement of its constituting symbols in the n-dimensional space. Provided one considers the elements of arrangement (such as sequence) belonging to the alphabet, a symbolic construct is a non-empty and finite *set* of symbols.

The rules of a language (for determining the permissible symbolic constructs intensionally) may be syntactic ("grammar") as well as semantic.

A language is a non-empty set of permissible symbolic constructs.

Definition D28: Representation, Symbol, Alphabet, Symbolic construct, Language

- DR** = $\{dr \in \mathbf{D} \mid \exists r \in \mathbf{R} [r = \{\langle dr, \text{is-repres} \rangle\}]\}$ denotes the set of all representations. •
- SB** = $\mathbf{E} \cap \mathbf{DR}$ denotes the set of all symbols. •
- AB** = $\{ab \in \wp(\mathbf{SB}) \mid 1 = |ab| < \infty\}$ denotes the set of all alphabets. •
- SC** = $\{sc \in \wp_m(ab) \mid ab \in \mathbf{AB} \wedge 1 = |sc| < \infty\}$ denotes the set of all symbolic constructs. •
- L** denotes the set of all languages.

$$\mathbf{L} = \{l \in \wp(\mathbf{SC}) \mid \forall sc \in l \exists F \subseteq \mathbf{F} [sc \models F \wedge \exists! ab \in \mathbf{AB} [sc \in \wp(ab)]]\} \setminus \emptyset. \quad \bullet$$

Notes:

- The permissible symbolic constructs of a language can be determined in two ways:
 - (a) by enumerating all of them (determination by extension), or
 - (b) by defining a set of rules $F \subseteq \mathbf{F}$ on the set of symbolic constructs $l \in \wp(\mathbf{SC})$, where each $f (sc \in l) \in F$ holds iff sc is a permissible symbolic construct (determination by intension).
- These definitions hold for *any* kind of symbolic language of *any* dimension.

A representing action is an action having a conception as well as a language as its inputs and a representation as its output. A human actor performing a representing action is called a representer. The corresponding context is called representing context.

In this context we can also define the notions of reference and label. A reference is a special binary relationship between a conception and a representation. A label is an elementary representation used within a reference for referring to some conception. For defining these notions, we need the following primitives:

Primitives P12: Reference predicates

- ‘referred-to-by’ $\in \mathbf{P}$ denotes the predicate characterising a conception within a reference.
- ‘referring-to’ $\in \mathbf{P}$ denotes the predicate characterising a representation within a reference.

Definition D29: Representing action, Representer, Representing context, Reference,**Label**

NR denotes the set of all representing actions.

$$\mathbf{NR} = \{n \in \mathbf{N} \mid \exists! dc \in \mathbf{DC} \exists! l \in \mathbf{L} \exists! dr \in \mathbf{DR}$$

$$[dc = \text{Inputof}(n) \wedge l = \text{Inputof}(n) \wedge dr = \text{Outputof}(n)]\}. \quad \bullet$$

OR = $\{o \in \mathbf{OH} \mid \exists n \in \mathbf{NR} [o \in \text{Actorof}(n)]\}$ denotes the set of all representer. •

XR = $\{x \in \mathbf{X} \mid \exists n \in \mathbf{NR} [x \in \text{Inputof}(n)]\}$ denotes the set of all representing contexts. •

RF denotes the set of all references.

$$\mathbf{RF} = \{rf \in \mathbf{R} \mid \exists dc \in \mathbf{DC} \exists dr \in \mathbf{DR}$$

$$[rf = \{\langle dc, \text{referred-to-by} \rangle, \langle dr, \text{referring-to} \rangle\}]\}. \quad \bullet$$

LB denotes the set of all labels.

$$\mathbf{LB} = \{lb \in \mathbf{E} \cap \mathbf{DR} \mid \exists rf \in \mathbf{RF} \exists dc \in \mathbf{DC}$$

$$[rf = \{\langle dc, \text{referred-to-by} \rangle, \langle lb, \text{referring-to} \rangle\}]\}. \quad \bullet$$

We will now define special sub-notions of the notions of conception and representation, which are relevant in our context, model and model denotation, respectively. Models are clear, precise, unambiguous and purposely abstracted conceptions, and model denotations are representations of models. For capturing these features formally, we introduce a special predicate:

Primitive P13: Model-characterising predicate

- ‘is-model’ $\in \mathbf{P}$ denotes the predicate characterising the features of being clear, precise, unambiguous and purposely abstracted. •

Definition D30: Model

$$\mathbf{MC} = \{d \in \mathbf{DC} \mid \exists r \in \mathbf{R} [r = \{\langle d, \text{is-model} \rangle\}]\} \text{ denotes the set of all models.} \quad \bullet$$

The following auxiliary function is introduced for the sake of convenience:

Function F10: Concof

Let Concof: **DR**→**DC** be a function determining the conceptions behind representations.
 where Concof (dr) = (dc ∈ **DC** | ∃ n ∈ **NR** [dc = Inputof (n) ∧ dr = Outputof (n)]). •

Definition D31: Model denotation

MD denotes the set of all model denotations.

MD = {md ∈ **DR** | ∃ mc ∈ **MC** [mc = Concof (md)]}. •

A modelling action is a sequence of perceiving, conceiving and representing actions performed by the same human actor, who is called the modeller.

Definition D32: Modelling action, Modeller

MN denotes the set of all modelling actions.

MN = {n ∈ **N** | ∃ np ∈ **NP** ∃ nc ∈ **NC** ∃ nr ∈ **NR** [n = sequ (np, nc, nr) ∧

Inputof (n) = Inputof (np) ∧ Outputof (np) = Inputof (nc) ∧

Outputof (nc) = Inputof (nr) ∈ **MC** ∧ Outputof (nr) = Outputof (n) ∈ **MD** ∧

Actorof (n) = Actorof (np) = Actorof (nc) = Actorof (nr)]}. •

MOH = {o ∈ **OH** | ∃ n ∈ **MN** [o ∈ Actorof (n)]} denotes the set of all modellers. •

Let us now come to the final concepts of this section, knowledge, data, message, information, communication, and shared knowledge.

Knowledge is a relatively stable and sufficiently consistent set of conceptions. Relative stability does not mean that knowledge may never change. Sufficient consistency means that there may be some violations of consistency rules, which however do not severely affect the functioning of the persons in question. Since these criteria are fuzzy, we do not attempt to formalise them.

We distinguish between (personal) knowledge and knowledge shared by a group of people (shared knowledge). While personal knowledge can be easily defined in terms of the concepts we have already, the definition of shared knowledge is far more difficult and needs in particular the notion of communication as pre-requisite (see below).

Let us begin with the notion of (personal) knowledge which can be conveniently introduced via the following function:

Function F11: Knowlof

Let Knowlof: **OH**→**DC** be a function determining the knowledge of a human actor,

where Knowlof (oh) = {k ∈ \wp (**DC**) |

∃ ke ∈ Elementsof (k) ∃ nc ∈ **NC** [ke = Output (nc) ∧ oh = Actorof (nc)]}. •

Definition D33: Knowledge

K denotes the set of all (personal) knowledge.

K = {k ∈ \wp (**DC**) | ∃ oh ∈ **OH** [k = Knowlof (oh)]}. •

The pre-requisite for shared knowledge is the exchange of knowledge by means of message transfer and communication, on the basis of representations of knowledge. Any set of those representations is called data:

Definition D34: Data

DT ⊆ \wp (**DR**) denotes the set of all data. •

A message is composed of data, transmitted by one actor (the sender), and intended for a non-

empty set of other actors (the receivers). Message transfer can - in the simplest case - be viewed as a sequence of actions, the sending action by the sender and the receiving actions by the receivers, whereby the message between the sending action and the receiving action is said to be on a channel.

For defining the notions of message and message transfer, we introduce the following special predicates:

Primitives P14: Message-related predicates

‘sending’ $\in \mathbf{P}$ denotes the predicate characterising any sending action.

‘receiving’ $\in \mathbf{P}$ denotes the predicate characterising any receiving action. •

Definition D35: Sending action, Receiving action, Message transfer, Sender, Receiver, Message

MNS denotes the set of all sending actions.

$\mathbf{MNS} = \{mns \in \mathbf{N} \mid \exists r \in \mathbf{R} [r = \langle mns, \text{sending} \rangle]\} \wedge$
 $\text{Inputof}(mns) \in \mathbf{DT} \wedge \text{Outputof}(mns) \in \mathbf{DT} \wedge$
 $\forall i \in \text{Inputof}(mns) \forall o \in \text{Outputof}(mns) [i, o \in \mathbf{DT}]\}.$ •

MNR denotes the set of all receiving actions.

$\mathbf{MNR} = \{mnr \in \mathbf{N} \mid \exists r \in \mathbf{R} [r = \langle mnr, \text{receiving} \rangle]\} \wedge$
 $\text{Inputof}(mnr) \in \mathbf{DT} \wedge \text{Outputof}(mnr) \in \mathbf{DT} \wedge$
 $\forall i \in \text{Inputof}(mnr) \forall o \in \text{Outputof}(mnr) [i, o \in \mathbf{DT}]\}.$ •

MNT denotes the set of all message transfers.

$\mathbf{MNT} = \{mnt \in \mathbf{N} \mid \exists mns \in \mathbf{MNS}, mnr \in \mathbf{MNR} [mnt = \text{sequ}(mns, mnr) \wedge$
 $\text{Inputof}(mnt) = \text{Inputof}(mns) \wedge \text{Outputof}(mns) = \text{Inputof}(mnr) \wedge$
 $\text{Outputof}(mnr) = \text{Outputof}(mnt) \wedge \text{Actorof}(mns) \neq \text{Actorof}(mnr)]\}.$ •

MOS = $\{o \in \mathbf{O} \mid \exists mns \in \mathbf{MNS} [o = \text{Actorof}(mns)]\}$ denotes the set of all senders. •

MOR = $\{o \in \mathbf{O} \mid \exists mnr \in \mathbf{MNR} [o = \text{Actorof}(mnr)]\}$ denotes the set of all receivers. •

MDS denotes the set of all messages to be sent. •

$\mathbf{MDS} = \{dt \in \mathbf{DT} \mid \exists mnt \in \mathbf{MNT} [dt = \text{Inputof}(mnt)]\}.$ •

MDC denotes the set of all messages on a channel (between sender and receiver). •

$\mathbf{MDC} = \{dt \in \mathbf{DT} \mid \exists mns \in \mathbf{MNS} \exists mnr \in \mathbf{MNR}$
 $[dt = \text{Outputof}(mns) \wedge dt = \text{Inputof}(mnr)]\}.$ •

MDR denotes the set of all messages received. •

$\mathbf{MDR} = \{dt \in \mathbf{DT} \mid \exists mnt \in \mathbf{MNT} [dt = \text{Outputof}(mnt)]\}.$ •

MES = $\mathbf{MDS} \cup \mathbf{MDC} \cup \mathbf{MDR}$ denotes the set of all messages. •

On these grounds we can now define the notion of information. Information is the knowledge *increment* resulting from the receiving action in a message transfer. Thus information can be formally defined as the *difference* between the conceptions interpreted from a received message and the personal knowledge before the receiving action.

Definition D36: Information

I denotes the set of all information.

$I = \{i \in \wp(\mathbf{DC}) \mid \exists mnr \in \mathbf{MNR} \exists oh \in \mathbf{OH}$

$[i = \text{Concof}(\text{Outputof}(mnr)) \setminus \text{Knowlof}(oh) \wedge oh = \text{Actorof}(mnr)]\}$ •

To achieve genuinely shared knowledge, a single message transfer is often not sufficient. It is necessary to make sure that the conceptions represented by the sender are reasonably well understood by the receivers, that is that the sender and receivers have similar conceptions about the domain in question. This is the very purpose of communication.

Human actors are communicating if they exchange representations, such that one of the actors (in the role of the sender) is representing his conceptions in a language, and the other actors (in the role of the receivers) forms perceptions of these representations, forming conceptions thereof, and possibly answering (taking over the role of sender) by representing their own conceptions in the same or another language, which can in turn be perceived by the first actor (taking over the role of receiver), and so on. For successful communication, it is not only necessary to use languages which are understood by the respective receivers. Equally important is that the communication partners share already a certain amount of initial common knowledge about the domain, the context and the goal of the communication.

Definition D37: Communication

Communication is an exchange of messages, i.e. a sequence of mutual and alternating message transfers between at least two human actors, called communication partners, whereby these messages represent some knowledge and are expressed in languages understood by all communication partners, and whereby some amount of knowledge about the domain of communication and about the action context and the goal of the communication is made present in all communication partners. •

While a human actor cannot actually know what is in the minds of other people, the result of repeated communication may be that their conceptions about some domain stabilise to such an extent that they conceive the situation as one where their knowledge about it is the same. This is what we call shared knowledge:

Definition D38: Shared knowledge

Shared knowledge is that knowledge of the individuals in a group of human actors, which they assume to be identical (or at least similar) to that of the others, as resulting from the negotiation process implicit in some communication.

4.4 The Layer of System Concepts

Before we can define the specific kinds of systems we are interested in, we first have to define the general notion of system and some related concepts.

In our view, a system is a model (i.e. a specific conception) of a domain, called the system domain, where all the elements are interrelated such that they form a coherent whole. A system is conceived to have at least one particular systemic property that is not possessed by any of its elements.

A model, and thus also a system, is an output actand of a conceiving action by a conceiver. The structure of a system is thus given by the nature and composition of this output actand. In the case of a system, this actand is a *composite* thing, composed of a set of *transitively coherent relationships*. The conceiver of a system is called the system viewer.

Let us define first the notion of sets of (directly and transitively) coherent relationships. Two relationships R1 and R2 are *directly* coherent iff they have at least one predicated thing in common. Two relationships R1 and R3 are *transitively* coherent iff they are directly coherent or there exists a relationship R2 such that R1 and R2 as well as R2 and R3 are transitively coherent. A set of transitively coherent relationships may be represented as a fully connected graph.

Definition D39: Coherence of a set of relationships

RC denotes the set of all sets of directly coherent relationships.

$\mathbf{RC} = \{rc \in \wp(\mathbf{R}) \mid \exists rp \in \wp(\mathbf{R} \times \mathbf{R}) [\forall \langle r_x, r_y \rangle \in rp [r_x \in rc \wedge r_y \in rc \wedge r_x \neq r_y \wedge \text{Predthingsin}(r_x) \cap \text{Predthingsin}(r_y) \neq \emptyset]]\}$. •

\mathbf{RTC} denotes the set of all sets of transitively coherent relationships.

$\mathbf{RTC} = \{rtc \in \wp(\mathbf{R}) \mid \exists rc \in \mathbf{RC} [(\langle r_x, r_y \rangle \in rc \wedge \langle r_y, r_z \rangle \in rc) \Rightarrow \langle r_x, r_z \rangle \in rtc \wedge (\langle r_x, r_y \rangle \in rtc \wedge \langle r_y, r_z \rangle \in rtc) \Rightarrow \langle r_x, r_z \rangle \in rtc]\}$. •

On these grounds we can define the general notion of system as follows: A system is a model which in turn is a non-empty and finite set of transitively coherent relationships. It is thus a composite thing. It is furthermore also a predicated being contained in a non-empty set of relationships called the systemic properties of the system, whereby none of these systemic properties is an element of the system itself.

Definition D40: System

\mathbf{SY} denotes the set of all systems.

$\mathbf{SY} = \{sy \in \mathbf{MC}\} \wedge \{sy \in \mathbf{RTC} \mid \exists r_{sp} \in \mathbf{R} [sy \in \text{Predthingsin}(r_{sp}) \wedge r_{sp} \notin sy] \wedge 1 < |sy| \leq \infty\}$. •

To reduce complexity, it is often useful to decompose a system into sub-systems. A sub-system is a system whose elements are fully contained in another system (the super-system), but which does not contain all elements of that super-system.

Definition D41: Sub-system

\mathbf{SS} denotes the set of all system-sub-system pairs.

$\mathbf{SS} = \{\langle sy_b, sy_p \rangle \in \mathbf{SY} \times \mathbf{SY} \mid \text{Elementsof}(sy_b) \subset \text{Elementsof}(sy_p)\}$. •

Several classifications on systems can be made. We define three classifications of systems, which are relevant for defining organisational systems and information systems.

A dynamic system is a systems being conceived as containing at least one transition. A system is called static otherwise.

Definition D42: Dynamic System, Static system

\mathbf{SYD} denotes the set of all dynamic systems.

$\mathbf{SYD} = \{sy \in \mathbf{SY} \mid \exists t \in \mathbf{T} [t \in \text{Elementsof}(sy)]\}$. •

\mathbf{SYS} denotes the set of all static systems.

$\mathbf{SYS} = \mathbf{SY} \setminus \mathbf{SYD}$. •

An active system is a system being conceived as containing at least one action. A system is called passive otherwise.

Definition D43: Active system, Passive system

\mathbf{SYA} denotes the set of all active systems.

$\mathbf{SYA} = \{sy \in \mathbf{SY} \mid \exists n \in \mathbf{N} [n \in \text{Elementsof}(sy)]\}$. •

\mathbf{SYP} denotes the set of all passive systems.

$\mathbf{SYP} = \mathbf{SY} \setminus \mathbf{SYA}$. •

An open system is a system that contains a transition which may be caused by an external transition in the system environment. A system is called closed otherwise.

Definition D44: Open system, Closed system

\mathbf{SYO} denotes the set of all open systems.

$\mathbf{SYO} = \{sy \in \mathbf{SY} \mid \exists t_e, t_i \in \mathbf{T}$

- $[\text{sequ}(t_e, t_i) \wedge \text{Prestateof}(t_i) \subseteq \text{Poststateof}(t_e) \wedge t_i \in \text{sy} \wedge t_e \notin \text{sy}]$. •
- SYC** denotes the set of all closed systems. •
- SYC** = **SY** \ **SYO**. •

4.5 The Layer of Organisational and Information System Concepts

The layers of concepts defined in the previous sections allow us to define the notions of organisational and information system.

An organisational system is an open, dynamic and active system. It contains actors, actions and actands, and the relationships among these elements. An organisational system has the systemic property that its actors are pursuing at least one specific (organisational) goal.

Definition D45: Organisational system

OS denotes the set of all organisational systems.

$$\mathbf{OS} = \{os \in \mathbf{SYO} \cap \mathbf{SYD} \cap \mathbf{SYA} \mid \text{Elementsof}(os) \subseteq (\mathbf{O} \cup \mathbf{N} \cup \mathbf{D}) \wedge \exists g \in \mathbf{G} \exists r_{sp} \in \mathbf{R}$$

$$[r_{sp} = \{ \langle (\text{Elementsof}(os) \cap \mathbf{O}), \text{pursuing} \rangle, \langle g, \text{pursued-by} \rangle \} \}. \bullet$$

To define information systems we restrict ourselves to knowledge- and data-processing actions. Knowledge- and data-processing actions are actions having exclusively knowledge or data as input or output.

Definition D46: Knowledge/data-processing action

NKD denotes the set of all knowledge- and data-processing actions.

$$\mathbf{NKD} = \{n \in \mathbf{N} \mid \exists! d_{ik}, d_{ok} \in (\mathbf{K} \cup \mathbf{DT}) [d_{ik} = \text{Inputof}(n) \wedge d_{ok} = \text{Outputof}(n)]\}. \bullet$$

NDT denotes the set of all data-processing actions.

$$\mathbf{NDT} = \{n \in \mathbf{N} \mid \exists! d_{ik}, d_{ok} \in (\mathbf{DT}) [d_{ik} = \text{Inputof}(n) \wedge d_{ok} = \text{Outputof}(n)]\}. \bullet$$

Finally, we define the main subject of our area, i.e. information systems.

An information system is a sub-system of an organisational system. It contains actors, actions and actands, and the relationships among these elements, whereby all actions are knowledge- and data-processing actions. An information system has the systemic property that its actors are pursuing at least one specific (informational) goal.

Definition D47: Information system

IS denotes the set of all information systems.

$$\mathbf{IS} = \{is \in (\mathbf{SYO} \cap \mathbf{SYA}) \mid \text{Elementsof}(is) \subseteq (\mathbf{O} \cup \mathbf{NKD} \cup \mathbf{DT} \cup \mathbf{K}) \wedge \exists g \in \mathbf{G} \exists r_{sp} \in \mathbf{R}$$

$$[r_{sp} = \{ \langle (\text{Elementsof}(is) \cap \mathbf{O}), \text{pursuing} \rangle, \langle g, \text{pursued-by} \rangle \} \}. \bullet$$

Computerised information sub-systems have only computerised actors.

Primitive P15: Computer-characterising predicator

'is-comp' $\in \mathbf{P}$ denotes the predicator characterising an actor as a computer. •

A computerised actor is an actor characterised by the predicator 'is-comp'. We assume that the set of human actors and the set of computerised actors are disjoint.

Definition D48: Computerised actor
$$\mathbf{OCO} = \{o \in \mathbf{O} \setminus \mathbf{OH} \mid \exists r \in \mathbf{R} [r = \{<o, \text{is-comp}>\}]\}$$

denotes the set of all computerised actors. •

Since a computerised information sub-system has only computerised actors, the actands are only data.

Definition D49: Computerised information sub-system
$$\mathbf{CIS} = \{\text{cis} \in \mathbf{IS} \mid \text{Elementsof}(\text{cis}) \subseteq (\mathbf{OCO} \cup \mathbf{NDT} \cup \mathbf{DT})\}$$

denotes the set of all computerised information sub-systems. •

4.6 Summary of Primitives, Axioms, Definitions and Functions ¹

The following concepts, informally introduced in chapter 3 (and partially also mentioned in the previous sections), have not been formalised:

Relative time [E11, D13]
 Absolute time [E11, D14]
 Co-action [E14]
 Semiotic level [E25]
 Intensional model, Extensional model [E28]
 Meta-model [E29]
 Communication [E37, D37]
 Shared knowledge [E38, D38]
 Norm [E39]

The other concepts of our framework are formalised as follows:

Primitive P1: Thing [E1]

The set of all things is denoted by \mathbf{Z} •

Definition D1: Relationship [E3]

The set of all relationships is denoted by \mathbf{R} .

$$\mathbf{R} = \{r \in \mathbf{Z} \mid r \subseteq \mathbf{U} \wedge \mathbf{U} = \{\langle q, p \rangle \mid q, p \in \mathbf{Z}\} \wedge 1 = |r| < \infty\}. \bullet$$

Definition D2: Predicator [E2]

The set of all predicators is denoted by \mathbf{P} .

$$\mathbf{P} = \{p \in \mathbf{Z} \mid \exists u \in \mathbf{U}, q \in \mathbf{Z} [u = \langle q, p \rangle]\}. \bullet$$

Definition D3: Predicated thing [E2]

The set of all predicated things is denoted by \mathbf{Q} .

$$\mathbf{Q} = \{q \in \mathbf{Z} \mid \exists u \in \mathbf{U}, p \in \mathbf{P} [u = \langle q, p \rangle]\}. \bullet$$

Primitives P2: Set membership predicators

'has-element' $\in \mathbf{P}$ and 'is-element-of' $\in \mathbf{P}$ are special predicators characterising a set and an element of a set in the context of a set membership, respectively. •

Definition D4: Set membership [E4]

The set of all set memberships is denoted by \mathbf{SM} .

$$\mathbf{SM} = \{sm \in \mathbf{R} \mid sm = \{\langle q_1, \text{has-element} \rangle, \langle q_2, \text{is-element-of} \rangle\} \wedge q_1, q_2 \in \mathbf{Q} \wedge q_1 \neq q_2\}. \bullet$$

Abbreviations:

- $sm = (q_2 \in q_1)$;
- $\{q_2, \dots\} = q_1$; or $\{q_2, \dots\} \subseteq q_1$;
- $\wp(q_1)$ denotes the set of all subsets of q_1 .
- $\wp_m(q_1)$ denotes the set of all sub-multi-sets of q_1 .

Definition D5: Elementary thing [E4]

¹ **Legend:** The numbering refers to the primitives [Pn], axioms [An], definitions [Dn] and functions [Fn] in the previous sections, and to the explanatory definitions [En] in chapter 3.

The set of all elementary things is denoted by **EZ** .

$$\mathbf{EZ} = \{ez \in \mathbf{Z} \setminus \mathbf{R} \mid \neg \exists u \in \mathbf{U} [u = \langle ez, \text{has-element} \rangle]\}.$$

Axiom A1:

$$\forall p \in \mathbf{P} [p \in \mathbf{EZ}].$$

Definition D6: Entity [E5]

The set of all entities is denoted by **E** .

$$\mathbf{E} = \mathbf{EZ} \cap \mathbf{Q}.$$

Definition D7: Composite thing [E4]

The set of all composite things is denoted by **CZ** .

$$\mathbf{CZ} = \mathbf{Z} \setminus \mathbf{EZ}.$$

Function F1: Elementof

Let Elementof: $\mathbf{CZ} \in \wp(\mathbf{Z})$ be a function from composite things to sets of things,

where Elementof (cz) = {z ∈ Z | ∃ sm ∈ SM

$$[sm = \{\langle cz, \text{has-element} \rangle, \langle z, \text{is-element-of} \rangle\}]$$

Function F2: Elementsof

Let Elementsof: $\mathbf{CZ} \rightarrow \wp(\mathbf{Z})$ be a function from composite things to sets of things,

where Elementsof (cz) = ∅ if cz ∈ EZ,

else Elementsof (cz) = Elementof (cz) ∪ {Elementsof (z) | z ∈ Elementof (cz)}.

Axiom A2:

$$\forall cz \in \mathbf{CZ} [cz \notin \text{Elementsof} (cz)].$$

Function F3: Predthingin

Let Predthingin: $\mathbf{R} \rightarrow \wp(\mathbf{Q})$ be a function from relationships to sets of predicated things,

where Predthingin (r) = {q ∈ Q | ∃ p ∈ P [⟨q, p⟩ ∈ r]}.

Function F4: Predthingsin

Let Predthingsin: $\mathbf{Q} \rightarrow \wp(\mathbf{Q})$ be a function from predicated things to sets of predicated things,

where Predthingsin (r) = ∅ if r ∈ E,

else Predthingsin (r) = Predthingin (r) ∪ {Predthingsin (q) | q ∈ Predthingin (r)}.

Primitives P3: Transition predicates

‘before’ ∈ P and ‘after’ ∈ P denote special predicates for defining transitions, whereby the first one characterises the pre-state of a transition, and the second one the post-state.

Definition D8: Transition [E7]

The set of all transitions is denoted by **T** .

$$\mathbf{T} = \{t \in \mathbf{R} \mid \exists s_b, s_a \in \mathbf{CZ} [t = \{\langle s_b, \text{before} \rangle, \langle s_a, \text{after} \rangle\} \wedge s_b \neq s_a]\}.$$

Abbreviation: t: s_b ⇒ s_a.

Definition D9: State [E7]

The set of all states is denoted by **S** .

$$\mathbf{S} = \{s \in \mathbf{CZ} \setminus \mathbf{T} \mid \exists t \in \mathbf{T} [s \in \text{Predthingin} (t)]\}.$$

Function F5: Prestateof

Let $\text{Prestateof}: \mathbf{T} \rightarrow \mathbf{S}$ be a function from transitions to states,
 where $\text{Prestateof}(t: s_b \Rightarrow s_a) = s_b$ denotes the (pre-)state before the transition t . •

Function F6: Poststateof

Let $\text{Poststateof}: \mathbf{T} \rightarrow \mathbf{S}$ be a function from transitions to states,
 where $\text{Poststateof}(t: s_b \Rightarrow s_a) = s_a$ denotes the (post-)state after the transition t . •

Definition D10: State-transition structure [E8]

If $t_x: s_1 \Rightarrow s_2$, $t_y: s_3 \Rightarrow s_4$ are transitions, then the following basic state-transition structures exist:

(1) **Sequence:**

$\text{sequ}(t_x, t_y)$ if $s_3 \subseteq s_2$.

The resulting state-transition structure has s_1 as pre-state and s_4 as post-state.

Longer sequences are defined as follows:

$\text{sequ}(t_x, t_y, t_z) \Leftarrow \text{sequ}(t_x, t_y) \wedge \text{sequ}(t_y, t_z)$.

(2) **Choice:**

$\text{choice}(t_x, t_y)$ if $s_1 \cap s_3 \neq \emptyset$. The result is either transition t_x or t_y , but not both.

(3) **Concurrency:**

$\text{concur}(t_x, t_y)$ if $s_1 \cap s_3 = \emptyset$. The result is $(s_1 \cup s_3) \Rightarrow (s_2 \cup s_4)$.

Let \mathbf{ST} denote the set of all state-transition structures formed by these rules. •

Definition D11: Composite transition [E9]

The set of all composite transitions is denoted by \mathbf{CT} .

$\mathbf{CT} = \mathbf{ST} \cap \mathbf{T}$. •

Definition D12: Transition occurrence [E10]

(1) A transition $t: s_1 \Rightarrow s_2$ is enabled to occur in state s if $s_1 \subseteq s$.

(2) If a transition $t: s_1 \Rightarrow s_2$ occurs in state s ,
 then s is changed to the new state $s' = (s \setminus s_1) \cup s_2$. •

The occurrence of a transition $t: s_1 \Rightarrow s_2$ in state s leading to state s' is denoted as:

$\text{occ}(t): s \rightarrow s'$.

Definition D15: Type, Population, Instance [E6]

The set of all types of predicated or composite things is denoted by \mathbf{TY} .

$\mathbf{TY} \subseteq \wp(\mathbf{Q} \cup \mathbf{CZ})$. •

The set of all populations of types of predicated or composite things is denoted by \mathbf{PO} .

$\mathbf{PO} = \{\text{po} \in \wp(\mathbf{Q} \cup \mathbf{CZ}) \mid \exists \text{ty} \in \mathbf{TY} [\text{po} \subseteq \text{ty}]\}$. •

The set of all instances of types of predicated or composite things is denoted by \mathbf{IN} .

$\mathbf{IN} = \{\text{in} \in \mathbf{Q} \cup \mathbf{CZ} \mid \exists \text{ty} \in \mathbf{TY} [\text{in} \in \text{ty}]\}$. •

Definition D16: Rule [E12]

The set of all rules is denoted by \mathbf{F} .

$\mathbf{F} \subseteq \wp(\mathbf{S}) \cup \wp(\mathbf{T})$. •

Primitives P4: Actor-characterising predicators

Let ‘performing’ $\in \mathbf{P}$ denote the predicator indicating the capability of a thing to bring about or perform a transition. •

Let ‘performed-by’ $\in \mathbf{P}$ denote the predicator indicating that a transition is performed by a thing. •

Definition D17: Action [E14]

The set of all actions is denoted by \mathbf{N} .

$$\mathbf{N} = \{t \in \mathbf{T} \mid \exists q \in \mathbf{Q} \exists r \in \mathbf{R} [r = \{\langle q, \text{performing} \rangle, \langle t, \text{performed-by} \rangle\} \wedge q \in \text{Prestateof}(t)]\}. \bullet$$

Definition D18: Composite action [E14]

The set of all composite actions is denoted by \mathbf{CN} .

$$\mathbf{CN} = \mathbf{ST} \cap \mathbf{N}. \bullet$$

Function F7: Actorof

Let Actorof: $\mathbf{N} \rightarrow \wp(\mathbf{Q})$ be a function determining the actors performing an action,

where Actorof(n) = $\{q \in \mathbf{Q} \mid q \in \text{Prestateof}(n) \wedge \exists r \in \mathbf{R} [r = \{\langle q, \text{performing} \rangle, \langle n, \text{performed-by} \rangle\}]\}$. •

Definition D19: Actor [E13]

The set of all actors is denoted by \mathbf{O} .

$$\mathbf{O} = \{q \in \mathbf{Q} \mid \exists n \in \mathbf{N} [q \in \text{Actorof}(n)]\}. \bullet$$

Function F8: Inputof

Let Inputof: $\mathbf{N} \rightarrow \wp(\mathbf{Z})$ be a function determining the input actands of an action,

where Inputof(n) = $\{z \in \text{Prestateof}(n) \mid \neg(z \in \text{Actorof}(n))\}$. •

Function F9: Outputof

Let Outputof: $\mathbf{N} \rightarrow \wp(\mathbf{Z})$ be a function determining the output actands of an action,

where Outputof(n) = $\{z \in \text{Poststateof}(n) \mid \neg(z \in \text{Actorof}(n))\}$. •

Definition D20: Actand [E15]

The set of all actands is denoted by \mathbf{D} .

$$\mathbf{D} = \{z \in \mathbf{Z} \mid \exists n \in \mathbf{N} [z \in \text{Inputof}(n) \vee z \in \text{Outputof}(n)]\}. \bullet$$

Definition D21: Resource [E15]

The set of all resources is denoted by \mathbf{RS} .

$$\mathbf{RS} = \{z \in \mathbf{Z} \mid \exists n \in \mathbf{N} [z \in \text{Prestateof}(n)]\}. \bullet$$

Primitive P5: Context-characterising predicator

Let ‘is-context’ $\in \mathbf{P}$ denote the predicator indicating the characteristic of being a specific context. •

Definition D22: Action context [E16]

The set of action contexts is denoted by \mathbf{X} .

$$\mathbf{X} = \{d \in \mathbf{D} \mid \exists n \in \mathbf{N} \exists r \in \mathbf{R} [r = \{\langle d, \text{is-context} \rangle\} \wedge d \in \text{Inputof}(n)]\}. \bullet$$

Primitives P6: Goal-pursuing predicators

Let ‘pursued-by’ $\in \mathbf{P}$ denote the predicator indicating the pursuit of a goal of an actor performing an action, to achieve the desired output state of that action. •

Let ‘pursuing’ $\in \mathbf{P}$ denote the predicator indicating an actor pursuing a goal. •

Definition D23: Goal [E17]

The set of goals is denoted by \mathbf{G} .

$$\mathbf{G} = \{d \in \mathbf{D} \mid \exists n \in \mathbf{N} \exists o \in \mathbf{O} \exists r \in \mathbf{R} \bullet$$

$[r = \{<d, \text{pursued-by}>, <o, \text{pursuing}>\} \wedge d \in \text{Inputof}(n) \wedge o \in \text{Actorof}(n)]$. •

Primitive P7: Human-characterising predictor

‘is-human’ $\in \mathbf{P}$ denotes the predictor characterising any human being or any being with equivalent capabilities and liabilities. •

Primitive P8: Domain-characterising predictor

‘is-domain’ $\in \mathbf{P}$ denotes the predictor characterising any domain. •

Primitive P9: Perception-characterising predictor

‘is-perception’ $\in \mathbf{P}$ denotes the predictor characterising any perception. •

Primitive P10: Conception-characterising predictor

‘is-conception’ $\in \mathbf{P}$ denotes the predictor characterising any conception. •

Primitive P11: Representation-characterising predictor

‘is-repres’ $\in \mathbf{P}$ denotes the predictor characterising any representation. •

Definition D24: Human actor [E19]

$\mathbf{OH} = \{o \in \mathbf{O} \mid \exists r \in \mathbf{R} [r = \{<o, \text{is-human}>\}]\}$ denotes the set of all human actors. •

Definition D25: Domain, Perception, Perceiving action, Perceiver [E19]

$\mathbf{DD} = \{dd \in \mathbf{D} \mid \exists r \in \mathbf{R} [r = \{<dd, \text{is-domain}>\}]\}$ denotes the set of all domains. •

$\mathbf{DP} = \{dp \in \mathbf{D} \mid \exists r \in \mathbf{R} [r = \{<dp, \text{is-perception}>\}]\}$ denotes the set of all perceptions. •

$\mathbf{NP} = \{n \in \mathbf{N} \mid \exists! dd \in \mathbf{DD} \exists! dp \in \mathbf{DP} [dd = \text{Inputof}(n) \wedge dp = \text{Outputof}(n)]\}$ denotes the set of all perceiving actions. •

$\mathbf{OP} = \{o \in \mathbf{OH} \mid \exists n \in \mathbf{NP} [o \in \text{Actorof}(n)]\}$ denotes the set of all perceivers. •

Definition D26: Conception, Conceiving action, Conceiver, Conceiving context [E20]

$\mathbf{DC} = \{dc \in \mathbf{D} \mid \exists r \in \mathbf{R} [r = \{<dc, \text{is-conception}>\}]\}$ denotes the set of all conceptions. •

$\mathbf{NC} = \{n \in \mathbf{N} \mid \exists! dp \in \mathbf{DP} \exists! dc \in \mathbf{DC} [dp = \text{Inputof}(n) \wedge dc = \text{Outputof}(n)]\}$ denotes the set of all conceiving actions. •

$\mathbf{OC} = \{o \in \mathbf{OH} \mid \exists n \in \mathbf{NC} [o \in \text{Actorof}(n)]\}$ denotes the set of all conceivers. •

$\mathbf{XC} = \{x \in \mathbf{X} \mid \exists n \in \mathbf{NC} [x \in \text{Inputof}(n)]\}$ denotes the set of all conceiving contexts. •

Definition D27: Interpreting action, Interpreter, Interpreting context [E21]

\mathbf{NI} denotes the set of all interpreting actions.

$\mathbf{NI} = \{n \in \mathbf{N} \mid \exists np \in \mathbf{NP} \exists nc \in \mathbf{NC} [n = \text{sequ}(np, nc) \wedge \text{Inputof}(n) = \text{Inputof}(np) \wedge \text{Outputof}(np) = \text{Inputof}(nc) \wedge \text{Outputof}(nc) = \text{Outputof}(n) \wedge \text{Actorof}(n) = \text{Actorof}(np) = \text{Actorof}(nc)]\}$. •

$\mathbf{OI} = \{o \in \mathbf{OH} \mid \exists n \in \mathbf{NI} [o \in \text{Actorof}(n)]\}$ denotes the set of all interpreters. •

$\mathbf{XI} = \{x \in \mathbf{X} \mid \exists n \in \mathbf{NI} [x \in \text{Inputof}(n)]\}$ denotes the set of all interpreting contexts. •

Definition D28: Representation [E23], Symbol, Alphabet, Symbolic construct, Language [E22]

$\mathbf{DR} = \{dr \in \mathbf{D} \mid \exists r \in \mathbf{R} [r = \{<dr, \text{is-repres}>\}]\}$ denotes the set of all representations. •

$\mathbf{SB} = \mathbf{E} \cap \mathbf{DR}$ denotes the set of all symbols. •

AB = $\{ab \in \wp(\mathbf{SB}) \mid 1 = |ab| < \infty\}$ denotes the set of all alphabets. •

SC = $\{sc \in \wp_m(ab) \mid ab \in \mathbf{AB} \wedge 1 = |sc| < \infty\}$ denotes the set of all symbolic constructs. •

L denotes the set of all languages.

L = $\{l \in \wp(\mathbf{SC}) \mid \forall sc \in l \exists F \subseteq \mathbf{F} [sc \models F \wedge \exists! ab \in \mathbf{AB} [sc \in \wp(ab)]]\} \setminus \emptyset$. •

The permissible symbolic constructs of a language can be determined in two ways:

(a) by enumerating all of them (determination by extension), or

(b) by defining a set of rules $F \subseteq \mathbf{F}$ on the set of symbolic constructs $l \in \wp(\mathbf{SC})$, where each $f (sc \in l) \in F$ holds iff sc is a permissible symbolic construct (determination by intension).

Primitives P12: Reference predicates

‘referred-to-by’ $\in \mathbf{P}$ denotes the predicate characterising a conception within a reference. •

‘referring-to’ $\in \mathbf{P}$ denotes the predicate characterising a representation within a reference. •

Definition D29: Representing action, Representer, Representing context [E23],

Reference, Label [E24]

NR denotes the set of all representing actions.

NR = $\{n \in \mathbf{N} \mid \exists! dc \in \mathbf{DC} \exists! l \in \mathbf{L} \exists! dr \in \mathbf{DR}$

$[dc = \text{Inputof}(n) \wedge l = \text{Inputof}(n) \wedge dr = \text{Outputof}(n)]\}$. •

OR = $\{o \in \mathbf{OH} \mid \exists n \in \mathbf{NR} [o \in \text{Actorof}(n)]\}$ denotes the set of all representer. •

XR = $\{x \in \mathbf{X} \mid \exists n \in \mathbf{NR} [x \in \text{Inputof}(n)]\}$ denotes the set of all representing contexts. •

RF denotes the set of all references.

RF = $\{rf \in \mathbf{R} \mid \exists dc \in \mathbf{DC} \exists dr \in \mathbf{DR}$

$[rf = \{\langle dc, \text{referred-to-by} \rangle, \langle dr, \text{referring-to} \rangle\}]\}$. •

LB denotes the set of all labels.

LB = $\{lb \in \mathbf{E} \cap \mathbf{DR} \mid \exists rf \in \mathbf{RF} \exists dc \in \mathbf{DC}$

$[rf = \{\langle dc, \text{referred-to-by} \rangle, \langle lb, \text{referring-to} \rangle\}]\}$. •

Primitive P13: Model-characterising predicate

‘is-model’ $\in \mathbf{P}$ denotes the predicate characterising the features of being clear, precise, unambiguous and purposely abstracted. •

Definition D30: Model [E26]

MC = $\{d \in \mathbf{DC} \mid \exists r \in \mathbf{R} [r = \{\langle d, \text{is-model} \rangle\}]\}$ denotes the set of all models. •

Function F10: Concof

Let Concof: $\mathbf{DR} \rightarrow \mathbf{DC}$ be a function determining the conceptions behind representations.

where Concof(dr) = $(dc \in \mathbf{DC} \mid \exists n \in \mathbf{NR} [dc = \text{Inputof}(n) \wedge dr = \text{Outputof}(n)])$. •

Definition D31: Model denotation [E26]

MD denotes the set of all model denotations.

MD = $\{md \in \mathbf{DR} \mid \exists mc \in \mathbf{MC} [mc = \text{Concof}(md)]\}$. •

Definition D32: Modelling action, Modeller [E27]

MN denotes the set of all modelling actions.

MN = $\{n \in \mathbf{N} \mid \exists np \in \mathbf{NP} \exists nc \in \mathbf{NC} \exists nr \in \mathbf{NR} [n = \text{sequ}(np, nc, nr) \wedge$

$\text{Inputof}(n) = \text{Inputof}(np) \wedge \text{Outputof}(np) = \text{Inputof}(nc) \wedge$

$\text{Outputof}(nc) = \text{Inputof}(nr) \in \mathbf{MC} \wedge \text{Outputof}(nr) = \text{Outputof}(n) \in \mathbf{MD} \wedge$

Actorof (n) = Actorof (np) = Actorof (nc) = Actorof (nr)]}. •
MOH = {o ∈ **OH** | ∃ n ∈ **MN** [o ∈ Actorof (n)]} denotes the set of all modellers. •

Function F11: Knowlof

Let Knowlof: **OH** → **DC** be a function determining the knowledge of a human actor, where Knowlof (oh) = {k ∈ \wp (**DC**) |
 \forall ke ∈ Elementsof (k) ∃ nc ∈ **NC** [ke = Outputof (nc) ∧ oh = Actorof (nc)]}. •

Definition D33: Knowledge [E33]

K denotes the set of all (personal) knowledge. •
K = {k ∈ \wp (**DC**) | ∃ oh ∈ **OH** [k = Knowlof (oh)]}. •

Definition D34: Data [E34]

DT ⊆ \wp (**DR**) denotes the set of all data. •

Primitives P14: Message-related predicators

‘sending’ ∈ **P** denotes the predicator characterising any sending action. •
‘receiving’ ∈ **P** denotes the predicator characterising any receiving action. •

Definition D35: Sending action, Receiving action, Message transfer, Sender, Receiver, Message [E35]

MNS denotes the set of all sending actions. •
MNS = {mns ∈ **N** | ∃ r ∈ **R** [r = {<mns, sending>}]} ∧
Inputof (mns) ∈ **DT** ∧ Outputof (mns) ∈ **DT** ∧
 \forall i ∈ Inputof (mns) \forall o ∈ Outputof (mns) [i, o ∈ **DT**]}. •

MNR denotes the set of all receiving actions.

MNR = {mnr ∈ **N** | ∃ r ∈ **R** [r = {<mnr, receiving>}]} ∧
Inputof (mnr) ∈ **DT** ∧ Outputof (mnr) ∈ **DT** ∧
 \forall i ∈ Inputof (mnr) \forall o ∈ Outputof (mnr) [i, o ∈ **DT**]}. •

MNT denotes the set of all message transfers.

MNT = {mnt ∈ **N** | ∃ mns ∈ **MNS**, mnr ∈ **MNR** [mnt = sequ (mns, mnr) ∧
Inputof (mnt) = Inputof (mns) ∧ Outputof (mns) = Inputof (mnr) ∧
Outputof (mnr) = Outputof (mnt) ∧ Actorof (mns) ≠ Actorof (mnr)]}. •

MOS = {o ∈ **O** | ∃ mns ∈ **MNS** [o = Actorof (mns)]} denotes the set of all senders. •

MOR = {o ∈ **O** | ∃ mnr ∈ **MNR** [o = Actorof (mnr)]} denotes the set of all receivers. •

MDS denotes the set of all messages to be sent.

MDS = {dt ∈ **DT** | ∃ mnt ∈ **MNT** [dt = Inputof (mnt)]}. •

MDC denotes the set of all messages on a channel (between sender and receiver).

MDC = {dt ∈ **DT** | ∃ mns ∈ **MNS** ∃ mnr ∈ **MNR**
[dt = Outputof (mns) ∧ dt = Inputof (mnr)]}. •

MDR denotes the set of all messages received.

MDR = {dt ∈ **DT** | ∃ mnt ∈ **MNT** [dt = Outputof (mnt)]}. •

MES = **MDS** ∪ **MDC** ∪ **MDR** denotes the set of all messages. •

Definition D36: Information [E36]

I denotes the set of all information.

I = {i ∈ \wp (**DC**) | ∃ mnr ∈ **MNR** ∃ oh ∈ **OH**
[i = Concof (Outputof (mnr)) \ Knowlof (oh) ∧ oh = Actorof (mnr)]} •

Definition D39: Coherence of a set of relationships

RC denotes the set of all sets of directly coherent relationships.

$$\mathbf{RC} = \{rc \in \wp(\mathbf{R}) \mid \exists rp \in \wp(\mathbf{R} \times \mathbf{R}) [\forall \langle r_x, r_y \rangle \in rp [r_x \in rc \wedge r_y \in rc \wedge r_x \neq r_y \wedge \text{Predthingsin}(r_x) \cap \text{Predthingsin}(r_y) \neq \emptyset]]\}.$$

RTC denotes the set of all sets of transitively coherent relationships.

$$\mathbf{RTC} = \{rtc \in \wp(\mathbf{R}) \mid \exists rc \in \mathbf{RC} [(\langle r_x, r_y \rangle \in rc \wedge \langle r_y, r_z \rangle \in rc) \Rightarrow \langle r_x, r_z \rangle \in rtc \wedge (\langle r_x, r_y \rangle \in rtc \wedge \langle r_y, r_z \rangle \in rtc) \Rightarrow \langle r_x, r_z \rangle \in rtc]\}.$$

Definition D40: System [E30]

SY denotes the set of all systems.

$$\mathbf{SY} = \{sy \in \mathbf{MC}\} \wedge \{sy \in \mathbf{RTC} \mid \exists r_{sp} \in \mathbf{R} [sy \in \text{Predthingsin}(r_{sp}) \wedge r_{sp} \notin sy] \wedge 1 < |sy| \leq \infty\}.$$

Definition D41: Sub-system [E32]

SS denotes the set of all system-sub-system pairs.

$$\mathbf{SS} = \{\langle sy_b, sy_p \rangle \in \mathbf{SY} \times \mathbf{SY} \mid \text{Elementsof}(sy_b) \subset \text{Elementsof}(sy_p)\}.$$

Definition D42: Dynamic System, Static system [E31]

SYD denotes the set of all dynamic systems.

$$\mathbf{SYD} = \{sy \in \mathbf{SY} \mid \exists t \in \mathbf{T} [t \in \text{Elementsof}(sy)]\}.$$

SYS denotes the set of all static systems.

$$\mathbf{SYS} = \mathbf{SY} \setminus \mathbf{SYD}.$$

Definition D43: Active system, Passive system [E31]

SYA denotes the set of all active systems.

$$\mathbf{SYA} = \{sy \in \mathbf{SY} \mid \exists n \in \mathbf{N} [n \in \text{Elementsof}(sy)]\}.$$

SYP denotes the set of all passive systems.

$$\mathbf{SYP} = \mathbf{SY} \setminus \mathbf{SYA}.$$

Definition D44: Open system, Closed system [E31]

SYO denotes the set of all open systems.

$$\mathbf{SYO} = \{sy \in \mathbf{SY} \mid \exists t_e, t_i \in \mathbf{T}$$

$$[\text{sequ}(t_e, t_i) \wedge \text{Prestateof}(t_i) \subseteq \text{Poststateof}(t_e) \wedge t_i \in sy \wedge t_e \notin sy]\}.$$

SYC denotes the set of all closed systems.

$$\mathbf{SYC} = \mathbf{SY} \setminus \mathbf{SYO}.$$

Definition D45: Organisational system [E39]

OS denotes the set of all organisational systems.

$$\mathbf{OS} = \{os \in \mathbf{SYO} \cap \mathbf{SYD} \cap \mathbf{SYA} \mid \text{Elementsof}(os) \subseteq (\mathbf{O} \cup \mathbf{N} \cup \mathbf{D}) \wedge \exists g \in \mathbf{G} \exists r_{sp} \in \mathbf{R}$$

$$[r_{sp} = \{\langle (\text{Elementsof}(os) \cap \mathbf{O}), \text{pursuing} \rangle, \langle g, \text{pursued-by} \rangle\}]\}.$$

Definition D46: Knowledge/data-processing action

NKD denotes the set of all knowledge- and data-processing actions.

$$\mathbf{NKD} = \{n \in \mathbf{N} \mid \exists! d_{ik}, d_{ok} \in (\mathbf{K} \cup \mathbf{DT}) [d_{ik} = \text{Inputof}(n) \wedge d_{ok} = \text{Outputof}(n)]\}.$$

NDT denotes the set of all data-processing actions.

$$\mathbf{NDT} = \{n \in \mathbf{N} \mid \exists! d_{ik}, d_{ok} \in (\mathbf{DT}) [d_{ik} = \text{Inputof}(n) \wedge d_{ok} = \text{Outputof}(n)]\}.$$

Definition D47: Information system [E40]

IS denotes the set of all information systems.

$$\mathbf{IS} = \{is \in (\mathbf{SYO} \cap \mathbf{SYA}) \mid$$

$$\text{Elementsof}(is) \subseteq (\mathbf{O} \cup \mathbf{NKD} \cup \mathbf{DT} \cup \mathbf{K}) \wedge \exists g \in \mathbf{G} \exists r_{sp} \in \mathbf{R}$$

$$[r_{sp} = \{ \langle (\text{Elementsof}(is) \cap \mathbf{O}), \text{pursuing} \rangle, \langle g, \text{pursued-by} \rangle \} \}].$$

•

Primitive P15: Computer-characterising predicator

'is-comp' $\in \mathbf{P}$ denotes the predicator characterising an actor as a computer.

•

Definition D48: Computerised actor

$$\mathbf{OCO} = \{o \in \mathbf{O} \setminus \mathbf{OH} \mid \exists r \in \mathbf{R} [r = \{ \langle o, \text{is-comp} \rangle \}]\}$$

denotes the set of all computerised actors.

•

Definition D49: Computerised information sub-system [E41]

$$\mathbf{CIS} = \{cis \in \mathbf{IS} \mid \text{Elementsof}(cis) \subseteq (\mathbf{OCO} \cup \mathbf{NDT} \cup \mathbf{DT})\}$$

denotes the set of all computerised information sub-systems.

•



*Grau, teurer Freund, ist alle Theorie, und grün des Lebens goldner Baum.
(All theory, dear friend, is grey, but the golden tree of actual life springs ever green.)*

Johann Wolfgang von Goethe, Faust (Studierzimmer)

5 A SAMPLE APPLICATION OF OUR CONCEPTUAL FRAMEWORK

5.1 Description of the Test Case

As a small and simple test case, we chose "Business of Japan Wines, Inc." [ISO95]. It is described by a number of informal statements, as follows:

Business of Japan Wines, Inc. (JWI)

- S1: Japan Wines, Inc. is a wine distribution centre.
- S2: The business of this centre is to manage the inventory of products and to distribute products to retail shops corresponding to their orders.
- S3: All the orders received in a day are processed on the next day.
- S4: Every day the centre checks the inventory and places necessary orders to wineries to keep the inventory on a proper level.
- S5: This centre is not responsible for accounting businesses such as for product pricing, billing to retail shops and handling bills from wineries.
- S6: A more detailed description of the business is as follows:
- S7: The centre receives orders from retail shops by phone from 9:00 a.m. to 5:00 p.m.
- S8: Figure 5.1-1 shows a recording form of an order received from a retail shop.
- S9: An order may consist of many detailed items (referring to products).
- S10: Each detailed item is recorded in a line of the form.
- S11: When the centre receives an order, it immediately checks the inventory stock of each of the detailed items.
- S12: The centre takes back ordered products under agreement with retail shops.
- S13: When the centre accepts an order, each detailed order item is classified into one of two files: 'assigned ordered items' file and 'waiting ordered items' file.
- S14: If the centre has enough free stock of a detailed ordered item, it is recorded in the 'assigned ordered items' file and the free stock quantity of the corresponding item is updated properly.
- S15: Otherwise, it is recorded in the 'waiting ordered items' file.
- S16: Every day after 5:00 p.m., necessary orders to wineries and instructions of deliveries are produced as follows:
- S17: Orders to wineries are placed for those products whose free stock quantities are smaller than the minimum stock quantities such that their free stock quantities become the maximum stock quantities.
- S18: Figure 5.1-2 shows the order form to wineries.

- S19: The minimum and maximum stock quantities are defined for each product.
- S20: The centre provides delivery instruction tickets for each delivery truck by gathering the ordered items in the 'assigned ordered items' file, considering the destinations and the total amount of the orders for each item.
- S21: Figure 5.1-3 shows a delivery instruction ticket.
- S22: Next morning, each delivery truck picks up products from the warehouse according to the delivery instruction tickets and delivers them.
- S23: After delivery, it returns its delivery instruction tickets after marking either 'accomplished' or 're-deliver' to report the result of the delivery.
- S24: Products corresponding to 're-deliver' are brought back to the warehouse.
- S25: The centre cancels out each entry of the 'assigned ordered items' file by reflecting the 'accomplished' result of the returned delivery instruction tickets by 5 p.m.
- S26: The ordered items that become 're-deliver' remain unchanged and are included in the delivery instructions for the following day.
- S27: New products from wineries are arriving in the time from 10:00 a.m. to 4:00 p.m.
- S28: When new products are arriving, the centre assigns them to the detailed ordered items in the 'waiting ordered items' file in order of first-in first-out, re-classifies them into the 'assigned ordered items' file and updates the free stock quantities properly.

Order Form (from Retailer)

Date: _____
 Order Number: R _____
 Retail Shop: _____
 Phone: _____
 Location: _____

Item No.	Item	Quantity
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____

Figure 5.1-1: Order Form (from Retailer)

Order Form (to Winery)

Date: _____
 OrderNumber: W _____
 Winery: _____
 Phone: _____
 Location: _____

Item No.	Item	Quantity
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____
_____	_____	_____

Figure 5.1-2: Order Form (to Winery)

Delivery Instruction Ticket

Date: _____
 Delivery Number: D _____
 Truck No.: _____

Result (mark appropriate one):
 • accomplished _____
 • re-deliver _____

Order Number: R _____
 Retail Shop: _____
 Phone: _____
 Location: _____

Item No.	Item	Quantity
_____	_____	_____
_____	_____	_____
_____	_____	_____

Figure 5.1-3: Delivery Instruction Ticket

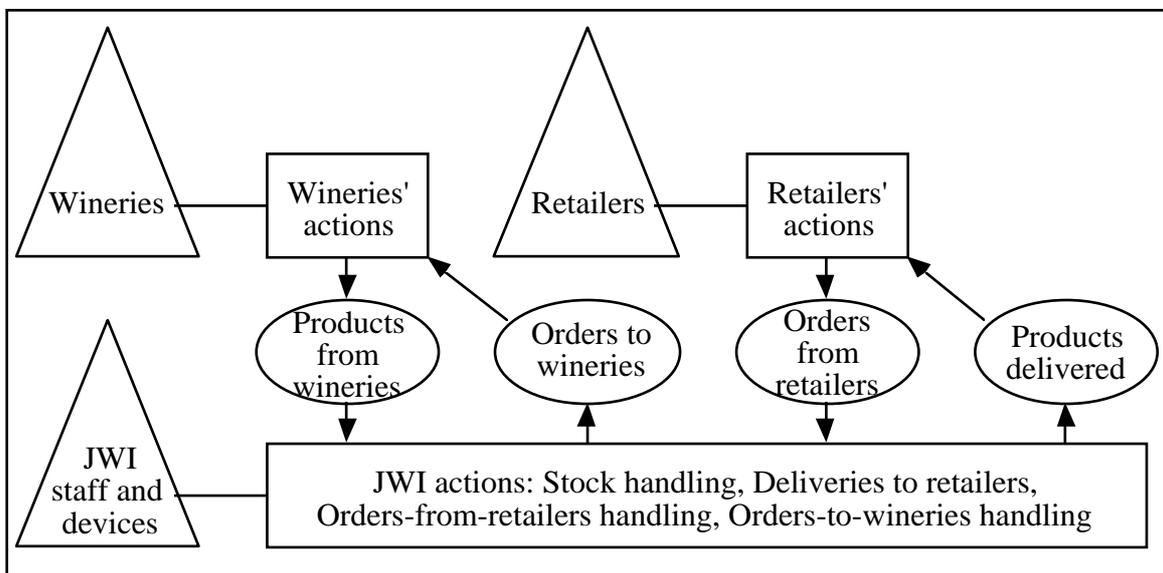
5.2 Interpretation of the Test Case in Terms of FRISCO

Given such an informal description, there may be various different interpretations as to what precisely are the things, entities, relationships, actions, actors, actands, etc. This is a general phenomenon observed in the context of modelling any domains, which we call "modelling uncertainty".

We could of course interpret this test case "on the fly", by going through our definitions and giving for each one examples from the test case. This would however have the distinctive disadvantage that cases of modelling uncertainty might unnecessarily blur the interpretations here and there.

To avoid this problem, we first do a modelling exercise and provide a *partial* model of the domain of this test case. Only then we will go through our list of definitions, give examples and refer, wherever necessary, to the statements of the test case description as well as to our partial model.

A mostly graphical denotation of our partial model is given in figures 5.2-1...5. Figure 5.2-1 gives an overview of the (composite) actions, actors and actands we conceived in this domain. Figure 5.2-2 provides a somewhat more detailed view of the actions, actors and actands of JWI, whereby the actands 'Products from wineries' and 'Products delivered' contain also data elements, such as 'Delivery papers from wineries' and 'Delivery papers to retailers'. Figure 5.2-3 shows an excerpt thereof, showing some of the knowledge- and data-oriented aspects of JWI. Finally, figures 5.2-4 and 5.2-5 shows the most important relationship types and a few populations thereof, contained within the data-oriented actands of JWI, in a tabular and graphical notation, respectively.



Legend: Triangles: Actors
 Rectangles: Actions
 Ovals: Actands
 Lines: Actors perform actions
 Arrows: Actands are input or output of actions

Figure 5.2-1: Overview of actions, actors and actands

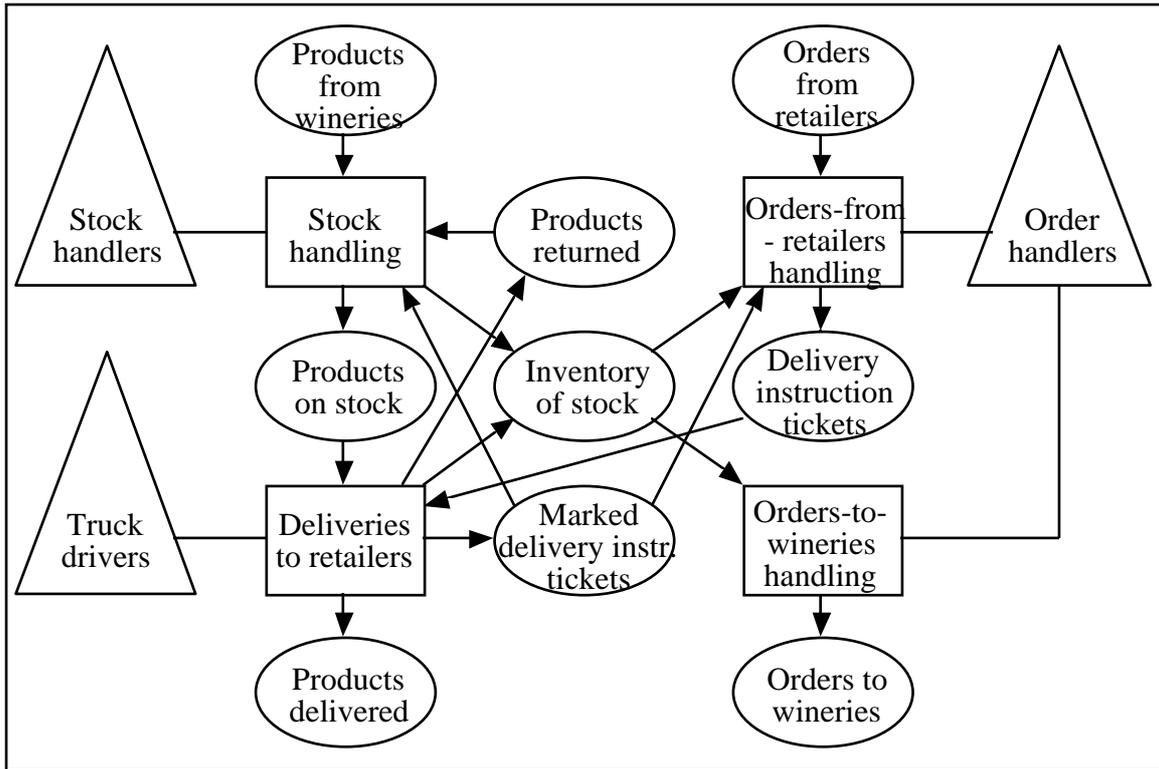


Figure 5.2-2: Actions, actors and actands of JW1 (the legend is the same as in figure 5.2-1).

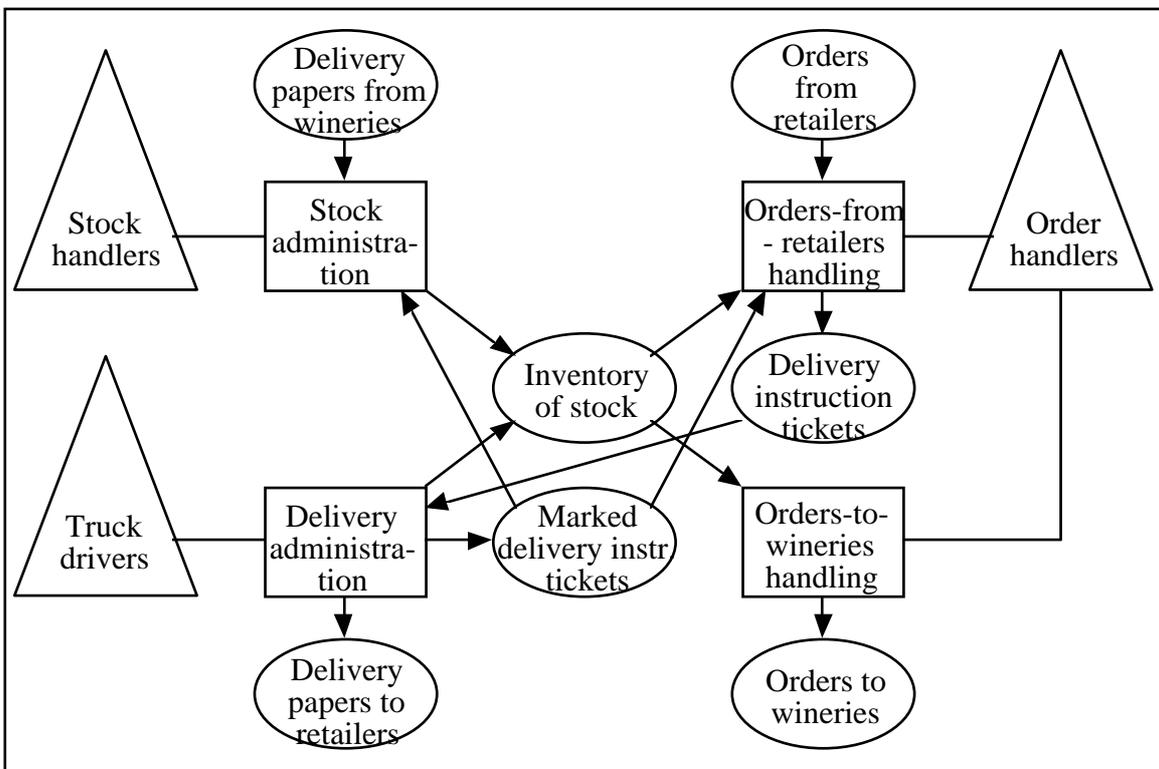


Figure 5.2-3: Knowledge- and data-oriented actions, actors and actands of JW1 (the legend is the same as in figure 5.2-1)

R1:

Product		Quantity
is on stock in		is on stock for
Item number	Item name	Integer
F-92-304	Fourtaney-1992	150

R2:

Product	Quantity
is maximum stock	is maximum for
Item number	Integer
F-92-304	1000

R3:

Product	Quantity
is minimum stock	is minimum for
Item number	Integer
F-92-304	100

R4:

Order	Product	Quantity
asks for	ordered in	of
Order number	Item number	Integer
R-96080016	F-92-304	200
D-96080021	F-92-304	150
W-96080011	F-92-304	1000

R5:

Order	Day
is placed on	when placing
Order number	Date (dd.mm.yy)
R-96080016	15.08.96
D-96080021	15.08.96
W-96080011	15.08.96

R6:

Order	Sort of order
is of	is assigned to
Order number	Sort code
R-96080016	from-retailer
D-96080021	delivery-to-retailer
W-96080011	to-winery

R7:

Company	Order
is placing or getting	is from or to
Company name	Order number
Hotaka Beverages	R-96080016
Hotaka Beverages	D-96080021
Chateau Fourtaney	W-96080011

R8:

Company	Phone
has	is of
Company name	Phone number
Hotaka Beverages	unknown
Chateau Fourtaney	+33-987654321

R9:

Company	Location
is at	is of
Company name	Address
Hotaka Beverages	Tokyo Japan
Chateau Fourtaney	Bordeaux France

R10:

Delivery order	Order from retailer
is executing	is executed by
Delivery number	Order number
D-96080021	R-96080016

R11:

Delivery order	Truck
is assigned to	is doing
Delivery number	Truck number
D-96080021	T-13

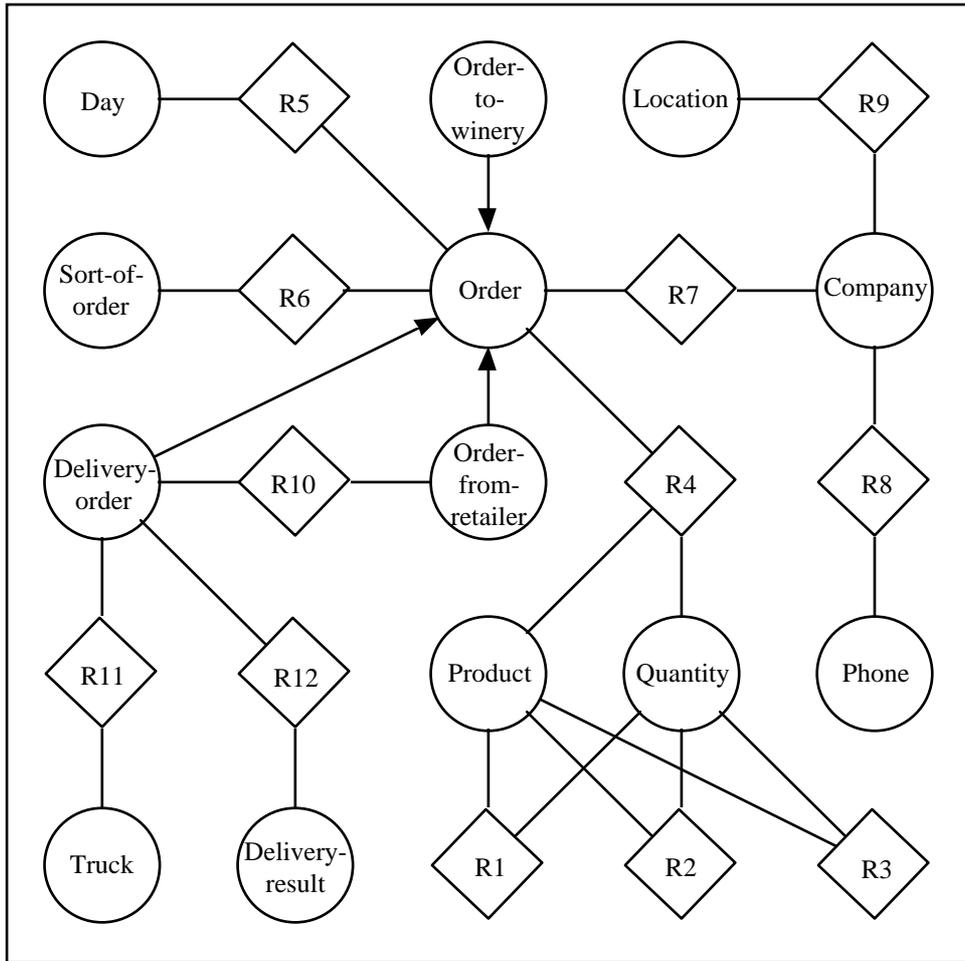
R12:

Delivery order	Delivery result
is achieving	is achieved by
Delivery number	Result code
D-96080021	accomplished

Legend: Three-layer table headers:
 1st layer: Predicated thing types, 2nd layer: Predicators, 3rd layers: Label types
 Table bodies: Small sample population

Note: 'Delivery order' and 'Order from retailer' denote two sub-types of the type 'Order'

Figure 5.2-4: Relationships within the actands shown in figure 5.2-3



Legend: Circles: Predicated thing types,
 Diamonds connected to circles: Relationships types
 Arrows: Sub-type indicators (sub-type -> type)

Note: Predicators are not shown here.

Figure 3.3-5: Graphical overview of the relationship types of figure 3.2-4

The following listing gives our interpretation of this case in terms of our conceptual framework. The numbering refers to the explanatory definitions [En] in chapter 3, and/or to the primitives [Pn], definitions [Dn] and functions [Fn] in chapter 4. The numbering [Sn] refers to the informal statements in the above section 5.1. Note that the listing is not meant to completely cover the test case. These are merely examples.

Thing [E1, P1]:

- All the items listed underneath are things.

Predicator [E2, D2]:

- Predicators assigned to products: ‘is on stock in’ (quantity), ‘is maximum stock’ (quantity), ‘is minimum stock’ (quantity), ‘is ordered in’ (quantity) (R1, R2, R3, R4 in figure 5.2-4) [S8, S17, S18, S19, S21].
- Predicators assigned to orders: ‘asks for’ (product in quantity), ‘is placed on’ (day), ‘is of’ (sort of order), ‘from or to’ (company) (R4, R5, R6, R7 in figure 5.2-4) [S8, S18, S21].
- Predicator assigned to trucks: ‘is doing’ (delivery order) (R11 in figure 5.2-4) [S22].

Predicated thing [E2, D3]:

- Specific products, quantities, orders, days, sorts of orders, companies, phones, locations, trucks, delivery results (R1 ... R12 in figure 5.2-4) are predicated things.

Relationship [E3, D1]:

- A specific product with item number 'F-92-304' and item name 'Fourtaney-1992', is on stock in quantity '150' (binary relationship in R1 in figure 5.2-4).
- A specific order with order number R-96080016 asks for a specific product with item number 'F-92-304' (and item name 'Fourtaney-1992') in quantity '200' (ternary relationship in R4 in figure 5.2-4).

Set membership [E4, P2, D4]:

- The inventory of stock of JWI has as elements all the instances of the relationship types R1, R2, R3 (in figure 5.2-4).

Elementary thing [E4, D5]:

- All the predicates above are elementary things.
- A specific product, e.g. a brand/vintage of wine, is an elementary thing in this context.
- A specific company (retail shop or winery) is an elementary thing in this context.

Composite thing [E4, D7, F1, F2]:

- All the relationships above are composite things.
- The inventory of stock of JWI is a composite thing.

Entity [E5, D6]:

- A specific product, e.g. the brand/vintage of wine with item number 'F-92-304' and item name 'Fourtaney-1992', is an entity in this context. Note that the individual bottle of wine is of no interest in this case.
- A specific company (retail shop or winery) is an entity in this context.

Type of things [E6, D15]:

- 'product', 'quantity', 'order', 'day', 'sort of order', 'company', 'phone', 'location', 'truck', 'delivery result' (R1 ... R12 in figure 5.2-4) denote types of (predicated) things.
- 'delivery order', 'order from retailer', 'order to winery' (R10, R11, R12 in figure 5.2-4, see also figure 5.2-4) denote sub-types of the type denoted by 'order'.

Population [E6, D15]:

- See table bodies of R1, ..., R12 in figure 5.2-4.

Instance [E6, D15]:

- See table bodies of R1, ..., R12 in figure 5.2-4.

State [E7, D9]:

- { 'Quantity on stock for product F-92-304 is 150', 'Quantity of product F-92-304 ordered by retailer is 200' } is a state s1.
- { 'Quantity on stock for product F-92-304 is 0', 'Quantity of product F-92-304 delivered to retailer is 150' } is a state s2.
- { 'Quantity on stock for product F-92-304 is 0', 'Quantity of product F-92-304 ordered from winery is 1000' } is a state s3.
- Note that these states are sets of relationships out of R1, R4, R6 in figure 5.2-4.

Transition [E7, P3, D8]:

- $tx: s_1 \Rightarrow s_2$, $ty: s_2 \Rightarrow s_3$ are two transitions.
- Each one may occur several times (several transition occurrences).
- tx belongs to a transition type called ‘delivery-stock balancing’.
- ty belongs to a transition type called ‘orders-to-wineries placing’.

Pre-state [E7, F5]:

- s_1 is pre-state of t_x , s_2 is pre-state of t_y .

Post-state [E7, F6]:

- s_2 is post-state of t_x , s_3 is post-state of t_y .

State-transition structure [E8, D10]:

- The sequence of t_x and t_y , $sequ(t_x, t_y)$, is a state transition structure.

Composite transition [E9, D11]:

- The sequence of t_x and t_y , $sequ(t_x, t_y)$, is also a composite transition $t: s_1 \Rightarrow s_3$.

Transition occurrence [E10, D12]:

- See under transition.

Relative time [E11, D13]:

- A specific transition occurrence $occ(t_x)$ is before $occ(t_y)$.

Absolute time [E11, D14]:

- A specific transition occurrence $occ(t_x)$ is on 15.08.96, 5.15 p.m.
- A specific transition occurrence $occ(t_y)$ is on 15.08.96, 6.00 p.m.

Rule [E12, D16]:

- An order is placed on exactly one day, an order is from or to exactly one company, etc., are constraints on single relationship types (R5, R7 in figure 5.2-4).
- The quantity on stock for a product may be between zero and the maximum stock quantity for that product. This is a constraint on two relationship types (R1, R2 in figure 5.2-4) [S17].
- When the quantity on stock for a product becomes smaller than the minimum stock quantity for that product, an order to the appropriate winery is placed, whereby the quantity of that product on that order is the maximum stock quantity minus the actual quantity on stock for that product. This is trigger rule involving four relationship types (R1, R2, R3, R4 in figure 5.2-4) and two transition types (‘delivery-stock balancing’ and ‘orders-to-wineries placing’) [S17].

Actor [E13, P4, D19]:

- A specific winery (e.g. ‘Chateau Fourtaney’), a specific retailer (e.g. ‘Hotaka beverages’), a specific staff member of JWI (e.g. the truck driver ‘R. Kawasaki’), a computer used for stock and delivery administration, are actors in this case (see figure 5.2-2 or 5.2-3).

Action [E14, D17]:

- Stock handling, stock administration, delivery to retailer, delivery administration, orders-from-retailers handling, orders-to wineries handling, are action types (see figures 5.2-2 and 5.2-3).

Composite action [E14, D18]:

- Since there are no choices in the structure of JWI actions, thus its output being unique, that structure is also a composite action type.

Action occurrence [E14]:

- A particular occurrence of stock handling, etc.

Co-action [E14]:

- Since the actions of the JWI staff are (presumably) performed in a co-ordinated way, they can - as a whole - be viewed as one co-action.

Actand [E15, D20]:

- Orders to wineries, products from wineries, delivery papers from wineries, orders from retailers, products on stock, inventory of stock, delivery instruction tickets, products delivered, delivery papers to retailers, marked delivery instruction tickets, products returned, are actand types in this case (see figures 5.2-1, 5.2-2 and 5.2-3).

Input actand [E15, F8]:

- The actand type 'products from wineries' is input of the action type 'stock handling'.

Output actand [E15, F9]:

- The actand type 'products on stock' is output of the action type 'stock handling'.

Resource [E15, D21]:

- The actor type 'stock handlers' and the actand type 'products from wineries' are the resource types of the action type 'stock handling'.

Action context [E16, P5, D22]:

- The context of JWI business conventions applies to all JWI actions.

Goal [E17, P6, D23]:

- The staff of JWI as a whole has the goal to buy products from wineries, to manage the stock and inventory of these products, and to distribute them to retail shops corresponding to their orders, such that maximum profit is achieved.
- A goal of a truck driver is to deliver the ordered products to the retail shops reliably and fast, such that the retailers are satisfied with the services of JWI.

Goal-pursuing actor [E17]:

- The staff of JWI and a specific truck driver are goal-pursuing (composite or elementary) actors in this context.

Domain [E18, P8, D25]:

- The wine distribution centre [S1] and its direct surrounding, in particular the wineries [4] and the retail shops [4], may be considered as the conceived domain of this test case. Conceived domain components are for instance the centre itself, the wineries and the retail shops.

Human actor [E19, P7, D24]:

- A specific staff member of JWI (e.g. 'R. Kawasaki') is a human actor.

Perception [E19, P9, D25]:

- The "physical" pattern of letters you see when looking at the text and figures of section 5.1, is a perception (of some representation in this case).

Perceiving action [E19, D25]:

- The viewing of this pattern, forming a perception, is a perceiving action of yours.

Perceiver [E19, D25]:

- When viewing this pattern, you are a perceiver.

Conception [E20, P10, D26]:

- A conception formed out the perception of the text of section 5.1 is, e.g., all the things, predicates, relationships, states, transitions, rules, actors, actions, actands, goals etc. mentioned above. Your own conception of the test case may be the same, similar, or different.

Conceiving action [E20, D26]:

- Your interpretation of that perception is a conceiving action of yours.

Conceiver [E20, D26]:

- When forming that conception, you are a conceiver.

Conceiving context [E20, D26]:

- The context of the conceiving action by the author of this section 5.2 was to illustrate our conceptual framework. Your own context may be quite different.

Interpreting action [E21, D27]:

- The sequence of the above perceiving and conceiving action is an interpreting action.

Interpreter [E21, D27]:

- When doing that interpreting action, you are an interpreter.

Interpreting context [E21, D27]:

- See conceiving action.

Symbol [E22, D28]:

- The letters a, ..., z are symbols.
- The ovals, rectangles, triangles, lines and arrows in figures 5.2-1, 5.2-2 and 5.2-3, are symbols.

Alphabet [E22, D28]:

- The set of symbols {a, ..., z} is an alphabet.
- The set of symbols {oval, rectangle, triangle, line, arrow} is an alphabet for a graphical language for expressing action-actor-actand models.

Symbolic construct [E22, D28]:

- The sentence 'This is a symbolic construct.' is a symbolic construct (a sequence of letters, a sentence) in natural language.
- The arrangements of ovals, rectangles, triangles, lines and arrows in figures 5.2-1, 5.2-2 and 5.2-3, are symbolic constructs of a graphical language.

Language [E22, D28]:

- For the description of the test case (section 5.1), natural language plus some tabular language (figures 5.1-1, 5.1-2 and 5.1-3) are used. For its interpretation in our terms (this section), we are using natural language plus two graphical languages and one tabular language (figures 5.2-1 ... 5.2-5).

Representation [E23, P11, D28]:

- The text and figures of section 5.1 are an informal representation of some conception of the test case domain.
- The figures and part of the text of this section 5.2 are a more formal representation of

some conception of that domain.

Representing action [E23, D29]:

- The action of the author of this section 5.2 is an example of a representing action.

Representer [E23, D29]:

- The author of this section 5.2 is an example of a representer.

Representing context [E23, D29]:

- The context of that representing action was to illustrate our conceptual framework.

Label [E24, D29]:

- The string of characters 'F-92-304' is an entity label of type 'Item number'.
- 'Fourtaney-1992' is an entity label of type 'Item name'.
- 'D-96080016' is an entity label of type 'Order number'.
- The strings of characters 'is on stock in', 'asks for' are predicator labels.
(See R1 and R4 in figure 5.2-4.)

Reference [E24, P12, D29]:

- The relationship between a specific product, and the item number 'F-92-304', is an example of a simple reference. Another example is the relationship between the same product, and the item name 'Fourtaney-1992'. Thus, there is a synonym situation. It is assumed that the item number identifies a product uniquely, while the identification via item name may be doubtful. Therefore, the item number is chosen as the preferred synonym in this case. (See R1 in figure 5.2-4.)

Semiotic level [E25]:

- The written delivery instruction ticket form is at the physical level, the statements provided by the form are syntactic matters, stipulations (what is requested to the driver) are at the semantic level, the on-schedule guarantee of delivering wines (the intention of JWI) is at the pragmatic level and has been negotiated between JWI and its clients in contractual terms belonging to the social level.
- The contract for developing a computerised information sub-system for JWI by a software house has a physical form (the written document), elaborated according to some textual patterns (empirical level) in a given language (syntactics), its thrust is at the semantic level, it results of negotiations (pragmatics) and reflects accepted commitments (social level).
- The software issued due to the contract has a physical appearance (code plus documentation), it has been syntactically checked (correctness and completeness proofs), its functionalities are at the semantic level, its expected usefulness is at the pragmatic level and its expected usability and use are at the social level.
- Assuming that the software house and JWI reached an agreement by exchanging e-mail messages, message bits would be the physical aspect of the agreement, exchange protocols between parties are at the syntactic level, the signification of the message reflects its semantics, whereas the intention behind the message is at the pragmatic level, and finally the legal validity of what the exchanging parties agree on is at the social level.

Model [E26, P13, D30]:

- Our interpretation of the test case in terms of the concepts from [E1, P1] to [E17, D23], derived from the statements [S1] to [S28], is a semi-formal excerpt of a model whose purpose is to reach a better understanding of the business of JWI.

Model denotation [E26, D31]:

- The tables and diagram in figures 5.2-4 and 5.2-5, depicting entities and relationships, are an example of a denotation of a partial model concerning the data aspects of the business of JWI.
- The diagrams in figures 5.2-1, 5.2-2 and 5.2-3, depicting actors, actions and actands of JWI and its environment, are a denotation of a partial model for understanding "who is doing what with what".

Modelling action [E27, D32]:

- See perceiving, conceiving and representing action.

Modeller [E27, D32]:

- See perceiver, conceiver and representer.

Intensional model [E28]:

- The model denoted in figures 5.2-1, 5.2-2, 5.2-3 and 5.2-5, including the rules mentioned, is an intensional model.

Extensional model [E28]:

- The model denoted in figure 5.2-4 is (a very small portion of) an extensional model.

Meta-model [E29]:

- The first part of our conceptual framework, from [E1, P1] to [E17, D23], is a meta-model, populated above with some aspects of the test case.

Coherence [D39]:

- Our (partial) model of the test case is a set of coherent relationships.

System [E30, D40]:

- A system analyst (system viewer) conceives the JWI distribution centre (the system domain) as a system composed of related and coherent elements, according to the description given in the statements [S1] to [S28]. The system has the capability of delivering bottles of wine to retail shops in less than 48 hours (one of its systemic properties).

System denotation [E30]:

- See model denotation.

System component [E30]:

- All the entities, relationships, actions, actors, actands etc. of JWI are system component.

System environment [E30]:

- The decision where to draw the boundary of the system depends on the system viewer. Usually, one would conceive the actions of the retail shops or of the wineries as not belonging to the JWI system, but to its environment.

System viewer [E30]:

- See perceiver and conceiver, or interpreter.

System representer [E30]:

- See representer.

Dynamic system [E31, D42]:

- JWI is a dynamic system.

Static system [E31, D42]:

- JWI is not static.

Active system [E31, D43]:

- JWI is an active system.

Passive system [E31, D43]:

- JWI is not passive.

Open system [E31, D44]:

- JWI is an open system.

Closed system [E31, D44]:

- JWI is not closed.

Sub-system [E32, D41]:

- The stock-handling system (with the capability of providing up-to-date data values on products), and the delivery-system (with the capability of scheduling truck journeys), are both sub-systems (with their systemic properties) of JWI.

Knowledge [E33, D33]:

- The collection of conceptions invoked by the description of the business of Japan Wines Inc., as, for example, stated in [S1 ... S18].
- Other examples of knowledge are: knowledge about the wine market, wine transportation regulations, wine pricing and taxing regulations.

Data [E34, D34]:

- The representation of an order (e.g. items numbers and ordered quantities); the representation of the quantities of products on stock in the JWI inventory; the phone number of a retail shop, are examples of data.

Message transfer [E35, P14, D35]:

- The phone call (comprising data, e.g. an order) made by a retail shop (a sender) to JWI (a receiver); the stock report (data) sent by a computer of JWI (a sender of the type device) to the JWI stock manager (receiver, a human actor), are examples of message transfers.

Message [E35, D35]:

- See message transfer.

Sending action [E35, D35]:

- See message transfer.

Receiving action [E35, D35]:

- See message transfer.

Sender [E35, D35]:

- See message transfer.

Receiver [E35, D35]:

- See message transfer.

Information [E36, D36]:

- The knowledge obtained from a signal indicating that quantity on stock of a given product is smaller than the minimum stock quantity is necessary to perform the

appropriate action, i.e. place an order to a winery.

- Knowledge about wine pricing is obtained from a wine dealer journal.
- Knowledge about the wine market may be obtained through a consumer study reported in the same journal.
- A call emitted by the mobile telephone of a delivery-truck driver (for example, on truck failure or accident) conveys the knowledge necessary to re-schedule delivery tours.
- All these are examples of knowledge represented and communicated in some way. Provided that knowledge is new to the staff members of JWI, these are examples of information, helping to decide on actions as appropriate for managing the wine-distribution business.

Communication [E37, D37]:

- JWI communicates with retail shops and wineries about orders and deliveries.
- JWI may negotiate about a contract for developing a computerised information sub-system with a software house.
- Within JWI, there is communication going on about handling orders, stock of products and deliveries (see above examples).

Shared knowledge [E38, D38]:

- The staff members of JWI communicate with each other to share their personal knowledge about orders, deliveries, stock of products, contracts, the wine market, wine transportation regulations, wine pricing, taxing regulations, etc., for managing the wine-distribution business properly and effectively.

Organisational system [E39, D45]:

- The enterprise JWI, including its personnel and computers etc. (actors), its technical resources (e.g. barrels, bottles, packages, storage rooms) (actands), its goals (to distribute wine to retail shops and earn money) and business rules (as expressed in statements [S1] to [S28]), together with its ways and rules for internal and external communication, forms an organisational system. It is structured in departments, such as accountancy department, personnel department, stock administration department etc.
- In a wider sense, the full JWI organisational system might include:
 - other actors: business partners, trade unions, etc.
 - other resources: garages for truck repair, offices for employees, finances, etc.
 - other operating rules: promotions of employees, selling and buying trucks, hiring and firing employees, billing retail shops, etc.
 - other interactions and communications with retail shops (invoicing), with wineries (paying bills), with employees (waging), etc.

Norm [E39]:

- The society in which JWI is embedded in (e.g. Japan, the wine market world-wide), has certain norms which have to be obeyed by JWI in order to be successful.

Knowledge/data-processing action [D46]:

- See information and communication.

Information system [E40, D47]:

- That sub-system of the organisational system just described which encompasses all information-oriented aspects of the JWI business forms an information system. It contains, for example:
 - Actors dealing with business information such as managers, organisers, data processing personnel, people responsible for sales, purchase, and accounting of wines and for other resources (such as warehouses, wineries and delivery trucks);

- Messages such as forms and delivery instruction tickets, transitions and operations such as scheduling, delivering, re-delivering, work regulations, methods, tools and other resources used for managing the flow of information and communication within the enterprise and with its business partners, etc.

Information system denotation [E40]:

- Statements [S1] to [S28] are representations of conceptions dealing with, among other things, information-oriented aspects of the JWI organisation. They might be considered as an information system denotation expressed in a very informal way. A collection of entity-relationship diagrams, and a collection of actor-action-actand diagrams (e.g. figures 5.2-5 and 5.2-3) expressed in some graphical language, together with a natural-language description of the various rules and constraints, would an example of a semi-formal information system denotation. A set of abstract data types with formally defined signatures and algebraic specifications of operations would be an example of a formal information system denotation.

Computerised actor [P15, D48]:

- Any computer used for stock and delivery administration etc., is a computerised actor.

Computerised information sub-system (CISS) [E41, D49]:

- The computerised part of the information system including all hardware and software used for stock and delivery administration etc., forms a CISS.

What I tell you three times is true.

Lewis Carroll, *The Hunting of the Snark*

I hate to hunt down a tired metaphor.

Lord Byron, *Don Juan*

6 ELABORATION ON SELECTED TOPICS

During our work, we found that various disciplines have to offer contributions for a better understanding of our own field, information systems. We identified in particular three disciplines which we have been considering highly relevant to understand the foundations of our field:

- *semiotics*, the discipline concerned with theories of signs,
- *system science*, the discipline concerned with theories of systems, and
- *ontology*, the discipline concerned with theories of how the "world" may be viewed, conceived or modelled.

This observation gave rise to the metaphor of the "temple" with the three "pillars of wisdom" of the information system field, the "semiotic pillar", the "system pillar", and the "modelling pillar". It should be clear from reading the preceding chapters, that the relevant contributions of these disciplines have been fully integrated in our conceptual framework, and that the "temple-pillar" metaphor is thus out of date.

This chapter 6 may be seen as a tribute to the old metaphor. It is meant to provide supplementary material on these issues. The sections on semiotic concepts (6.1), system concepts (6.2), and modelling concepts (6.3) can be read independently of each other, and possibly even independently of the previous chapters.

6.1 Semiotic Concepts

A BRIEF HISTORICAL SURVEY OF SEMIOTICS

The discipline concerned with theories of signs, *semiotics* or *semiology*¹ has a long history dating from the ancient Greek philosophers. John Locke in his *Essay Concerning Human Understanding*, treats semiotics together with physics and ethics as one of the three main branches of human knowledge. Semiotics and semiology overlap, but they have different flavours.

Semiotics and semiology are intertwined with linguistics - the scientific study of languages and their use. Semiology, was in fact born in a purely linguistic context by Ferdinand de Saussure, whom most people regard as the father of modern linguistics. Semiology is also associated with writers such as Roland Barthes, or Umberto Eco. Semiotics, on the other hand, has grown out of the work of the American logician Charles Sanders Peirce [HW31, Moo+82, Mor46] who created by the way also the relational calculus, the foundation of relational database technology. Semiotics looks at signs in a broader perspective, but one that also incorporates a linguistic perspective. One of the most prominent relationships between semiotics / semiology

¹ Greek: 'semeion', diminutive of 'sema' = mark or sign.

and linguistics is that most linguists regard languages as being sign systems or as being based on such systems.

Studying the linguistic literature - at least as an outsider - leads one to the conclusion that there is no consensus with respect to the more detailed relationships between semiotics / semiology on the one side, and syntactics, semantics, pragmatics and other disciplines of linguistics on the other. There are many interpretations, and unfortunately there is much terminological confusion, possibly also regarding the general relevance of every one of the vast amounts of specific concepts recognised in the various branches of linguistics. There are many, seemingly rather different linguistic congregations concerning perspective as well as emphasis, and each one contributes to the overall confusing picture. Fractions like "the Prague School", "the Paris School", "the Copenhagen School", and also the fact that there are European versus American traditions in linguistics clearly indicate that there is a long way to consensus and proper insight in the underlying more general principles.

Here, we shall adopt the term 'semiotics' for the study of basic principles of representation. 'Semiotics' is chosen, because our work on this issue has been primarily inspired by Peirce's notion of sign. Semiotics provides the sign concept as a primitive notion, one which can be understood on the basis of demonstration, or ostensive definition, and upon which we can build sound technical definitions of "information", "meaning", "communication", and so on, terms which are used very vaguely in the information systems literature. Another important contribution to the information system area comes from the established division of semiotics into sub-disciplines which can be studied relatively independently, and used to focus attention on relatively self-contained problem areas.

The key principle behind the variants of the sign concepts considered in semiotics is that one thing stands for - represents - another one. The principle of representation has always been closely connected with the general notion of sign. But in the course of time, it has been manifested in a number of specific, more or less different sign concepts. However, common for the explanations of all these sign concepts is the assumption and incorporation of some sign components. All known sign concepts are defined with reference to at least two such components.

The classical sign concept dates back to the time of Hippocrates in Greek antiquity. In the classical interpretation 'semeion' was used for the symptoms available for a doctor trying to determine a hidden illness or when observing a possible cure of it. Expressed in Latin the classical sign was defined as a thing related with another thing in a certain way: "aliquid stat pro aliquo" (something that stands for something else). In the Hippocratic interpretation the symptom "is a sign" of an illness. A change in the symptom "is a sign" of the illness becoming worse or, alternatively, "is a sign" of the effect of some cure or of natural counter actions by the body.

If we take a closer look, it appears reasonable to regard this "medical" sign interpretation just as a special case of what commonly is called "natural signs". This category comprises natural causal phenomena as exemplified by phrases like 'The smoke from the chimney is a sign of fire in the stove' or 'This fresh track in the snow is a sign of a fox coming along here recently'. More generally a natural sign is a thing that can be observed or otherwise sensed which is caused by "a thing in nature".

- **Classical/natural sign** = < Symptom, Cause >

In this sign interpretation, there seems to be a considerable emphasis on the first component.

The symptom is the sign (of the cause). This is clearly reflected in common language use of the word 'sign', as illustrated by the phrases above, and this aspect does in fact contribute much to the general problems with insufficient basic understanding in semiotics.

The classical sign concept was employed by the Stoic philosophers to include both logic and epistemology. Apparently, already at that time it was common to consider the close association between signs and (at least logical) statements in some language. This connection became stressed at the beginning of our century when de Saussure introduced his sign concept as a binary relationship between two linguistic concepts: For him "le signe" (the sign) was the junction or combination of what he called "le signifiant" - a signifying or denoting component - with "le signifié" - the other component that on its side is the signification of the first.

- **de Saussure's sign** = < signifiant, signifié >

In his original approach "le signifiant" was restricted to be phonetic (a so-called "acoustic image" resulting from speech) that represented "le signifié", while that component, on the other hand, was what is meant by the orally expressed sound. However, Saussure's concept is equally well applicable in a non-oral linguistic perspective. This in a way is confirmed by one of Saussure's own fundamental linguistic principles, that of seeing a language as "constructed" from a system of linguistic signs, where there is freedom to chose the first component independently of the second one.

Already before 1870, but independently of Saussure, Peirce began to modify and expand the classical sign concept and defined it eventually as: "*A sign is something which stands to somebody for something in some respect or capacity*".

This definition of sign can be interpreted as a ternary relation:

- **Peirce's sign** = < representamen, sign object, interpretant >

where the three sign components according to Peirce are explained in this way:

- The *representamen* (by Peirce also called 'the primary sign') is a thing serving as the "carrier" of the sign independent of its meaning;
- The *sign object* is a thing which the sign alludes or refers to;
- The *interpretant* is some intermediary body able to interpret this relationship.

The role of semiotics in the development of the information system discipline is growing rapidly now [e.g. And90, ANN96, HAPK96, Sta96a].

SEMIOTIC LEVELS

The divisions of semiotics introduced by Charles Morris [Mor46] are *syntactics*, *semantics* and *pragmatics* which deal, respectively with the structures, meanings and usage of signs, reflecting the philosophical roots of the subject. In recent years and in business contexts the factors governing the economics of signs have become important. Hence we must add the *physics* of signs, concerning the media in which signs are embodied and processed, and the *empirics* of signs which concerns the statistical properties of sets of signs, treating such problems as the design of codes for channel with given noise characteristics (the problem that inspired the 'Mathematical theory of communication', founded by Shannon [SW49] and further formalised by Khinchin [Khi57]). Finally, the most recent advances in semiotics make it clear that we cannot fully account for the properties of signs without more explicit recognition of the *social* dimension in which they find their purposes. The full semiotic framework is given in

figure 3.5-2.

Physical Signs and Empirics

In the information systems field we need to model the physical means of representing signs. The commonest terms used for a phenomenon with this function are "signal", if it is dynamic, or "mark" if it is static or "sign token", in general. We use the methods of physics and engineering to study physical-tokens or signals and marks, looking for cheap, compact and easily manipulatable forms. We also need to take account of the capacity of human senses to discriminate between physical phenomena whilst, socially, there must be some consensus to treat certain signals or marks, qua physical phenomena, explicitly as tokens within a semiological system.

In this physical area of semiotics we model signals and marks, their sources and destinations and the routes over which they are transmitted. All the usual physical properties of objects (marks) and events (signals) are relevant - such as the numbers of distinct tokens per unit of time or space, their material and energy content, their rates of deterioration, speeds of transmission and so on. These are of particular importance in finding the costs of information systems, and information technology equipment manufacturers compete mostly in these terms.

The word "information" has many different common-sense usages. In the physical area it is often used in the sense of a collection of tokens as when we talk of the information in a database amounting to x megabytes, for example. "Communication", another of the ambiguous words in our discipline can be interpreted as an effective chain of cause-and-effect relationships linking different places and times. If communication in an information system fails at a physical level we mean that such a chain has been broken.

Models of information systems at the physical level describe the available ranges of *physical tokens* (signals or marks) available as input to or output from various physical components and they model the cause-and-effect relationships between them, their energy and material requirements. For example, when developing information systems we model the locations and telecom links between sites, the volumes of information (sic) in characters per hour which they generate and the volumes they store, in order to discover the capacity of channels and storage devices needed.

Irrespective of the physical representation, empirically, the number of symbol values available for encoding a message and the length of that message allow the extent of mapping a condition of a "source" such that transmission over a channel the recipient of the message may interpret it as a representation of that condition. Thus it provides "information" in the sense of answering the potential question as to what the source's state is. The aforementioned theory of communication connects the logarithmic measure of entropy with the probabilities of the various possible source states.

At the empirics level, "information" has at least two distinct meanings, viz. that of the available and of the usable variety [Sca93]. "Communication" is included in "mutual information", the entropy of one source conditional upon the signals at another to which it is correlated. Empirics is concerned with coding so that the statistical behaviour of the messages can be matched most efficiently to the statistical characteristics of the media (signal to error rates). Thus "meaning" in this context means the equivalence of codes. The concept of "redundancy" relates to the superfluous component of a message which in a common sense way, adds no meaning (but may serve for detection and correcting random transmission errors).

Syntactics

If we can find a satisfactory range of physical tokens that provide a reliable way of encoding a diversity of elements, we can ignore the problems of physical representation and their statistical properties in order to concentrate on the problems of syntactics. Here we are concerned with complex *structures* of signals/marks regardless of their actual expression in specific terminal tokens. We can define a syntax using names of syntactic categories and production rules, but we are indifferent about the rules specifying the terminal characters. In this syntactic domain, we focus on form regardless of how it is physically represented and regardless of any empirical or statistical properties of the language expressions in use.

Complexity and structural richness are concepts that belong in the syntactic domain. Syntaxes may be classified and ranked according to their power to generate structures. The well known Chomsky order [Cho68]:

- finite state languages
- context free
- context dependent

forms a part of this structure. The rules for generating and parsing formal expressions allow us to measure the complexity of a sign system, the complexity of a single formula can be measured also. When a formula can be produced or parsed in two or more ways we have an instance of *syntactic ambiguity*, another characteristic semiological property in the syntactic domain.

Symbolic forms can be transformed (or mapped into one another) according to rules. These may add to the richness of a syntactic system especially if we introduce a concept of *validity* which allows us to create a logic. Formulae which can be obtained from one or more other formulae (the premises) using the transformation rules are called "theorems". Logical validity is determined by a set of axioms which belong to the set of valid formulae from which all the other logically valid formulae can then be derived using transformations. The closure of the set of axioms under the permitted transformations gives us a set of logically valid formulae and another set of formulae that are always false, as a property of the language. This leaves all the other formulae which may be valid (true) contingently upon our choice whether or not to treat them as valid for some purpose, for example because they represent some facts.

In such a logical system we can give "information" a precise range of meanings. First we construct the set of "state descriptions" each of which is a maximally informative statement, in the sense that it says as much about one state of affairs as the language can express. Any two different state descriptions would be contradictions. By defining a measure function (logical probability) over this set we can measure the information in any other formulae, such as x and y, in the diagram below. What we do is to select all the state descriptions which imply the formula and take the sum of their probabilities.

Measures of information may be derived from these logical probabilities. Many different precise definitions can be obtained in this way depending upon the logic employed, the definition of the basic measure function and the derivation formula [CB53, Bar64]. These are totally different from the familiar entropy measure introduced by Shannon [SW49].

Consider the other difficult words:

- "Meaning" has a precise definition in the syntactic domain using the transformation rules. Two sets of formulae have the same meaning (paraphrase each other) if each can be deduced from the other. The meaning of one includes the meaning of the other if it allows the other to be deduced from it.

- "Communication" is a syntactic concept if we consider the ability of one device to communicate with another by virtue of their having compatible structures. By coupling such devices we can study the syntactic properties of complex machines. Communication between devices is possible syntactically if the well-formed-formulae output from one may be treated as an input of well-formed formulae to the other. This is the sense in which we talk commonly talk of communicating computing devices, those which are "plug-compatible".

The Technical Platform

Notice that the work we do on hardware (physics), telecommunications (empirics) and software (syntactics) constitute the technical platform for an information system. Strictly speaking all the problems at these levels may be formulated and solved without any regard for the relationships between the tokens and what they are presumed to represent in the "real" world nor any regard for the people who use them nor their intentions in doing so. The natural sciences and mathematics provide us with the necessary concepts and methods of enquiry.

These technical dimensions enable us to form *utterances* or physical tokens within some empirical domain and possessing a more or less rich structure. Working on problems of the technical platform we can use the word "actor" in a metaphorical sense when we talk about a physical device. We are, in the domains "physical-empirical-syntactical" (or the hardware/software domain) indifferent to any human values or intentions which we associate with responsible agents. These aspects are added in the next three domains of semiotic framework which are concerned with the human information functions that are essential to the conduct of business. The information systems professional who confines his attention to computers embedded within other machines can probably limit his concerns to the problems in the areas of the physics (including empirics) and syntactics of information, but the design of business and administrative information systems will fail unless the next three levels are given due attention.

Semantics

Semantics is concerned with meaning but, as we have seen "meaning" can be given a variety of meanings in the physical, empirical and syntactic domains. Many writers prefer to reduce semantics to a problem within the technical platform, especially if they value logical and mathematical rigour above all else. The most thorough treatments of semantics in this technical spirit are achieved by defining a "real" world comprising a set of discrete individuals and then defining mappings from formulae in the system of utterances into this real world system. This kind of "meaning function" provides a neat mathematical treatment (see the elegant semantics of English by Richard Montague [Tho74]) provided that one can accept the assumptions behind this ready-made reality of discrete individuals postulated in the theory.

Those assumptions are not unreasonable ones to make if we are dealing with simple problems where the people involved share a well established consensus about the boundaries separating the discrete individuals in the world. But they fail when that consensus breaks down. Some aspects of business, especially those where computers have a long history of application (for example, accounting and production scheduling) do allow us to make the assumption of an objective reality but when opinions differ and negotiations are needed to draw boundaries, then we have to treat the information system as much a means of creating the reality as a means of describing it. As yet, there is no widely accepted formal method of treating semantics on the basis of a subjectively created reality.

These positions illustrate the use of two quite different *semantic principles*. One, an objectivistic principle assumes that meanings are mappings from syntactic structures onto objective features of a real world which is the same for everyone and which everyone knows independently of language. The second, which we might call constructivistic, assumes that meanings are constructed and continuously tested and repaired, through people using syntactic structures to organise their co-ordinated actions, the repair taking place when they judge that the language-action relationships have failed.

Each of these semantic principles has a range of problems for which it is appropriate. Principle (1) suits simple routine administration, (2) suits problems where conflict and negotiation are endemic and (3) can be used for one-way communication of reports where the content and validity are not open to challenge. In future business systems, principle (2) will have to be used more often because the negotiation of meanings is unavoidable in systems that have a wide scope whether in subject matter or people involved in using it.

How should we interpret our tricky words in the semantic domain?

- "Meaning" has different meanings according to the semantic principle adopted. We have sketched a few such principles without giving them precise operational definitions, each of them will generate its own exact meaning of "meaning".
- "Communication" can be understood in a semantic sense by linking two representational triads.

Semantically we are indifferent what medium (speech, writing) is used, what language (Dutch, English), how the message is encoded or by what medium it is transmitted. Communication is successful if the semiotic tetrahedrons are the same or at least similar, judged by some operational procedure.

- "Information" can be analysed from a semantic view in many ways. Carnap and Bar-Hillel [CB53, Bar64] define semantic information measures using the method introduced above for "syntactic information". Another approach uses two triads in which a single interpreter's view of a second signification is informed (is shaped) by an utterance. The first significatum mediates between the utterance and the second significatum.

For example, when a proposal is presented at a staff meeting, the applause uttered by an audience will signify a certain attitude which, in turn, may signify a likelihood to act positively on the proposal in future. This kind of connection may be modelled formally in some cases but generally the information will also be *relative to the interpreter*. This idea can be operationalised in many different precise ways, each giving rise to a precise meaning of semantic information.

Pragmatics

It may be argued that a sign must always have an intention imputed to it either by its creator or its interpreter. Pragmatics is the branch of semiotics concerned with the relationships between signs (as meaningful utterances) and the behaviour of responsible agents, in context. Many computer-based systems are designed without regard for the pragmatic level.

The *context* is essential if we want to understand signs in the pragmatic domain. At the other levels we can more or less forget the context, but signs used for action often have little meaning when taken out of context, and the interpreter, at that concrete time and place, will make pragmatic sense of the utterance [And90]. For example, consider two workmen installing a clumsy machine into a small space - their interactions may seem like inarticulate grunts but they may be sufficient to accomplish a complex and difficult task. Functional grammars are beginning to give us a formal approach to these problems. Software which cannot take account of context may be tedious to use and misleading to interpret.

It is also possible to model some aspects of intentional sign use in terms of speech act theory by studying so called illocutionary verbs (e.g., state, request, suggest, judge ...) which reveal the intention of the speaker/writer and require some intentional response from the listener/reader (e.g., assent, refuse, object, concur ...). Some pragmatic analyses can be handled in formal ways, for example illocutionary logic. However much pragmatic understanding of signs can only be acquired and used in the situation, for example during negotiations.

Now let us consider the difficult terms:

- "Communication" is perhaps a key word at the pragmatic level where it is most fully explained. Communication takes place successfully when a meaningful utterance is used with a certain intention which is interpreted with the same intention by the listener. There must always be someone giving the sign its intention (e.g., the person who signs the cheque) and that intention must also be signified. It may be technically difficult to add to the utterances generated by computer (e.g. demands for money) a sufficiently trust-worthy indication that they bear the intention of a suitably authorised person.
- "Information": The amount of pragmatic "information" in a communication depends upon the richness of the consequences which stem from the intentional message. To give these consequences an operational form we need to take account of the social-level which is the context in which signs have any effect.
- The "pragmatic meaning" of a communication can be defined in terms of the actual social consequences which stem from it. This will include any changes in shared beliefs, commitments to action and changes of attitude. Precise definitions at this level will depend upon our having precise models of the social structure.

Social Level

No sign can be fully understood without regard for its potential or actual social consequences. Signs are the means by which physical artefacts (tokens) are used to create, sustain, alter and use the social world. A mere signal or token is not a fully developed sign unless it functions at this social level.

When, with one or more partners, we conduct a conversation (that is, a well-formed chain of speech acts) the result will be a change at the social level. Each *illocutionary* act will have a social consequence achieved by one or more listeners performing *perlocutionary acts*, which change the social world.

The social world consist of norms of many kinds - ways of behaving, sets of values, shared models of reality, common attitudes and so on. These define the shape or form of society and *pragmatic information* may be operationalised as the extent to which the message can change the social form.

- "Information" at the social level is perhaps best understood in a totally different way as a *process* of imparting form. A sign is very informative, in the pragmatic sense, if it produces far reaching social changes (for example by totally undermining some accepted scientific theory as did the results of Michelson & Morley's experiments, which ultimately led to relativity theory). The amount of social information (as opposed to the change induced) might be measured in terms of the richness of the structure in a social system.
- "Communication" (OED7: common participation) may be best used synonymously with "communion" (OED1: sharing or holding in common) in the sense of possessing a framework of common understanding necessary for signals or tokens to be used to inform.

We might measure the extent to which norms are shared between two groups in order to measure their level of social communication.

- "Meaning" seems to be used in a social sense when people talk of meaningful situations in life. Intuitively this concerns the coherence of what is perceived and understood, suggesting that social meaning could be defined in terms of the social norms involved. Thus the meaning of a sign token could be defined in terms of the norm structures which governed its use and without which it could not function fully as a sign.

To understand the social level, the concept of a norm is essential. "Knowledge" may be defined in terms of norms. The concept of a norm is not a vague one but one that can be operationalised either by experiment, to ascertain the existence of regularity of behaviour, perception, judgement etc. or by expressing the norm in a written form and asking for judgements about the validity of the formulation. An observer can also grasp a norm by using it and by trying to depart from it whilst acting as a member of the group to which the norm belongs. Norms can be modelled and, therefore, handled by our syntactic devices.

Although the distinction of the pragmatic and social levels must be clearly understood, in the context of the FRISCO framework they may be viewed as one, viz. that level where utterances may result in action (or reaction) based upon agreed or negotiated understanding of what is meant or intended.

Modelling and Models

Model denotations are signs. Hence we should be able to consider the semiotics of models. They must be well-constructed and function correctly at all semiotic levels if they are going to be useful. For computer work, we are mainly concerned with symbolic models. Such models may also be constructed to analyse aspects of signs at each of the semiotic levels. Let us look at the key concepts to be included in this range of information system models:

- Physical: use of various media for modelling - documents, wall charts, computer-based CASE tools etc.; physical size and amount of effort to manipulate them; human resources needed; economics;
- Empirical: variety of elements distinguished; error frequencies when being written and read by different users; coding (shapes of boxes); ergonomics of CHI for documentation and CASE tools;
- Syntactical: languages, natural, constrained or formal, logical and mathematical methods for modelling;
- Semantical: interpretation of the elements of the model in terms of the real world; ontological assumptions; operations for arriving at values of elements; justification of external validity;
- Pragmatic: roles played by models - hypothesis, directive, description, expectation; responsibility for making and using the model; conversations needed to develop and use the model;
- Social: communities of users; the norms governing use for different purposes; organisational framework for using the model;

These lists are indicative rather than exhaustive. Clearly, unless we have a purely mathematical interest in them, our understanding of models in the information system field should not be confined to their syntactic aspects. A person in a company who wants to install an information system, thereby using a range of models, should consider all the above aspects.

6.2 System Concepts

The notion of system¹ is not uniquely defined in the literature, but typically, it can be found explained as: "A collection of interrelated parts characterised by a boundary with respect to its environment" [Liv83] or just as: "A set of objects with a set of relations" [Lan71].

Most people intuitively agree on such simple definitions. Apparently they are broad enough to cover the meaning of usual linguistic constructs where 'system' is used.

But system is a much more difficult concept. If we look at what in practice are considered systems, and if we really think about it, it becomes obvious that some very important aspects of the system concepts are missing in the traditional definitions.

One can regard an organisation or a bicycle as systems. Also a Hitchcock film recorded on a video cassette, which is inserted in a video cassette player, which again is connected to a TV-set, could easily be interpreted as a system. Nothing is unusual with such system views, and they are well covered by the definitions. But if you buy some eggs from a farmer and use two of them for breakfast, then the domain of obviously interrelated phenomena: You, the farmer, the farmer's hen that laid the eggs, the frying pan you used to prepare the eggs, and the two eggs now in your stomach (and thereby in some transformed form a part of yourself) - this domain might probably not be regarded as a system, because it might be difficult to see a purpose for that. But it fits the definitions.

Or consider a single raindrop in an April shower: It consists of a vast number of water molecules, kept together by surface tension and constantly moving around among each other in a complicated manner controlled by a set of (thermo-) dynamic forces. Again according to the simple definitions above, the drop qualifies as a system. But that is strange, because when you on your way back from the farmer, happen to get soaked in the shower, you might feel it is caused by raindrops - not by systems.

On the other hand, a meteorologist studying possible weather situations that could cause rain, may see a purpose in regarding a raindrop as a system in interaction with the surrounding atmosphere, but in most other situations a raindrop is just a raindrop.

The key to understanding the system concept is to realise that a system is *not* an absolute or objective phenomenon. Systems are not *a priori* given. As Checkland expresses it, there must be a describer/observer who conceives or thinks about a part of the world as a system [Che81]. This subjective view of the system concept should be stressed: It is important, that there is a system viewer who can see a purpose in regarding something as a system.

The purpose can be expressed as at least one meaningful relationship between the domain of elements considered as a whole and the environment. Such a relationship is called a "systemic property". It is a property the system viewer associates with the elements conceived as a whole. One system viewer sees the domain as a system having one set of systemic properties, while another viewer may see other systemic properties concerning the same domain.

Therefore, it is necessary to consider the following concepts and distinctions:

System, System Domain and System Viewer

¹The term 'System' is derived from the Greek phrase 'Syn histanai' ~ to put together.

A system is a special model of a domain, the system domain, conceived by a system viewer, whereby all the things contained in that model (all the system components) are seen as being coherent and related to form a whole. A system is conceived as having assigned to it, as a whole, a characterising predicate, the systemic properties.

Most often, the systemic properties of a system cannot be attributed to any of its components. For instance, none of the bicycle parts has the "bicycle property" which they have together when they all are assembled properly, and neither the video cassette, the cassette player nor the TV set themselves can do, what they can together as a "video system". The farmer-you-frying-pan-eggs-hen set on the other hand may not constitute a whole with any sensible systemic property.

A system viewer is a person who conceives the system domain as a system.

The environment of a system are such conceived aspects of the world which are not considered as part of the system.

One objection against this "subjective" system concept is that it is sensible to talk about "designing and implementing a system" or "to interact with a system". This gives associations to the word 'system' as denoting something that can be interacted with in a rather concrete way and not just as a conception. However, there is no inconsistency, because these and similar phrases are just (conveniently) short terms for more elaborate expressions. For example, to interact with a system means to interact with phenomena in the system domain that is conceived as a system because of its systemic properties. To design and implement a system means to bring together ("Syn-histanai") and structure phenomena in a particular part of the world (which then becomes the system domain) with the purpose of constructing them such they together have certain systemic properties.

Do we really mean the same system? One serious cause for the current confusion in our professional domain is, that people, usually, think about a system as something that can be objectively determined, for example by a specification of its parts and their relationships, as the above quoted definitions may indicate. People overlook that a systemic property must also be described. Otherwise it has no sense to talk about the domain in question as a system. Without the whole conceived to have at least one systemic property, it is just a model. But even if we focus on systemic properties, there is still the problem, that they may depend on the system viewer: Which systemic properties should be considered in a given situation?

As an example take the simple domain of a car and its driver in the traffic of a city. One person may see it as a useful transport system in action, which is able to move large objects from one location to another in a convenient way. The driver alone cannot, nor can the car, but in combination they can. However, a policeman on his job will regard the same domain differently - as a controllable system which behaviour can be directed by road regulations, traffic lights, arm signals and by certain traffic rules. Again, an environmental activist would probably regard the car as a dangerous polluting system, which is a potential cause of injury or death to persons in the traffic.

Here we have three views of the same domain, but with quite different sets of systemic properties. All three persons could in fact be the same system viewer - a transport conscious public servant caring about the conditions for people in the city, who just conceives different systemic properties by regarding the same domain from different points of view.

Let us elaborate this car example a little further in order to illustrate the difficulties we face when we regard something as a system:

Consider for example the question about which parts and which activities are involved in the possible system views: Are the driver and the car two interacting sub-systems - one with the systemic property of being able to observe the traffic and to control the car, and the other with the property of being able to transform chemical energy into movement in a controlled manner. Or is the car to be regarded as a single system with the driver, motor, gear, and steering devices as sub-systems each with their own systemic properties? Is the motor the active part and the chassis a passive component, or is it the other way around - the car as a device transporting among other things the motor. A quite another system view - but still one from the same domain - could be to regard the car as a moving Faraday cage protecting the driver from certain kinds of dangerous electrical fields. There are many possible system views, and still the domain is extremely simple compared with the organisational domains usually considered as system domains.

If we regard a business enterprise, an institution or any other kind of organisation as a system - an organisational system - we have a domain which is much more complicated than a car and driver. Furthermore, the number of possible views of an organisation is most often enormous.

In our area parts of these organisational domains are often conceived as information systems. But what is, in fact, the domain of an information system in any of the senses of the term, and what are the systemic properties of such information systems? It is a claim of this report that if this is not specified and explained properly, it has no sense to talk about information systems. Furthermore it is claimed that if one tries to explain information system without observing in general the distinction between system domain and system, the attempt will fail. People simply will not be able to understand what kind of system it is.

System Type and System Exposition

In order really to understand the concept system it is necessary to be aware of a number of important aspects:

- that the system domain always comprises several elements
- that all elements are related to each other such that it constitutes a transitively coherent whole,
- that the whole is conceived to have at least one systemic property,
- that it is only relevant to incorporate a thing as an element of the particular system domain if in the system view it somehow contributes to the systemic property,
- that when viewing a thing as an element of a system domain then only those aspects of the thing that directly or indirectly contributes to the systemic properties are relevant for the system view.

When a systemeer¹ in a system viewing process gradually realises what (currently) "is the system", i.e. becomes conscious of all relevant aspects of the involved elements and of each of the systemic properties, it is very useful to be aware of the type of system in question and to produce a system exposition in accordance with the system type.

- A system type is a type that determines the potential kinds of systemic properties, elements of the system domain and roles of the elements in achieving the systemic properties.
- A system exposition is a representation of all the elements of the system domain where each element is specified by all its relevant aspects and all the roles it plays, being of importance

¹A system developer

for the system view. (The system viewer may conceive one and the same thing in the system domain to play more than one role in the system.)

A system type can be regarded as a system viewing template to be used by a systemeer in order to decide which kinds of things (and thereby which aspects of the things) to consider relevant as "system viewing actands" in realising what actually "is the system". A system type comprises:

- Properties determining "the nature" of the systemic properties, for example for open active systems that the system is seen as something that changes things in the domain of the environment and that the environment is seen as changing things in the system domain. This set of properties may be called the *system characteristic*.
- Properties determining the kinds of things which it is relevant to incorporate in the exposition of the system domain, and for each kind the kinds of roles they may play in respect to the potential kinds of systemic properties. Examples of such kinds of things are for dynamic systems: states, transitions and transition occurrences, and for open active systems (among other things): actions, actands, actors, transitions in the domain of the environment caused by actions in the system, etc. This set of properties may be called the *exposition characteristic*.

A more detailed elaboration of concepts related with the system viewing process can be found in [Lin92a]. A semi-formal description of it based on an example is presented in [Lin92b].

The Systemeering Process¹

The subjective nature of systems is probably the main cause of the serious problems recognised in practice in systems development (systemeering). Conceiving something as a system may appear as relatively simple for a single person as long as the system will be kept only in his mind. However, it is absolutely not a trivial task for the system viewer to be explicit about it, such that other persons can understand it. And exactly that is the core of the problem, because the process of "developing a system" nearly always involves the expertise of several - often many - persons who may have quite different organisational and professional background and who not necessarily will be fully aware of all the concepts relevant in systemeering. Thereby the practical problem of establishing inter-subjectivity among the involved persons about the system may easily become quite overwhelming.

The main reason is that for a very long period each of the involved persons will have their own personal view of what the system in question is, and these views may be different. But adding to the problem is that people are not always aware of the need of a long and intense communication period as a part of the project. Most often without reflecting about any uncertainty, each person will simply assume that the system they collectively aim at is exactly the one covered by the person's own view.

Even if everybody is conscious of the diversity of system views, in the systemeering process they must still collectively overcome three kinds of uncertainties:

- What are the elements of the system domain?
- What are the systemic properties?
- In which way do the elements contribute to the systemic properties?

The combined area of uncertainties about all these things from all the individual system views may be called the systemeering domain.

¹The process of developing a system

In order to co-ordinate their efforts such that everybody finally works in the same direction - on the same system - each of the system viewers must express themselves unambiguously about their own individual system, and each one must try to understand the other viewers' systems. Furthermore, as the mutual understanding of the different systems (hopefully) grows, they must aim at and finally agree on a synthesis - a single common view - the system.

In the long and enduring communication process that is necessary to achieve this goal, the only means of the involved partners is mutually to produce, distribute, read and understand (partial) system representations expressed in some commonly accepted and usually system-type-dependent system language. Regardless of whether the representations are expressed orally, on paper or on a screen by means of some CASE-tool, and independent of whether it is expressed as text or by graphical means - being explicit about the systems by means of system representations is the only way the involved system viewers can unite their views.

- A *system representation* is a representation produced by a system viewer about certain aspects of a system.
- A *system language* is a professional language designed to enable the representation of a certain type of system.

To succeed in the exchange of system views and thereby ending with a single inter-subjectively shared system requires knowledge on two abstraction levels and the corresponding linguistic means for expressing it:

- System-type-specific knowledge
- Systemeering-domain-specific knowledge

The knowledge specific for the system type comprises insight with all the kinds of things and corresponding concepts which are relevant to consider for all potential instances of that system type, and an according terminology necessary to express oneself properly about the assumed system. This kind of knowledge is a part of the qualifications required for professional systems. It serves as a kind of template for the systemeering work, and it constitutes the basis for the professional language that is used by systems.

The knowledge specific for the systemeering domain is knowledge of the concepts and kinds of things and of the (local) professional terminology that is relevant, respectively used in the particular systemeering domain (and more specifically in the system domains). This knowledge is obtained through the systemeering process, by observing the system domains and by communicating with people who are active there and/or supposed to know of them.

Organisational Systems

An organisational system belongs to a system type that primarily is characterised as being open and active (where the latter implies also that it is dynamic). For open active systems (OAS) - and thereby for organisational systems too - it is relevant to consider the following:

The behaviour of an OAS is generally reflected as:

- *Internal function:*
Conceptions of changes in the system domain caused by processes in the domain itself.
- *External function:* Here the following two kinds are distinguished:
 - *Impression:* Conceptions of changes in the system domain as caused by the environment.

- *Expression*: Conceptions of changes in the domain of the environment as caused by the system.

The very fact that something is regarded as a system often serve the purpose of hiding the internal function and focus on the external function. (Cf. the phrase "a black box system"; see the discussion on sub-systems below).

The following terminology is recommended: The internal function of an OAS is referred to as "the function *in* the system", while the external function is "the function *of* the system". The latter is equivalent with the systemic properties of an OAS, and may be, therefore, also called the *systemic function* of the system.

Open active systems can be classified in several ways according to their behaviour (for details see [Ack71]). Here we shall only distinguish between three kinds of OAS based on the following distinctions. A reaction of an OAS is an expression that is seen as unconditionally caused by an impression. An action of an OAS is an expression that is seen as being completely independent on any kind of impression.

Thereby we can define the three kinds of systems:

1. A reactive system is an OAS where each expression is a reaction, and where each impression immediately causes a reaction.
2. A responsive system is an OAS (possibly also a reactive system) where it holds for at least one expression that a certain impression or a temporal pattern of impressions is a necessary, but not a sufficient dynamic condition for its occurrence. The receipt of an order is a necessary impression to a "sales system", for the expression "delivery of the ordered goods", but it is not a sufficient condition.
3. An autonomous system is an OAS (possibly also a responsive system, but not a reactive system) where at least one expression is an action. A human being and most (if not all) organisations can be regarded as autonomous systems.

In conceiving something as an organisational system (OS), the domain most often comprises in practice an organisation of some kind. It could be an enterprise, an institution, a company, a factory, etc., or it could be a functional, geographic or organisational part thereof, typically a department or a specific kind of the business. As part of the domain it may also be relevant to incorporate a number of concepts generally relevant in an organisational context, for example public services, laws or other kinds of constraints imposed by society, or aspects of the particular professional field of the organisation.

However, for an OS it is generally relevant to consider the following kinds of things behind the system exposition:

- Actors - human actors as well as artificial actors and all kinds of symbiotic compositions of these two kinds.
- Actions (together with the associated goals) such that a (not exclusive) distinction is made between those influenced by impressions from the environment and those either directly constituting expressions of the system or only contributing to (or in some cases even explicitly counteracting) the expressions. Actions that are irrelevant for the expression of the system should be ignored in the exposition.
- Co-actions, i.e. co-ordinated actions performed by several actors together.
- Knowledge that is necessary for the actors to know the relevant pre-states of their actions and the respective goals. A goal may be situation dependent.
- Internal and/or external dynamic criteria for the initiation of actions (temporal, impressive and actor- or action-caused transitions).

- Communication between actors to ensure that they have the information necessary to perform their actions.
- How the knowledge in the organisation being of relevance for the system is represented as data in order to enable the preservation or communication of it. That includes all relevant aspects of the use of data technology and/or data-technical sub-systems to accomplish the preservation or communication.

In practice, some aspects of organisational culture, social norms, empathy (i.e. knowledge that cannot be properly represented), resources in general (energy, skills, intellect, etc.), ecology, economy, etc., may be added to this list.

The Sub-system Concept

It seems to be a widely accepted intuitive experience in the systemeering area that the notion of sub-system is a useful one. In this report, for example, "information system" is defined as "a sub-system of an organisational system".

However, when it comes to the point of being less intuitive and more explicit about the concept, there is little consensus about what really characterises a sub-system - or rather what should characterise it, if the concept shall be a useful one. The influence from the absoluteness of the "classical" system concept together with some apparent preference to associate the understanding of sub-system with the subset concept seem to be the main cause of the confusion.

The "old", simple interpretation of the concept system as being just "a set of interrelated parts", made it rather obvious to think of "sub-system" as: A subset of the parts together with an appropriate subset of their mutual relationships.

However, with the introduction of the notion that in order for something to be a system, it must have at least one systemic property, the matters became more difficult: Should the definition of sub-system then also involve the specification of a subset of the systemic properties? Intuitively this notion could be reasonable, and it may even work in some cases, but the problem is that this is not always so: Consider, for example, a well-functioning mechanical watch. It can be conceived to have the systemic property that under certain conditions it "shows the time". A possible sub-system of such a watch is the energy supplying device for the clockwork consisting of the spring, the winding knob, the exchange and click mechanism for tightening the spring, and a part of the frame to support these mechanical parts. The only sensible systemic property of such a sub-system is that it serves as a storage of mechanical energy. But then we have a serious problem with the subset notion applied on the systemic properties, because being an energy storage is in no way a subset of the systemic property of showing the time.

The problem of defining a sensible sub-system concept by means of subset relationships becomes even more difficult with the notion of a system as a subjective issue. Apart from the systemic properties not being absolute, but rather depending of the system viewer, one element in the system domain may now also potentially be viewed as several different system components in the system. Consider, for example, an organisation that is viewed as an organisational system and a person from that organisation: Here the person may appear as an actor of the type "Salesman" that is the agent of various sales activities. But independent hereof, the person may also be conceived as an entity of the type "employee" relevant in connection with calculations of salaries and the planning of sales campaigns. The person may even be regarded as an actand of the type "transport object" that is relevant in defining the activity "transport by car during sales trips". This causes the following question: Should a

possible subset relationship applied in attempts to define a sub-system concept then refer to the system domain alone, or to the system alone, or to both? It is certainly difficult to find logical or pragmatic arguments that universally justifies any of these choices. (For further aspects of the problems encountered when one is aiming at defining sub-system by means of subsets, see the more comprehensive discussion in [Lin90b].)

It is necessary to consider the sub-system concept differently - in fact, in a way that very well is in accordance with the way people intuitively apply it in practice. The "solution" is to realise that when viewing something as a system then only one system should be considered at a time. Applied here, either one must consider that which is regarded as the system or that which is regarded as the sub-system.

The advantage of this sub-system interpretation is exactly what appears to be the main positive feature of the intuitively applied concept: Depending on which level of detail as regard potential system components you want to consider, you can use the concept to encapsulate unnecessary details on a chosen level of abstraction.

Applied on organisational systems (OS) one obvious way to consider the relationship between an OS and a sub-system of it, is to conceive the sub-system equivalent with what an actor in the OS does (or a part of that). Typically a whole department (a possible system candidate in itself) may be considered a single actor in the OS, and (part of) what is done in that department in respect to other departments (i.e. possible systemic properties of the department system) may be conceived as a single action on the OS-level. A data-processing system may be conceived as a single (artificial) actor carrying out data-processing actions in the OS, even if we know that it, in fact, is composed of a lot of parts like CPU, database, programs, printers, VDU's etc.

6.3 Modelling Concepts

The discipline of ontology had been confined to the study rooms of a few philosophers for centuries. Only recently, in the early seventies of the 20th century, this situation changed dramatically, when people began to recognise the need to provide "good" models of organisational domains in the context of developing computerised information sub-systems, such as database systems, knowledge-based systems, expert systems, or decision support systems. The discipline of ontology has all of a sudden turned out to be of highly practical value for our modern society. The result of this development is an enormous number of modelling approaches and languages for information systems, brought forward - not by philosophers - but mostly by computer scientists or organisation scientists.

Meta-models

Those modelling languages are intended and used for representing, more or less precisely and comprehensively, models of information system domains, including of course the domain of communication within such a system. A common property of all modelling languages is that they are all based upon a structure of concepts, which reflects their particular view on the conceived "phenomena of the world". The conception of the structure of concepts on which a modelling language is based, and all constraints thereupon is given in the meta-model of that modelling language. Thus, a meta-model determines the way one may view, conceive or model the "world", i.e. it determines a particular ontological view.

A conception of a specific domain is called a base model (to distinguish it from the meta-

model). A type-instance relationship exists between meta-model and base model. That is, the meta-model determines all possible base models denotable in the modelling language.

A base model is often, for practical reasons, split into two parts: One part, the intensional model, specifies the possibilities and necessities of a domain only, while the other part, the extensional model, specifies the actual domain without the first part. A specific extensional model is called a population of the corresponding intensional model. A database schema is an example of an intensional model, and a database, being a population of this database schema, is an example of an extensional model. An intensional model can exist in its own right, without any corresponding extensional model specified. An extensional model however must be always used in conjunction with its corresponding intensional model. One extensional model relates to exactly one intensional model, but for one intensional model there may be many possible extensional models.

The intensional vs. extensional dichotomy is principally different from the meta-model vs. base model dichotomy. The first one distinguishes a representation of what is possible, from a representation of what actually exists, respectively. The latter dichotomy distinguishes between what is application-independent and built into the modelling language, and what is dependent on a specific application domain, respectively. To illustrate this point: The type-instance dichotomy may also be (and usually is) present within a base model. The traditional distinction between database schema and instances is an example of the intensional vs. extensional dichotomy within a base model. Not only the schema, but the complete base model, including schema, instances, and their interrelationship, is a population of the meta-model [BF91, OPF92] (see figure 6.3-1).

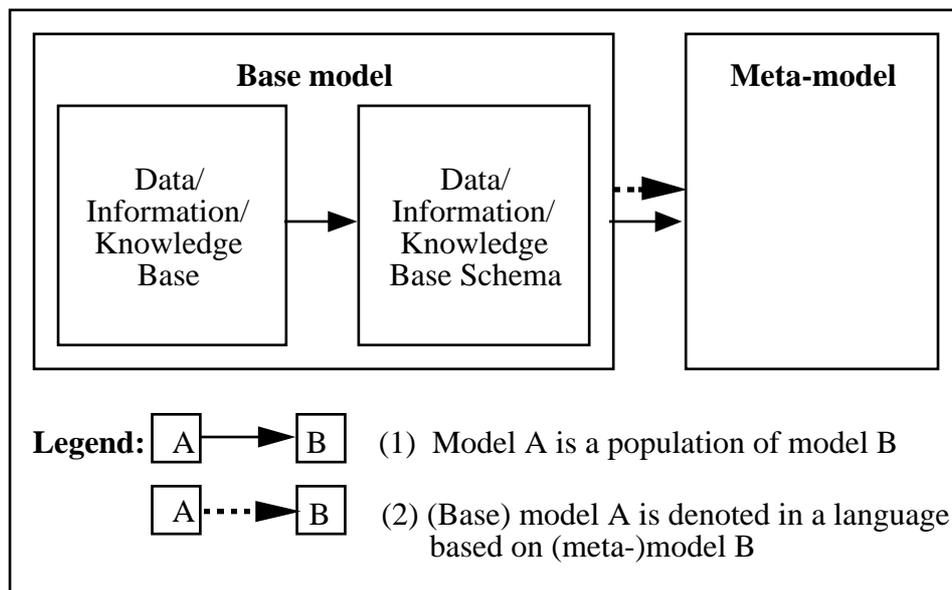


Figure 6.3-1: Two dichotomies: (1) Intensional vs. extensional; (2) Meta-model vs. base model

Note:

The concept pair extensional/intensional model can be applied across meta-levels as well: A base model, viewed as a whole, is always a population of the meta-model of the language in which the base model is described. A meta-model is always a population of the corresponding meta-meta-model, etc.

In this way, a hierarchy of extensional/intensional models is formed. It is only identical with the meta-level hierarchy, if the base model is not split at all (e.g. in semantic network approaches), or if the base model is viewed as a whole, that is, if the splitting of the base model into an extensional and an intensional model is ignored, for that matter. If however, this splitting of the base model is considered as part of the

extensional/intensional model hierarchy, as it is done in some approaches [e.g. ISO82], the situation is quite different. In those approaches, we find the following hierarchy:

- Extensional base model (e.g. data base)
- Intensional base model (e.g. data base schema)
- Excerpt of meta-model, referring only to the intensional base model (e.g. meta-schema)
- Meta-meta-model (of the excerpt only; e.g. meta-meta-schema)

In this case, the extensional/intensional model hierarchy differs significantly from the meta-level hierarchy. Neither the levels themselves are identical, nor the contents of the models above the intensional base model (e.g. meta-model vs. meta-schema).

Differences Between Meta-models

The probably most fundamental kind of difference between various meta-models is due to different global classifications of the conceived "phenomena of the world". Talking for example about "static" phenomena, some meta-models apply a classification into entities, attributes and relationships [e.g. Sun74, Che75], while others use only entities and relationships [e.g. Abr74, Fal76, NH89, Hal95]. Some meta-models provide furthermore specific, invariant classes of relationships, for example case models [e.g. Fil68]. Similar examples can be found in some semantic database models, where for example "ordinary relationships", like parent-child relationships, are distinguished (in the meta-model) from part-of relationships, and where the parts are said to be "aggregated into a complex thing" [e.g. SS77]. Other examples are the distinction between "ordinary entities" and points in space, in some meta-models which are used, for example, for CAD/CAM applications. The situation is quite similar looking at "dynamic" phenomena. Some meta-models distinguish between "ordinary entities" and points of time, in particular those used for planning-oriented or history-oriented applications [e.g. BFM79, Rol87]. Others work with special notions of events or transitions.

The essential question is how the conceived "phenomena of the world" should be classified, that is, what underlying ontological view should be chosen. There are many possible classifications, e.g., into "things" and "predicators", or "entities" and "relationships", or "entities", "attributes" and "relationships", or "entities", "events", "attributes" and "relationships", or "states" and "transitions", or "actors", "actions", "actands", or "changers", "changes" and "changed items", etc. All the other aspects of a modelling language, the constraining axioms, the specifications of the types of operations, as well as the concrete constructs of the specification language, are dependent on those choices.

A direct consequence of the multitude of different global classifications of phenomena found in the various meta-models, is that we find also different borderlines between meta-models and base models. Aspects appearing in one approach in the meta-model (and are thus built into the information system language), must be explicitly stated in the base model in another approach. An example may illustrate this point: If one is using a (time-less) entity-relationship model, and work with a history-oriented application, one needs to model the time axioms in the base model. If, however, one is using in this case a time-oriented entity-relationship model, one will probably find the time axioms already built in this meta-model.

There are furthermore a lot of terminological differences between the various meta-models, synonym and homonym situations. For example, "thing" in one meta-model may be exactly the same as "entity" in another one. "Entity" in one may be not the same as "entity" in another one. The one "entity" may be a sub-type of the other "entity", or they may have little or nothing at all in common. As an example, let us compare (time-less) entity-relationship models with time-oriented entity-relationship models. "Entity" in the time-less version covers

also points of time on the level of the base model. "Entity" in the time-oriented version does not cover points of time, since the notion of "point of time" is explicitly defined, besides "entity" itself, on the level of the meta-model. Thus, "entity" in the time-oriented version is a *sub-type* of "entity" in the time-less version.

Other differences do not have to do with the meta-model, but with notation. For example, in graphical notations all sorts of graphical symbols are in use for entities, relationships, events, etc. As with terminology, the choice of notation is highly subjective.

Another aspect which has to do with notation is the clustering of statements or constructs of the base model into "user- or designer-friendly" portions which need to be cross-referenced. As a matter of fact, there exists (probably) no modelling language which can express all phenomena relevant to information systems, and can totally avoid cross references. There are numerous ways to cluster those statements or constructs in some way. For example, if one compares fact-oriented with object-oriented approaches, one will find significant differences in the way statements are clustered (besides maybe other, more fundamental differences). Which way to choose is again subjective to a large extent, but there may be justifiable choices as well. There are some bad reasons for the mentioned differences. Competition among companies developing methods and tools for information system development is one of them. The widespread attitude of researchers in the information system field to create Yet Another Modelling Approach (the "YAMA Syndrome"), without addressing the question of its justification and its scientific value added sufficiently, is another one. The variety of existing modelling languages and dialects is therefore unnecessarily large.

Apart from a lot of subjectivity, a lot of arguments like "apples taste better than oranges", there are also some good reasons for differences, as far as substantial aspects are concerned. Different designers and users of an information system may view the world differently. Even more important is in this context, that for different kinds or different aspects of information systems, different modelling languages may be more or less suitable. For example, for a simple information system with a "snapshot" database, some simple dialect of a thing-predicator or entity-relationship modelling language may be suitable. In the case of a more sophisticated information system with history handling, trigger mechanisms and/or feedback mechanisms, such a modelling language may be less suitable. It may be that some aspects can be specified only in an inconvenient and cumbersome way. For such a kind of application case, other modelling languages, like time-oriented, event-oriented, or systems-dynamics-oriented ones, may be much better suited. On the other hand, the use of those latter modelling languages for simple "snapshot" database applications, would be an overkill.

A direct consequence of the multitude of different global classifications of phenomena found in the various modelling languages, is that we find also different borderlines between meta-models and base models. Aspects appearing in one approach in the meta-model (describing the underlying concepts of a modelling language, the relationships among them, and various constraints), must be explicitly stated in the base model (specifying the application domain) in another approach. An example may illustrate this point: If one is using a general (time-less) thing-predicator modelling language, and work with a history-oriented application, one needs to model the time axioms in the base model. If, however, one is using in this case a time-oriented thing-predicator modelling language, one will probably find the time axioms already built in the meta-model.

The Need for Meta-model Transformations

Different departments of an organisation serve usually quite different purposes. Thus, it is

likely that the various departmental information sub-systems belong to (partially) different kinds of systems. It will be therefore not reasonable to select and apply only one (reference) modelling language for the whole organisation. Even for one information sub-system, it may be appropriate to select different modelling languages for different aspects or perspectives of the system (e.g. the data, process, and behaviour perspective as distinguished in [Oll+88]). However, if one selects and applies various unrelated and incompatible modelling languages, communication will be hampered. In other terms, while the suitability aspect demands the use of a variety of modelling languages, the communication aspect demands that these modelling languages must be compatible and thus harmonised.

Organisations evolve. The goals or purposes of the whole organisation or of single departments may change, departments may be fused, split or re-organised, etc. The consequence is that the information system as a whole and/or the various departmental sub-systems must be adapted accordingly, to maintain the effectiveness of the communication. Not only the concrete base models must be adapted (first-order evolution), but also the modelling languages to specify these models (second-order evolution), due to changing suitability requirements [e.g. FOP91, FOP92, OPF92, Pro74]). This adaptation and evolution of modelling languages must be performed in an upward-compatible way. That is, a change should result in base models which are compatible and consistent with the new modelling language. Otherwise severe communication problems will occur.

The number of modelling approaches used for the design of information systems has been growing dramatically. The questions as to why such a variety of approaches is necessary at all, what the differences between or similarities of the various approaches are, and whether or not it might be possible to determine a conceptual foundation common to all (existing and possibly future) modelling approaches, have become urgent [Lin90a].

The aspect of suitability of modelling languages leads to the conclusion that a reasonable diversity of modelling languages is necessary. Currently however, the diversity of modelling languages is too large, and differences between them are often unjustified.

The conclusion from this line of arguments is that harmonisation of modelling languages is absolutely necessary.

Transformation of Meta-models

One approach to achieve harmonisation of modelling languages could be to provide an all-embracing reference modelling language, which covers the notions and constructs of all the various existing modelling languages, and to provide suitable translation mechanisms between the reference modelling language and the others. Attempts of this sort have been made [e.g. Ama87, Oll+88]. The major problem of such an approach is that it is easy to create *yet another*, even more sophisticated modelling language whose notions and constructs are (partially) not covered by the reference modelling language, and thus not (entirely) compatible with it.

A more successful approach is to give up the idea of a reference modelling language for good, and to provide a suitable, open *ordering and transformation scheme* for the modelling languages, based on the idea of *meta-model transformations*. In such a scheme, the various meta-models are specified, as well as the transformational relationships between them, resulting in a *system of interrelated meta-models*.

An approach to such an ordering and transformation scheme will be sketched underneath. The

conceptual background and the principles of this approach will be presented, as well as some illustrations and practical considerations.

Relevant Characteristics of Modelling Languages

Modelling languages can be compared on the basis of many different criteria or characteristics [e.g. Law88]. Criteria or characteristics may either depend on, or be independent of a particular application or situation. Some of them can be determined in an objective way, and others are more subjective [HO92]. The characteristics of modelling languages that are most important for the meta-model transformation scheme, are the following ones:

- **Expressiveness**
is a quality stating to what degree a given modelling language is capable of denoting the models of any number and kinds of application domains.
- **Arbitrariness**
is a quality stating the degrees of freedom one has when modelling one and the same application domain. The more liberal a modelling language is, the more semantically equivalent models of one and the same domain can be found. If the modelling language allows one and only one model for each domain to be modelled, the arbitrariness is zero. Such a modelling language is called deterministic ([Fal93a, Fal93b]).
- **Suitability**
is a quality stating to what degree a given modelling language is generally applicable, or specifically tailored for the particular task of modelling a specific kind of application domain.

It is neither always possible nor desirable, to provide as much as possible of each of these qualities. There will be often a balance or a trade-off. For example, there may be a special-purpose modelling language or language which is highly suited for that special purpose and for a specialist in that special field, but which has a low overall expressiveness. There may be a general-purpose language with a high expressiveness, but which is only moderately well-suited for many or most purposes. Very often, a trade-off between expressiveness and suitability is required. A modeller who likes his artistic freedom will probably prefer a highly liberal language (one with a high degree of arbitrariness). If one is aiming however at a *discipline* of modelling, artistic freedom and a liberal language will be undesirable. Rather one would prefer an entirely *deterministic* language.

A domain can, in general, be modelled in various different ways, using the same (non-deterministic, liberal) language or different languages to denote the resulting models. In order to be able to determine whether or not two model denotations refer to models of exactly the same domain, one has to establish the notion of equivalence of models. Two model denotations (a) and (b) are said to be equivalent, if and only if the one model denotation (a) can be transformed into a model denotation (b), and this model denotation (b) can be transformed back into the original model denotation (a) without any loss, by means of appropriate transformation rules, and if and only if the same holds also for the inverse pair of transformations. This statement applies regardless of whether the models are denoted in the same or in different languages, and regardless of the meta-level of the models. Whether equivalence of models and model denotations can be established with confidence in practice, depends both on the syntactical correctness and the semantical validity of the transformation rules.

Basic Transformations Among Meta-models

In the meta-model transformation scheme, the various modelling languages are formally related on the basis of their meta-models. A set of basic transformations among meta-models is defined, which allows to relate any possible meta-model, and therefore any existing or future modelling language. Thus, the meta-model transformation approach provides an open ordering and transformation scheme among meta-models.

Each of these basic transformations is motivated by its influence on the mentioned characteristics of modelling languages, expressiveness, arbitrariness and suitability. The following basic transformations are distinguished. More complex transformations can be constructed by composing several basic ones.

Specialisation

The first transformation among meta-models is motivated by providing the possibility to influence suitability of modelling languages. A modelling language can be made more specifically tailored for modelling a particular type of application domain by introducing more specialised concepts. For the meta-model of such a special-purpose modelling language, new concepts have been added to the concept structure, which are derived from or defined in terms of already existing concepts. The resulting meta-model is called a specialisation of the meta-model of the more generic modelling language. Note that a specialisation does *not* affect the *expressiveness* of a modelling language, it only facilitates the modeller in describing application domains in more compact base models by using new concepts which are derived from already existing concepts.

For illustrating the notion of specialisation, consider as an example a simple thing-predicator modelling language. If one uses this modelling language for designing a database about a time- and history-oriented application domain, a thing type 'Time-point' will certainly occur in the base model, being an instance of the concept 'Thing-type' that is part of the meta-model of thing-predicator models. This thing type 'Time-point' will have ('Begin' and 'End') relationships with some "ordinary", non-time thing types whose instances are to be marked by points of time to indicate their period(s) of existence. However, should one use a specialisation of thing-predicator models which provides the facility of time and existence indications in its meta-model as a specialised notion, then it will be more convenient to represent the time- and history-oriented application domain in a base model. This base model in terms of the time-oriented meta-model will be more compact than a base model in terms of the general thing-predicator model.

Thing-Type	Thing-instance
is-type-of	is-instance-of
Person	P1
Department	D1
Membership	M1
Time-point	T1
Time-point	T2
Begin	B1
End	E1

Thing-instance	Thing-instance	Predicator -of-thing
composed-of	plays	is-played-by
M1	P1	is-member-of
M1	D1	has-member
B1	M1	has-begin
B1	T1	is-begin-of
E1	M1	has-end
E1	T2	is-end-of

Legend:

Two-layer table headers: Part of a meta-model
 First layers: Thing types of meta-model
 Second layers: Predicators of meta-model
 Table bodies: Sample population of meta-model = Model including instances (specific application)

Figure 6.3-2: General, simple meta-model including a sample population (specific application)

A small example of this sort of specialisation is shown in figures 6.3-2 and 6.3-3. Figure 6.3-2 shows (part of) the meta-model of a very simple thing-predicator model version, and a population, being a sample base model. The meta-model can be specialised into the one shown in figure 6.3-3, by means of appropriate rules, such as: Delete 'Time-point', 'Begin' and 'End' from the population of 'Thing-type'; delete the corresponding instances (in this case 'T1', 'T2', 'B1', 'E1') as well as the predicators 'has-begin', 'is-begin', 'has-end', and 'is-end' from the population of the meta-model; create instead the two relationship types 'Begin' and 'End' in the meta-model, and place the deleted instances there accordingly. 'Non-time-thing-instance', 'Time-point', 'Begin', and 'End' are sub-types of 'Thing-instance', while 'Non-time-thing-type' is a sub-type of 'Thing-type'. These sub-types are not shown in figure 6.3-3.

It is evident that while the time-oriented meta-model is less compact than the more general meta-model, a base model specified in terms of the time-oriented meta-model is generally more compact than a base model specified in terms of the more general meta-model.

Furthermore, in a time-oriented modelling language, rules about time and time relationships (time axioms) are included in the meta-model, to exclude base models which make no sense. In the more general-purpose modelling language, these time axioms should be specified explicitly in the base model. This leads to the next kind of transformation between meta-models.

Restriction

The introduction of new concepts, most often, goes together with some general rules concerning these new concepts. Furthermore, it may be desirable to restrict the freedom to model one and the same application domain by means of different, but equivalent base models. Both situations require a restriction of the possible set of base models. In other terms, one requires to diminish the arbitrariness of a modelling language. To arrive at the meta-model of such a less liberal, or more deterministic, modelling language, means to add new constraints to the existing set of constraints being imposed on the concept structure. The resulting meta-model is called a restriction of the meta-model of the modelling language offering more arbitrariness.

For example, to achieve a decent time-oriented modelling language, time axioms must be added after introducing time-stamping as a specialised notion, such as: An instance of a 'Non-time-thing-type' must be born before it can die; births and deaths of an instance of a 'Non-time-thing-type' must alternate; etc. (compare figure 6.3-3) [e.g. FW88, Fal93a].

Another example has to do with the different dialects of thing-predicator models concerning the "arity" of relationship types. Besides an n-ary dialect [e.g. Fal76, NH89, Hal95]), which allows any number of entity types involved in a relationship type, also a binary dialect of thing-predicator models exists which allows only binary relationships [e.g. Abr74]. The binary dialect can be considered as a restriction of the n-ary dialect of thing-predicator models by adding a constraint which restricts the possible occurrence frequencies of 'Thing-instances' in the predicator 'composed-of' to two (compare figure 6.3-2).

It should be realised that the addition of a constraint does not necessarily affect the expressiveness. In the above example, for each base model in the n-ary dialect, an equivalent base model with only binary relationships can be constructed, and vice versa [e.g. Fal77, NH89, Hal91, Hal95]).

Degeneration

In order to cause a decrease in expressiveness of a modelling language, it is necessary to restrict the possible set of domains that can be modelled by the concepts in the concept structure. A simple possibility to arrive at the meta-model of such a less-expressive modelling language, is to remove a concept from the concept structure, such that the domains that could be modelled by the removed concept, cannot be modelled in another way by concepts from the remaining concept structure. The resulting meta-model is called a degeneration of the meta-model of the modelling language being more expressive. Note that a degeneration is different from the inverse of a specialisation, because a specialisation does not affect the expressiveness of a modelling language.

Non-time-thing-type	Non-time-Thing-instance
is-type-of	is-instance-of
Person	P1
Department	D1
Membership	M1

Non-time-thing-instance	Non-time-thing-instance	Predicator-of-thing
composed-of	plays	is-played-by
M1	P1	is-member-of
M1	D1	has-member

Non-time-Thing-instance	Time-point
has-begin	Is-begin-of
M1	T1

Non-time-thing-instance	Time-point
has-end	is-end-of
M1	T2

Legend:

- Two-layer table headers: Part of a meta-model
- First layers: Thing types of meta-model
- Second layers: Predicators of meta-model
- Table bodies: Sample population of meta-model = Model including instances (specific application)

Figure 3.3-3:

Time-oriented meta-model, being a specialisation of the meta-model in figure 6.3-2, including the same sample population (specific application)

A degeneration in our small example would be, e.g., to eliminate the notion of 'Time-point' from the time-oriented meta-model in figure 6.3-3. Only the static aspects of this domain could then be expressed, i.e. the lower two tables would disappear.

Influence of Meta-model Transformations on Language Characteristics

The basic transformations being introduced are necessary and may be sufficient to relate all possible meta-models of existing or future modelling languages, given that there exists a meta-model with a sufficient expressiveness. In order to prove this and other claims, it is necessary to choose a suitable formal language for specifying meta-models and the transformations

among meta-models. For the choice of such a suitable meta language, the same problems are faced as for the selection of a language for a specific domain on the application level.

In most practical cases, modelling languages differ in more than one of the characteristics, suitability, arbitrariness and expressiveness. This is due to the fact that the meta-models of these modelling languages are related by means of a combination of specialisations, restrictions, and/or degenerations.

Specialisation and restriction do not necessarily influence the expressiveness of modelling languages. Specialisation adds new and more specialised concepts to a modelling language, but because these new concepts are defined in terms of already existing concepts, the expressiveness of the modelling language is not influenced (see figures 6.3-2 and 6.3-3). Restriction diminishes the number of possible base models by adding constraints imposed on the concept structure. However, as the same domain may be modelled by more than one (equivalent) base model, the number and kind of domains that can be modelled, is not necessarily diminished. An illustration has already been given in the comparison of an n-ary dialect and a binary dialect of thing-predicator models.

Specialisation, restriction, as well as degeneration of general-purpose modelling languages lead usually towards more-special-purpose modelling languages. The latter are based upon concepts which are more specialised for a specific situation than the concepts forming the conceptual basis for general-purpose modelling languages. Furthermore, situation-specific constraints which have to be specified explicitly when using a general-purpose modelling language, are usually already part of the special-purpose modelling language and thus need not to be specified explicitly when using the latter one.

General-purpose and special-purpose modelling languages do not necessarily differ with respect to expressiveness. However, the usage of a modelling language, being specific for the type of application domain in question, may be more convenient and may lead to more compact base models. Of course, it is still possible (and often the case in practice) that the set of domains that can be modelled by a special-purpose modelling language, is a subset of the domains that can be modelled by a general-purpose modelling language.

Construction of Systems of Meta-models

The basic meta-model transformations provide the means for relating modelling languages. A system of meta-models may have the structure of a general hierarchy with one root. This root should be a meta-model of a very simple and basic, but highly expressive modelling language. Transformations (composed of the basic set of transformations) can be applied to the root meta-model, such that meta-models of more specialised, restricted, and/or degenerated modelling languages are created step-wise, whereby their transformational relationships to the root meta-model, and thus to each other, are always well-defined. The meta-models of any existing or future modelling languages can thus be placed in this hierarchy, provided the modelling language behind the root meta-model has sufficient expressiveness. The meta-model transformation scheme is an open scheme since it allows hierarchies of arbitrary breadth as well as of arbitrary depth. The meta-models in the hierarchy are upward-compatible (with respect to expressiveness).

Given a finite set of modelling languages, a top-down procedure or a bottom-up procedure can be followed to relate these modelling languages. Following a top-down procedure, a meta-model is chosen which is at least as expressive as any of the meta-models under consideration. Upon that meta-model, compositions of specialisations, restrictions, and degenerations are defined, until this results in a system of meta-models (of existing and possibly not-yet-existing

modelling languages) which relates all meta-models of the target modelling languages. Alternatively, a bottom-up procedure can be followed, which starts by analysing the meta-models of the target modelling languages, and works towards a root meta-model, relating all meta-models in consideration by means of inverse specialisations, inverse restrictions, and/or inverse degenerations. The choice of the procedure may depend on the ease of identifying a suitable root meta-model *a priori*.

It is realised that alternative hierarchies can be constructed to relate a finite set of meta-models. For example, it is in principle possible to introduce new concepts in a meta-model, which are of no importance for arriving at a target meta-model. For pragmatic reasons, this situation is certainly undesirable. It is therefore forbidden to introduce new concepts or constraints (in intermediate meta-models) which are not used in further specialisations, restrictions, or degenerations. Note that this rule excludes a "broad" or "universal" root meta-model with a large variety of concepts. Furthermore, a priority order on transformations is imposed. In the case of the top-down procedure, the order of transformations is specialisation first, then restriction, and finally degeneration (if desired).

The acceptance of a system of meta-models for practical use depends on the validity of the meta-models and the transformations defined in that hierarchy. A possible validation procedure is to populate all meta-models in the system, using one and the same particular, significant application domain, and the subsequent cross-checking of the various corresponding and supposedly equivalent base models.

A more sophisticated example (than the one illustrated in figures 6.3-2 and 6.3-3) of constructing a system of meta-models is, in fact *our conceptual framework itself*. We start out with a root meta-model (the most fundamental layer, sections 3.2 or 4.1), and introduce various specialisations and restrictions to arrive at the layer of actors, actions and actands (sections 3.3 or 4.2).

A question of practical importance is of course, how to find a global root meta-model with an expressiveness sufficient to cover all aspects of modelling. The fundamental layer of our conceptual framework (section 3.2) is certainly quite a suitable starting point for this search.

Practical Use of Meta-model Transformations

The meta-model transformation scheme can be utilised for the harmonisation of the large variety of existing modelling languages, for their comparison and integration, as well as for supporting the evolution of existing modelling languages. That is, not yet existing, future modelling languages can also be created following this approach. The meta-model transformation scheme can be applied in the following (research) areas, among other possible areas:

Given the characteristics or contingencies of an application domain, the meta-model transformation scheme may serve as a means to compare a set of modelling languages and to select the proper modelling language. The characteristics of the domain are matched with the properties of the modelling language. The properties of different modelling languages can be identified by positioning and comparing the considered modelling languages in the system of meta-models.

Method engineering [KW92, SB93] is the engineering discipline to design, construct and adapt information system development methods. Methods tuned to a specific situation are called situational methods [HBO94]. The construction of situational methods out of fragments of existing methods requires to have meta-models of these fragments available in a so-called method base. A particular modelling language is an example of such a method fragment. To

arrive at a consistent situational method by assembly from these method fragments, horizontal integration (i.e. integration of fragments for different phases in the information system development life-cycle) and vertical integration (integration of fragments for the same phase, but focussing on a different perspective) of method fragments are required. The meta-model transformation scheme can be used for both horizontal and vertical integration.

First-order evolution [FOP91, FOP92, OPF92, Pro94] occurs in systems in which the meta-model and the corresponding modelling language(s) are stable, but each and every aspect of the base model can change, including application-specific types and laws (e.g. schema evolution), without the need to install a new system and thereby interrupt the primary processes in the organisation. Second-order evolvability is particularly important for large organisations with various sorts of applications, where also new sorts of applications become necessary from time to time. If such an organisation applies the meta-model transformation approach, specific transformations can be performed on the meta-models of an existing system of meta-models, to create new meta-models suitable for new sorts of applications, or for new aspects within existing base models. All the old base models or old parts of base models, and the actions thereupon will continue to work, and the primary processes in the organisation will not be interrupted.

Summary

Summarising, the meta-model transformation scheme can be characterised as:

- an approach towards harmonisation of modelling languages,
- providing a method to relate their meta-models
- by means of a basic set of transformations: specialisation, restriction, and degeneration,
- each influencing expressiveness, arbitrariness, and suitability of the languages,
- and which can be used for comparison, integration, and evolution of modelling languages.

La critique est aisée, et l'art est difficile.
 (Criticism is easy, and art is difficult.)

Philippe Néricault Destouches, *Le glorieux*

7 REFLECTIONS: Committed Positions on the Report by Individual FRISCO Members and Associates

This chapter presents a compilation of essays and position papers on the FRISCO theme. We felt that it is necessary to include in a report treating a difficult theme like this one, not only the reasonably common view we finally arrived at and which was presented in the previous chapters, but also reflections on and controversial views of individual FRISCO members and associates. Everything stated in this chapter is strictly on an individual, personal basis. The order of this compilation is alphabetically, according to the author's name.

FRISCO – “Mission * Impossible” ?

by Eckhard D. Falkenberg

A colleague, not being a member or associate of the FRISCO task group, once told me that he admired our courage to try to tackle a difficult task like this, but he did not think that we could possibly succeed. Now having the final result in front of me, I wonder whether or not his expectation was right. To find an answer, it will help to look for a moment back to the early days, when we discussed the objectives and scope of our group.

The initial objectives were

- to provide an improved paradigm for scientific and professional work in our field, and
- to propose a coherent and agreeable system of concepts for information systems, with
 - proper definitions and
 - a professional terminology.

The initial scope was

- to address information systems in a wide sense;
- to address the question, whether for different kinds of information systems different kinds of conceptual frameworks are needed, or only one conceptual framework;
- to address only the conceptual level of information systems;
- to address all classes of concepts relevant to this conceptual level, e.g. semiotic concepts, system concepts, architectural concepts, modelling concepts;
- to address and integrate all perspectives, e.g. data, process, and behaviour perspectives [Oll+88].

In the beginning, it was doubtful how far we should treat meta-models and modelling methods for information systems. Modelling methods were eventually *excluded* from the scope. Meta-models were initially left out as well, since we felt that the overwhelming number of competing meta-models in our field would make it difficult to say anything reasonable about them. We felt that trying to deal with the diversity and variety of those meta-models would be like "opening *Pandora's box*".

I will now discuss some of our results in the light of our initial objectives and scope. I will first discuss some of the *expected* results (a), and then some of the *unexpected* results (b).

Finally, I will discuss some *open* issues which were not or not sufficiently tackled by the group (c), and will draw some conclusions (d). I should emphasise that there are other results and open issues which I will not discuss here.

a) SOME EXPECTED RESULTS

a1) A Necessary Evil: Choice of an "ism"

I happened to be a member of ISO TC97/SC5/WG3, the working group who wrote the report "Concepts and Terminology for the Conceptual Schema and the Information Base" [ISO82]. In that working group, there was one member who was regularly disagreeing with most of what was said in the meetings. He brought forward many arguments and proposals how the issues should be tackled and how the concepts should be defined, without ever being understood by the majority of the other members, and consequently experiencing an extremely low level of acceptance. Only very late, the reason for all that hassle was found: He *tacitly* had been adhering to quite a different *philosophical position* than the rest of the working group!

In the FRISCO task group, we had problems of this kind as well, and sometimes I have the impression that some of them are still lingering on. Nevertheless, being aware of this kind of problems from the beginning, we discussed the possible philosophical positions at a fairly early stage. Eventually, objectivism (naïve realism) was ruled out, and a form of constructivism was chosen. As proper as this choice may be for the purpose at hand, it has a peculiar disadvantage: Probably most people are naïve realists, and the colloquial use of natural language is accordingly. People tend to speak "naturally" about "things in the real world" (vs. "imagined things"), about "concrete" (vs. "abstract") things, etc. For a naïve realist, an explanatory definition like E1 in chapter 3:

"A thing is any part of a *conception* of a domain (being itself a "part" or "aspect" of the "world")."

is presumably difficult to understand or accept, since this definition reduces his beloved "concrete and real things of the real world" to mere constructs in his mind. He may be quite puzzled by this definition, and he might eventually ask the question: "What then does a world independent of our cognitive and intellectual capabilities consist of, if things are conceptions?" The answer is easy for a constructivist; he will say: "Nobody knows! All what counts is how we perceive and conceive the world." But a naïve realist will probably be even more puzzled by such an answer.

Not only naïve realists will find it difficult to understand or accept our report, but also those people who expect plain "truth in concepts". According to our form of constructivism, seeking "truth in concepts" is a hopeless venture.

Nevertheless, our choice of constructivism has been in my opinion very fruitful for the FRISCO task, in particular since that choice triggered the development of ideas which I consider to belong to the *unexpected* results of our work (see point b2 below).

a2) Bread and Butter: Modelling Exercises Galore

The problem of choosing a reasonable ontological view or meta-model became relevant, as

soon as we left the early phase of defining concepts "out of the blue" in an *ad-hoc* way, and began to think about a more *systematic* way of cutting through the "jungle" of concepts. Since a large part of our task had been in fact a modelling exercise, we simply had to have a reasonable meta-model as a basis for that modelling exercise; and to create a meta-model is *yet another* modelling exercise. Almost inevitably and not at all surprisingly, the numerous views of the group members on that issue collided. But eventually we managed to do these (meta-) modelling exercises, and I'm still quite surprised that we succeeded at all, regardless of what one might think about the quality of our models. I guess that this was possible only due to one of the *unexpected* results we achieved (see point b1 below).

My point here is that if you look at the various portions of these models separately and independently, you will find hardly, or only sometimes, anything new or exiting, depending on *your* views. You have to look at the *interrelationships* between the various portions, to understand what goes beyond "bread and butter" (see point b1 below).

a3) What's in a Name? : Choice of a Terminology

Differences in terminology have been a significant source of problems in our work, in spite of the fact that we were fully aware of them from the beginning. Eventually, some preferred terms were chosen for the various concepts, recognising of course that many of those terms are used in the literature in the same way, in similar ways, or in completely different ways. If you look at a specific term in an isolated way, you may find it well-chosen, old-fashioned, inappropriate or whatever, depending on *your* views. The terminology we chose is relatively conservative. I guess that the average buzzword follower will be utterly disappointed by our report.

What really counts is the overall picture, the conceptual framework as a whole, and not the choice of a single term.

b) SOME UNEXPECTED RESULTS

b1) A Discipline Emerging:

From Separate "Pillars" to an Integrated and Coherent Conceptual Framework

At a certain stage, the metaphor of the FRISCO "*temple*" supported by three "*pillars*" came up, namely the semiotic pillar, the system pillar and modelling pillar. Although not a bad metaphor as such, when trying to further develop and refine our conceptual framework on the basis of these pillars, it soon became clear that there are too many unclear interrelationships and cross-references between them, and the whole temple was eventually replaced by a "*monolithic, pyramid-like*" structure. This *breakthrough* came only as late as 1994, and it was - as it is often the case with breakthroughs - unexpected in the light of our initial objectives and scope.

I regard this breakthrough as our most important achievement.

What does it mean to have this monolithic structure? It means essentially that we are in fact *not* talking about *different* disciplines, semiotics, system science, ontology, and may be more, but about *one* discipline only! It may best be called *the information system discipline*.

The pyramid metaphor fits much better than the temple-pillar metaphor to this result. As you know, many of the temples built with pillars have crumbled during the centuries, whilst most of the pyramids are still standing upright!

b2) Pandora's Box Wide Open: Meta-model Transformations

Meta-model transformations have been playing quite a peculiar role in the FRISCO task. They were not part of our initial scope, they entered the picture much later. In fact they were subject of quite another project *outside* FRISCO, which however, at least partially, was *triggered* by the FRISCO work [HO92, OF94, Oei95]. They entered the picture due to the need to resolve conflicts between various views held by different members of our group. A typical example of those conflicts was the following one:

According to our initial scope, we discussed of course also the dynamic aspects of information systems. As usual, there were different views on this subject in the group. Some members preferred the view that one has in this respect basically *three* fundamental concepts, namely "actor", "action" and "actand" (or "changer", "change" and "the changed items"). Some other members preferred the view that one has basically *two* fundamental concepts, namely "state" and "transition". Both parties had quite good, but of course *different* arguments in favour of their respective views. Who was "right"?

Applying the idea of meta-model transformations, the answer turned out to be quite simple and eye-opening: Both were "right"! The respective meta-models are *not conflicting* at all, once one realises that the one meta-model (the actor-action-actand meta-model) can be explained and defined as a *specialisation* of the other one (the state-transition meta-model), and that they consequently have to be placed on *different* (but in this case adjacent) levels within a *hierarchy* of meta-models.

I regard these resolutions of view collisions - which occurred quite often - as another major breakthrough. What initially looked like genuine conflicts of views, often (not always) turned out to be merely a matter of more or less specialisation, restriction or degeneration. I guess that *most* of the debates and disputes within the information system community on modelling issues and meta-models in the last 25 years or so, have been (and still are!) due to those *pseudo*-conflicts. What a waste of time and effort!

Our conceptual framework eventually was constructed, not necessarily according to every detail, but certainly according to the *spirit* of the meta-model transformation approach.

With this kind of approach in mind, there is no reason to be afraid of "opening Pandora's box" and facing the overwhelming number of views and meta-models in our field.

I guess that the other major unexpected result (see point b1) was only possible on the basis of *this* result. If we would not have been able to resolve view collisions *in the first place*, I doubt that we ever would have achieved an integrated and fully coherent conceptual framework.

c) SOME OPEN ISSUES

c1) In the Clouds: Fuzzy Spots in the Picture

Most aspects of the FRISCO conceptual framework are formalised.

I think that for the clear and full understanding of concepts, formalisation is certainly not a *sufficient* condition, but it may help a lot. On the other hand, if I find that certain aspects of a conceptual framework are *not* formalised, I become suspicious. Why are they not formalised? Is it because the authors find that a formalisation is "impossible" or too difficult, or simple but beyond their scope of interest? If the authors find it "impossible" or too difficult, is it because they did not dig deep enough to bring about a clear and full understanding?

There is in particular one concept informally introduced in chapter 3, which is not even brought near to any kind of formal treatment: "*Semiotic level*" [E25]. Is this concept "principally unformalisable", or is its analysis not yet deep enough? Is it sufficient to mention a few examples of aspects on each "semiotic level", to make the concept clear enough? How are the distinctions between the various levels justified, other than by some more or less convincing examples? All these are questions to which I still don't know the answer.

My impression is that the metaphor of the "*semiotic ladder*" is quite well-chosen. The ladder must be very high indeed: The higher you climb, the more nebulous the views get.

c2) My View or Your View?: Modelling Uncertainty

There is a general phenomenon occurring frequently in the context of modelling any domains, which has been called "modelling uncertainty". Given presumably one and the same domain, different modellers may come up with different models (denoted in some way). A further discussion among these modellers may eventually reveal that they constructed *different* models of *one and the same* domain indeed. Modelling uncertainty is by the way a prominent reason for choosing a *constructivistic* position as a basis for discussing information systems, and to reject an *objectivistic* position for this purpose, since modelling alternatives do not fit into the simple world of an objectivist.

Even in our relatively simple test case (chapter 5), this issue popped up and made it necessary to provide *some* (partial) model of the test case before interpreting it in terms of FRISCO.

Theoreticians often argue that the discussion about modelling alternatives is superfluous, since transformations exist between these alternatives. In practice however, things look quite different (as usual). In large-scale information system development projects, with several designer teams and interest groups involved, the problem to choose one out of several modelling alternatives arises quite frequently. It is practically very cumbersome to provide transformations between the various models. One model alone is usually already quite laborious to create and denote. These kinds of problems become even worse in projects where different designer teams are analysing partially different and partially overlapping domains. View integration of these overlaps may be severely hampered, if too many modelling alternatives are provided by the various designer teams. It may be even difficult to identify these overlaps at all. For these practical reasons, it is highly desirable to have a method that allows to choose one and only one modelling alternative deterministically and in a reproducible way, based on reasonable criteria [Fal93a, Fal93b, Hal93].

To the best of my knowledge, this kind of problem has been hardly tackled in our field. This issue would have fitted very well into the scope of the FRISCO task, in particular in the context of constructivism, meta-models and meta-model transformations, but it was never included. Too overwhelming were all the other problems the group was tackling. It remains an open issue.

d) MISSION ACCOMPLISHED?

Regarding our initial *scope*, I think that the FRISCO report addresses information systems indeed in a fairly wide sense, although some people might think that it could have been wider. It became evident already in the early phases of our work that for different kinds of information systems, only *one* global, basic conceptual framework is needed. The various

classes of concepts and perspectives we initially aimed at are also reasonably well treated in the report, although not every detail the one or the other reader might look for is covered.

The question as to whether or not the report meets our initial *objectives* is more difficult to answer. In a time when every little softwarehouse in Texas or around the world calls an improvement of the functionality of its products a "paradigm shift", it is a bit dangerous to use that term at all. Nevertheless, did we manage to provide a *genuinely* improved paradigm for scientific and professional work in our field? Well, not everything we came up with is really all that exiting and new. We certainly didn't invent constructivism. Many of the aspects of our conceptual framework, viewed in isolation, are not new at all. The terminology we are using is mostly on the conservative side.

But on the other hand, the points I mentioned under unexpected results (b1 and b2) might indeed be considered as an improved paradigm. The FRISCO conceptual framework covers quite a broad spectrum. The relevant aspects of several disciplines, whose interrelationships and mutual coherence were initially quite fuzzy, are now embedded in a *consistent and fully coherent* conceptual framework. The problems of inter-operability of meta-models and modelling approaches have been brought closer to a solution. The "peaceful co-existence" of different ontological views has now become a real possibility.

My conclusion is that the expectation of my colleague, who doubted that the FRISCO task group could possibly succeed, was too pessimistic. However, only time will tell whether we really met our objectives.

Is Information to Become the Phlogiston of the Late 20th Century? *Reflections on Information, Information Systems and Information Society*

by **Wolfgang Hesse**

In one of the past ISCO conferences, Hans-Erik Nissen referred to the phlogiston theory which was developed in chemistry in the 16th and 17th century. It is an example of a misleading scientific approach which was abandoned and forgotten as soon as other, more consistent and viable theories were developed [Nis92]. Nissen warned us to have a careful look on information system concepts in order not to design another, modern phlogiston theory. In fact, terms like „information“, „information systems“ and „information society“ are presently used in an inflationary manner, often repeated mechanically without any reflection and obscured by myths.

Let us start by listing some of these myths:

- Myth 1:** Information can be taken as a *synonym for data*.
- Myth 2:** Information is just *objects*, nominalisable, like matter and therefore can be bought and sold like other material products.
- Myth 3:** Human behaviour and acting can be *completely explained and described by information processes* and thus (at least principally) be *computerised*.
- Myth 4:** Information systems are pure technical artefacts. The one who has the best computers, networks, hardware and software, is the winner in the race for market penetration and profit making.
- Myth 5:** Computer links, Internet access for everybody, world-wide information and communication services will result in an *"Information Society"* which will be better informed, have better communication, be more democratic, less hostile, and offer better conditions for a human life to every individual.

In my opinion, the FRISCO approach - and particularly its philosophical foundation on semiotics and ontology - lays a good ground for a sound, consistent, stable, and de-mystified position concerning these issues. In the following, I will focus on three points:

- (1) the (quaternary)*sign relationship* in our semiotic approach, particularly with regard to the data / information dichotomy,
- (2) the *dynamic nature of "information"* - particularly in contrast to current tendencies for objectification and nominalisation both in computer technology and in the society,
- (3) a *holistic view of information systems* which results in a human-centred position (instead of a technology-centred one) concerning their social context and future development.

Ad 1: The Semiotic Tetrahedron and the Data / Information Dichotomy

The FRISCO group has chosen a four-corner version of the semiotic diagram and thus extended the former triangles to a tetrahedron (cf. chapter 3, figure 3.4-2). Placing the *actor* in the centre of the figure has not happened by accident but reflects the constructivist overall approach: There is no sense in considering terms like world, thing, time, etc., independent

from their *observers*.

No corner of the semiotic tetrahedron does make sense for itself. A *representation* (or sign token) is chosen by someone (a *representer*) to "represent" his *conception* of something (the *domain* or sign object) to someone else (an *interpreter*). The relationship between representation and domain, i.e. the question which domain is referred to by the representation or whether there is such a domain at all, is determined by the actors involved. Thus it depends on the actors (on the sender as well as on the receiver side) and their conceptions. Different actors may refer to the same domain by different representations or associate different conceptions with the same representations and thus assign different domains to these - all this leading to the usual mis-understandings and controversies in every-day life.

This means: There are no fixed sign relationships but they are always relative to the involved actors, the representer (sender) as well as the interpreter (receiver). This draws our attention to the nature of actors. Following FRISCO, these may be devices as well as persons (cf. chapter 3, definition [E13]). But, in order to be able to observe and perceive "parts or aspects of the world with their senses" they *have to be human beings* (chapter 3, assumption [b]). This ability attributed to a "device" can only be imagined as a *delegated* one: Human actors may delegate their representing and interpreting responsibilities to machines by programming them using their own understanding. In this case, the sign relationships drawn by such devices are nothing else than mirror images of the sign relationships conceived by their programmers. They do not result from any kind of "inner reflection" of the device but merely from matching the given representations with (directly or indirectly formulated) *prefabricated interpretation rules and contexts*. Consequently, this kind of interpretation is an indirect, mediated, delegated one and is to be carefully distinguished from direct, immediate interpretation done by human interpreters.

Furthermore, the notion of sign relationship can well be used for explaining the difference between *data* and *information*. In contrast to what we can often read in the gazettes and in many unreflecting ad's, we do *not* treat *information* and *data* as *synonyms* but we want to emphasise the difference between the two: Following their Latin origin, *data* are "given things", i.e. representations which have to be interpreted to convey *information*, i.e. raise conceptions of the interpreter associated with domains. Both terms, data and information, are aspects of linguistic communication which can be illustrated using the semiotic tetrahedron: By using "data" we concentrate on the representation (or sign token) corner whereas "information" puts the emphasis on the conception corner.

What is the nature of the *domain* (or sign object) corner of the semiotic tetrahedron? Following the constructivist's view, it is the result of the joint construction process of a group of observers. If such a joint construction succeeds, i.e. an agreement has been achieved, then such a result may be called a domain comprising "physical" or "imagined" things - depending on whether it is conceived to be localised in the "concrete world" or not. For example, the string 'a-p-p-l-e' may be interpreted as a specialisation of the general concept "fruit" with spherical form and associated to a physical thing localised in our neighbour's garden. But it might also be interpreted as a computer firm or some other trade mark. The correct interpretation depends on the interpreter and his understanding and integration into his own context.

Furthermore we have to be aware that representations are not only used to refer to "*physical*" things but also to refer to elements of other linguistic categories such as activities, processes, events, properties, exclamations, conjunctions, etc. Particularly the *verbal category* deserves attention. By naming (and thus representing) an activity or event, we have data - and can also

derive information - referring to a "*non-physical*" thing. This leads to my objection of what I called myth 1, and my

Conclusion 1:

By using the term *data* we emphasise the syntactic, by using *information* the semantic aspects of sign relationships. *Conceptions of domains* are not restricted to "physical" things, but may also be "non-physical" ones such as activities, events, processes or state transitions.

Ad 2: Nominalisation, Objectification, and the Dynamic Nature of Information

Let me start this section with a quotation taken from an elevator advice at a German University: "Bei Ertönen der Aufzugsnotrufglocke sind zur Durchführung der Personenbefreiung zu benachrichtigen: ... (NN)." This is difficult to translate literally, it means something like: "If the emergency bell rings you should call ... (NN) in order to save the enclosed persons." Let us do a short statistics on the German sentence: There are 1 verb, 4 nouns (consisting of 8 nominal components) and 6 other words, mainly articles, adverbs and prepositions.

Being aware that this is an exceptional example of bad German style, I want nevertheless to formulate the following thesis:

We live in the age of *nominalisation*. Data processing - and particularly data modelling - techniques even enforce this tendency by *objectifying* (i.e. giving static descriptions) of actions and processes. But nevertheless, there are *continuous, dynamic processes* in our conceived world which are not completely grasped by these techniques but only in a reduced, snapshot-like manner.

Let us do a short, very basic exercise in modelling some domain and consider the situation characterised by the following statement:

Client books hotel.

Following the entity/relationship approach, *client* and *hotel* are entity types and *books* is a relationship type (cf. figure 7.h-1).



Figure 7.h-1: A simple entity/relationship situation

One client can book many hotels and the same hotel can be booked by many clients. Thus we have an n-to-m-relationship which is normally resolved by introducing a new entity type *booking*.

What has happened? The original activity *to book* has first become a relationship and then an entity *booking* which can be *made*, *listed in a table*, *modified*, *cancelled* and which even can get *own attributes* such as *status* or *due-date*. As an entity, it is a *thing* but obviously a non-physical one. The referent of the sign token "booking" may be an event (possibly raising actions in a travel agency and elsewhere), a business process consisting of a bunch of particular actions or a state transition from "unbooked" to "booked" possibly recorded in a database.

I will call this process of transforming verbal elements to entities *objectifying* them. Our every-day life - and consequently, our information systems - are full of such objectifications: *Orders* of articles or services, *loaning* of books from a library, *reservation* of flights, train seats or concert tickets, enrolment of students to courses etc.

There is an interesting parallel tendency towards nominalisation in the social or political area. When we look to the language preferred by bureaucrats and politicians (but also used by other social groups including engineers, scientists, technical writers and also many journalists), we find a similar tendency for nominalisation - and a special fancy for it in German-spoken countries: *Entsorgung* (to get rid of waste), *Flurbereinigung* (to "clean" a certain piece of land from plants, creeks and other natural "irregularities" in order to use it more effectively), *Abwicklung* (to dissolve East German institutions and remove their employees from their jobs) and *Erkennungsdienstliche Behandlung* (to check the identity of a somewhat suspicious person) are just a few examples of modern German bureaucrats language ("Bürokraten-deutsch") which might well be complemented by Third Reich propaganda language samples but also by many examples taken from other living languages.

What do the two phenomena have in common? For me, they both reflect a progressing reductionist view which might strengthen tendencies of coldness and "petrification" in society: Nominalisation in bureaucrats language is a means of impersonal speaking and thus of communicating in an indifferent, heartless manner to other individuals without directly addressing them. It is used to "materialise" activities (for example, of observing, oppressing or even physically assaulting other citizens) and make such activities less assailable for potential critics. This way, language is used as an instrument of executing and maintaining power.

On the other side, automation of human activities and business processes makes them "material" in the form of data base entries, automatic signals and triggers etc. They have then to be performed in a certain order, in a determined manner, to follow pre-fixed, stated rules - which again shows a tendency for "petrifying" social activities. One might even suspect that computer technology is one of the triggering factors for nominalisation in the social world. Of course, this is not completely true, as computer technology is much younger than the nominalisation tendencies in society. But anyway, we can clearly observe how computerisation of every-day life sectors helps strengthening these tendencies.

By means of objectification, information systems map or simulate processes by taking snapshots of their beginning, finishing or intermediate states in a movie-like fashion. As a result, computer systems can keep track of what we do, how we act, what we process and what happens to us.

Chapter 3 of this report reflects a position common in the database and information system communities: An action is a transition (definition [E14]), a process is a composition of state transitions [E8], a transition is a binary relationship between two things, the pre-state and post-state [E7] and a relationship is a thing composed of other predicated things [E3]. This way, actions and processes are *reduced* to compositions (mostly sequences) of things (states) which represent *snapshots* of their dynamic behaviour. I do not criticise this approach (since I do not know a better one) but I want to emphasise the difference between *dynamic phenomena* of our domain (a part of the real world we can perceive and conceive) and their *static descriptions* in computer systems and databases. As information systems are more than just computer systems, business processes are much more than (and *should not be confused* with) keeping track or building records of actions and events. Computer systems can *simulate* or *support* business processes but they *cannot completely replace* them.

In this sense I agree with Nissen who advocates for using verbs instead of nouns wherever this is possible. I agree that *knowledge* is not a phlogiston-like substance filling our brain (or even that of computers) but the result of a very long and sophisticated process of "learning to know" about the world. Similarly *information* is not just formalised knowledge or pieces of interpreted data but primarily involves the *process of informing oneself or each other* - a fact which can, by the way, already be derived from the Latin origin "informare". Similarly, *communication* is more than just transfer of messages and more than what can be stored in snapshot-like communication protocols.

Conclusion 2:

Dynamic "things" must not be confused with static descriptions of their results or of intermediate and resulting states. Information is rather a *dynamic process* than a *static thing* or collection of things. It cannot be produced, consumed, stored, guarded, sold or prevented from unauthorised multiplication like a product of matter. When we "sell" or "buy" information, we sell or buy the *service of informing somebody or being informed* by him or her. This is even true for information delivered by a machine or database. It is information gathered and offered *by somebody* and we pay for *his or her service* delegated to the machine.

Ad 3: A Holistic View of Information Systems and Their Impact on Society

The FRISCO report defines *information system* as a "sub-system of an organisational system" (chapter 3, definition [E40]) and in turn, *organisational system* as "a system ... comprising the conception of how an organisation is composed ... and how it operates. ..." [E39]. The information system comprises all its "communication- and information-oriented aspects". Thus an information system is primarily a system of conceptions, and it is well distinguished from its computerised part (the computerised information sub-system, cf. definition [E41]) - if this does exist at all.

Thus an information system is not just a technical artefact but a conception referring to a collection of interdependent components such as human beings (including their actions and behaviour), organisational structures, work procedures, communication lines, computers, databases, and so on. Being an organisational system and an organisation being a social system, an information system cannot completely be formalised. We all know that, for example, the flow of communication between two human beings does not only depend on the communication line or technical interface they share but also on the degree of sympathy and respect they pay to each other. Any attempt to formalise (and thus determine) this part of their communication has to fail since it would affect the fundamental human attitude of behaving autonomously and being constrained only by personal responsibilities.

A *complete* analysis and understanding of an information system has to cover both its formalisable and non-formalisable parts. Installation of a new local network or a new database may partially improve an existing information system but it might be less efficient than changing the duties, responsibilities or hierarchy relations of some of the people involved. The FRISCO conceptual framework ranging from the physical level through the linguistic levels up to the social level offers the necessary space to deal with all relevant aspects of information systems.

Such a holistic view is also necessary when we try to better understand another modern buzzword - the *information society*. Fancy advertisements and politician's speeches aim to sell us communication devices and network services to get members of this glorified community.

They promise us to keep us better informed, ease our communication, shortly: to improve our daily life - cf. myth 5 I started with.

Why do I consider these promises a myth? They reduce again questions of human communication and co-existence to technical ones. They try to persuade us that the broader our ISDN connection is and the more Internet pages we can (theoretically) access the better is our position in the future information society. But they (deliberately or not) forget or try to make us forget that participating in a society is not only communicating with distant people by means of technical devices but that it is neighbourhood, personal closeness, accidental crossing of ways, human warmth or coldness, love, hatred and humour - i.e. a lot of things and phenomena which can neither be controlled nor commercialised.

This leads to my (last)

Conclusion 3:

The quality of life in an "information society" will not depend on the quality of its technical networking facilities but on its ability to use and adapt these facilities to the real social needs. Not the *amount* of information available by such networks will be decisive but its *quality*, its *relevance* to the urgent questions of human beings and its *usefulness* for solving inherent problems of individuals, organisations and human societies.

On Collaborative Linking Support for the FRISCO Concept System

by Pentti Kerola

The purpose of this reflection is to analyse the most essential conceptual interrelationships and their potential descriptions from the perspective of different utilisers of the FRISCO concept system. An open flexible and collaborative linking support, based on hypermedia functionality and experiential learning is proposed for future use and development.

Main Observations and Problems Included

The problem domain of this reflection is focused on the relationships of different FRISCO concepts with their environmental concepts and their different descriptive presentations.

In figure 7.k-1 we have the graphical overview of the major interrelationships between the sub-systems (sets) of FRISCO concepts and their conceptual environment.

The well-based choice and conceptual definition of the philosophical constructivist's position is the most fundamental 'origin' in this conceptual net, but it naturally raises the problem of the selection of another philosophical position, in the case somebody else with a different philosophical viewpoint would like to utilise the results of FRISCO effort.

The conceptual meta-framework for the FRISCO concept system is informally set up in sections 1.3 - 1.4. Actually there exists "open-loop reasoning" in 1.4 (a - d), where the selection of key concepts of information systems: organisation, information, communication, system, conception, model and language are 'inputs to borrow' some concepts from the more general theories and concept sets without clarifying why those concepts are significant and necessary from the constructivist's position on information systems and how they are related to the meta-framework.

The proper FRISCO concept system (chapters 2 - 4) has an interesting, but a little bit fuzzy, starting point. Fuzziness concerns reasoning from 1.3 into 1.5 and 2.1. E.g., why does the constructivist's philosophical position include 'value addition' as one of the most fundamental concepts? In section 2 the authors give a value- and organisation-theoretical reasoning "why we need such a framework in the first place, and what the objectives of such a framework should be" as expressed in 1.5. At the same time the small core set of concepts is also specified for information system science. However, there is little discussion who would use that core and for which purpose. As a reflection I could propose two feasible possibilities. The first one is that this core set is meant for all human actors as a common subset. The other alternative is that this core set of concepts would especially be relevant for managerial actors!

Chapters 3 and 4 form the most essential conceptual content of FRISCO. The main difference lies in the descriptive languages used. The integrated overview is a tutorial mixture of clarifying pictures and semi-formal textual definitions with specified assumptions. The purpose of the authors is "to arrive at a pragmatic minimal set of interrelated concepts" (in the introduction of chapter 3). From my perspective the descriptive problem is, how explicitly the interrelationships between the concepts and their different descriptions are presented. In the traditional book form it is natural, as now is the situation, that only the most immediate precedent relationships are explicit in textual and graphical presentations of the concepts. One essential shortage of chapter 3 is the lack of discussion of who are the users and how this

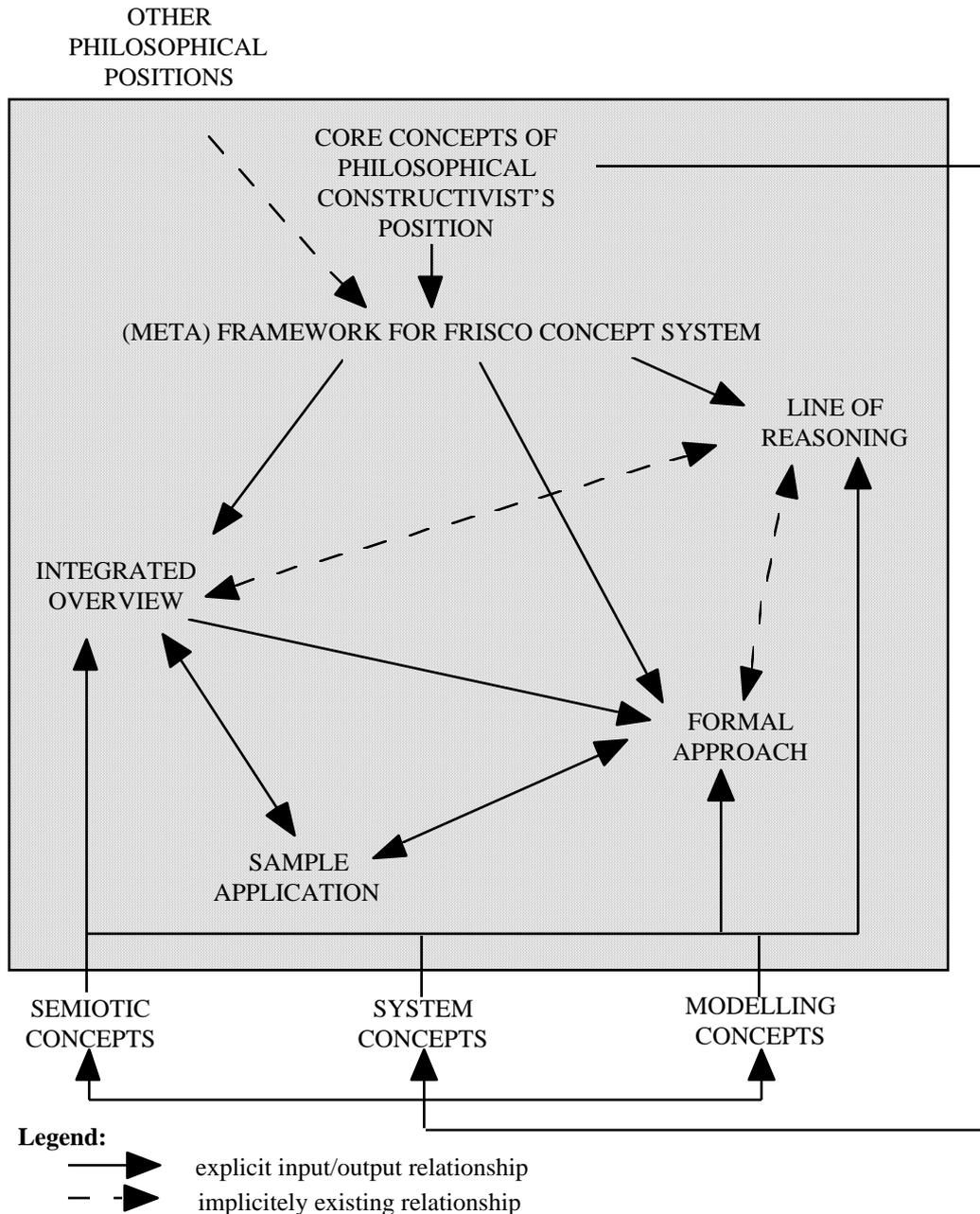


Figure 7.k-1: Major interrelationships between the FRISCO concept sub-systems and their environment

integrated concept set would be utilised. The purpose has been to develop the tutorial constructivist conceptual vocabulary of the professional information system language, but who and what kind of actors are participating in the learning process?

Readers could be puzzled that, from the perspective of this reflection, I have included in figure 7.k-1 'SAMPLE APPLICATION' to the FRISCO concept system, which could confuse readers. However, the sample application in chapter 5 can be seen as the most concrete description of the FRISCO concepts. The authors of the report are giving a recommendation for reading chapters 3 and/or 4 first before the application in chapter 5. Deductively this is natural, but the order could be reversed, too. The inductive approach is also preferable and typical for human actors in information system practice!

As a summary of these reflective observations I can conclude that there exists the diversified need for making explicit relationships, i.e. links between different FRISCO and environmental

concepts and their multiple descriptions for the different type of actors and their different purposes. In the following potential and relevant ideas of linking support are proposed for, at least partially, solving the problems and needs described above.

Linking Ability for Collaborative Hypermedia Functionality

The underlying question is how to develop open distributed and flexible software support for the use of FRISCO concepts. This solution is generally approached by the concept of collaborative hypermedia functionality implemented in the multi-tool, multi-user environment. Hypermedia functionality is seen as a value-added support functionality for software systems, allowing the information to be investigated in a non-linear, semi-structured way. Core hypermedia functionality consists of the creation, modification and deletion capabilities of various kinds of hyperlinks and nodes, and orientation and navigation capabilities achieved via the hyperlinks. The purpose is to provide its users with an associative way of organising, analysing and accessing information. We are especially concerned with collaborative hypermedia functionality i.e. systems that emphasise the groupware or team aspects of knowledge work. Even so, team support always takes place through supporting individual team members.

Collaborative support of this kind has been implemented in MetaEdit+, which is a multi-user, multi-method and -tool metaCASE environment [KLR96] (figure 7.k-2). MetaEdit+ consists basically of a set of tools for modelling and description e.g. Diagram, Table and Matrix Editors, a toolset for justifying and inspecting the decisions made, known as Debate Browser and a hypermedia functionality underlying the construction and argumentation tools, known as Linking Ability.

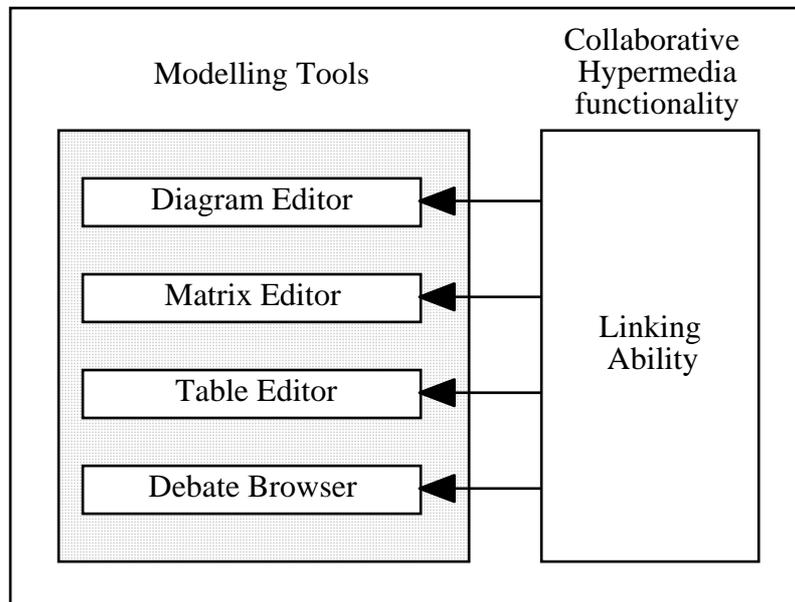


Figure 7.k-2: MetaEdit+ software architecture

Diagram Editor in MetaEdit+ simply enables the drawing of diagrams using a specified method, which can be determined by the user. The conversation on development issues is captured in the Debate Browser with the fixed design rationale method. The method consists of questions (q), answers (a), and arguments (r), which either support or contest the answers. The Debate Browser actually provides its users with three browsers: a graph browser, a

document browser and a node browser. The graph browser represents the overall structure of the design rationale in graphical form, while the document browser represents a focused view of the structure in textual form and the node browser enables exploration of individual nodes.

Collaborative aspects of these tools are received through the Linking Ability sub-system. The Diagram Editor (as well as other editors) enables the linking of any object to any other through hyperlinks. All hyperlinks may have attributes associated with them in order to provide more information. These associative hyperlinks integrate also the arguments, by means of which any argument node can be reached from the objects in a diagram. It also allows the user to ensure that all outgoing/incoming links from/to a diagram have been checked.

The interaction history list supports backward navigation by moving to a node that has already been visited. Two kinds of marks are also provided as guides for readers. A bookmark is a marker for keeping one's place in the collection of documents, and can be attached to a specified object in a document (if it might be of interest later) while a landmark directs the reader to an important region in the document. Their semantic difference is that, while a bookmark is meant to be personal, a landmark is meant to be informative for all readers.

In this case of reflection the issue is the FRISCO concept system, its use and further development, especially concerning the interrelationships between the FRISCO concepts and their conceptual environment. We have started to develop the analogue collaborative software support concerning the most essential concepts, methods and tools of quality based software design and inspection [TKO97].

Towards Larger and More Sophisticated Collaboration

The solution approach above necessarily needs the specified type of support environment: MetaEdit+ or corresponding. It is suitable for small team collaboration having compatible support resources. The proposal for larger and more open user participation would be to develop a WWW-application for the FRISCO concept system. Regrettably the existing descriptive tools are rather limited and many ideas described earlier cannot be implemented in this developmental phase of WWW.

A totally different kind and more sophisticated approach could be developed on the basis of Kolb's experiential learning theory [Kol84 and TKO97]. There the fundamental idea lies in the human information processing (including conceptual development) and in its multi-dialectic nature. Dialectic here means structure and interaction between two poles of dimension which are both opposed and complementary at the same time. In human information processing there are two fundamental dimensions: concrete/abstract dialectic and active/reflective dialectic. Experiential learning is modelled as holistic human adaptation to the world, resulting with four basic types of knowledge: accommodative (concrete and active), divergent (concrete and reflective), assimilative (abstract and reflective) and convergent (abstract and active) knowledge. If those basic types of knowledge are applied on the FRISCO concept system we could preliminary characterise chapter 2 as divergent, chapter 3 as convergent, chapter 4 as assimilative and chapter 5 as accommodative.

Furthermore, the Kolbian knowledge-creation theory argues that over time, individuals develop their typical possibility-processing structures in such a way that the dialectic tensions in and between the dimensions are consistently resolved in a characteristic fashion. People develop experiential styles that emphasise some abilities and characteristics over others. Using his knowledge typology, Kolb correspondingly defines four basic categories of experiential styles (ideal types in the Weberian sense): Accommodator, Diverger, Assimilator

and Converger. Explicitly the styles in reality are normally different kinds of dynamic combinations of those basic styles.

The final idea of this proposal is the concept of Kolbian collaborative hypermedia software. In principle, it includes about every conceptual object of human interest four interlinked nodes, each representing one Kolbian type of knowledge. This type of hypermedia software should be developed collaboratively. The original developers have their existing dominant experiential styles and preferences. Therefore during the early phases of development and use the Kolbian collaborative hypermedia software is "biased and skew". In the long run the collaborative reading and authoring would produce the more balanced output where all the knowledge types have been well covered, especially concerning interrelationships between the concepts and their different descriptions.

As a summary, the FRISCO members have made a significant integrative conceptual contribution for the information society giving a promising starting point for the larger collaboration in the scientific and educational efforts of informatics.

Acknowledgements

My friendly thanks to Harri Oinas-Kukkonen and Jouni Similä about the insightful discussions concerning this topic.

FRISCO - A Way to Insight?

Reflections on a Challenging and Eye-opening Project that Missed a Goal

by Paul Lindgreen

If you don't know where you are, a map won't help.

Anonymous

If you don't know where you are going, any way will do.

Chinese proverb

In May 1987 during the IFIP WG 8.1 business meeting in Sophia Antipolis one agenda point was: "Profile of future WG 8.1 activities". There I presented a memo proposing a possible 8.1- task aimed at - I quote:

- "an improved paradigm for our professional field."
- "a firm conceptual foundation for [...] systems, communication, the modelling of UoDs, data processing [etc.]"
- "proper arguments and recommendations for a reasonable, professional terminology."

The proposal was well accepted, and Eckhard Falkenberg and I were given the job of working out a document [IFIP 88] specifying aim and scope of a new WG 8.1 task - the one that early in 1988 officially became FRISCO.

Although it was a WG 8.1 task, we thought that the group should aim at a broader perspective and also comprise more far-viewing members who could see the need to take into consideration some of the less formal organisational / social views typically represented in WG 8.2. I don't remember my arguments at that time. Probably they were only intuitive. Now I am fully conscious of why that decision was important and a right one we took. All the arguments for why it is necessary to widen the scope beyond that of WG 8.1, I see nicely presented in chapter 2.

I shall not enter a detailed subjective description of the history of FRISCO, but a few comments are necessary for the reader to understand the opinions I express afterwards: From the beginning in 1988 and up to early summer 1994, I engaged myself in a lot of exciting FRISCO-work: as secretary of the project, as editor of the interim report [Lin90a], as author of 25 FRISCO- working papers and more than a dozen internal memos etc., and I took part in 19 of the (at that time) 20 plenary meetings and was active in numerous sub-group meetings on specific issues as well. However, during the following fall and winter, it became clear for myself and for everybody else that I viewed a number of issues quite differently than the majority of the group. The disagreements were primarily on the intended structure and means of chapter 3, in particular on which concepts to take into consideration and on what was needed in order to explain them properly. For me it was a great disappointment that the group could not support my suggestions. On the background of my former enthusiasm I was rather sad at that moment, and I seriously considered leaving FRISCO. I could see that the outcome would be far away from my own intentions. Anyway, I decided to stay and take my part of responsibility for our common child, although my eagerness in the work dropped to a rather low level, as I realised what was emerging.

Now, when the task is finished, two questions cry for an answer:

- 1.) Did FRISCO turn out to be a good idea, and was it worth all the effort?

2.) Is the result satisfactory?

I will answer the first question strictly from a personal point of view and the second one as a neutral observer - a role in which I as an "insider", of course, must fail.

1. Yes, it was a Very Good idea, and it was Fully Worth All the Effort!

For me the project turned out to be intellectually challenging and highly inspiring. FRISCO became the frame inside which I was privileged to carry out a lot of interesting, eye-opening activities together with experienced colleagues from whom I learned so much.

It was challenging and inspiring, because the goal which I saw from the beginning: *To construct and support by proper arguments a professional language for our area that is based on deep conceptual insight and a broad perspective of generality and relevance* - that goal became justified as a realistic one, although one that still requires a considerable effort to be reached. During the project I have frequently experienced smaller and larger steps in what I see as the right direction - steps that inspired me to carry on and learn more. At present, I am working alone on an alternative approach trying to come much closer to the goal than was possible in FRISCO. However, I realise that I can only do so by building to a large extent on the insight I gained from work with FRISCO colleagues.

The FRISCO activities were interesting and eye-opening for me, because I learned about scientific areas which, so far, I had known only by name. FRISCO catalysed for me a learning process that resulted in insight with a number of important concepts and principles, in particular from semiotics and sociology. In this learning process I found definitive reasons for - what I had suspected for a long time (but not fully realised why) - that the present overwhelming focus on computers (and also too narrowly on "information system"-oriented concepts) prevents us from gaining the more general insight that is necessary really to utilise the potential of modern data technology. Now I am convinced that what really matters is more insight into the concept *co-action* in general and concepts and principles of *communication* in particular, and into the *linguistic and social conventions* that are behind co-action and communication in organisations.

From the many discussions we had in FRISCO, I realised one thing of utmost importance for scientific communication: *It is absolutely essential always to be as explicit as possible about what you mean*. Persons who have worked together for a long time can rely on a lot of tacit knowledge and intuitive understanding, but when you want to get a scientific message through to other partners in society outside your own group - as is the case with this report - you cannot assume that there is a sufficient common understanding as regard the specific area. It is my experience that the only means ultimately to rely on as a foundation for professional communication is *our common language*. In particular, we have the problem in the area of "informatics" including what is called "information systems". In this area there is no common, inter-subjectively shared sense of any of the professional terms applied in explanations, and there is no agreement on which concepts are relevant and which are not.

2. No, the Result - the FRISCO Report - Unfortunately is Not Satisfactory!

Although I feel that I share the responsibility for our product, I am also partly very disappointed with a considerable part of it. I am afraid that instead of increasing the level of insight in our professional area, it may rather add to the prevailing confusion. When I have expressed this to other group members, some of them have pointed out, that FRISCO, indeed, has achieved a lot, and that my pessimism comes from having not realised that all we now

consider self-evident and well-justified knowledge is not necessarily known by everybody else in our professional community.

I agree that we have learned much, individually as well as collectively, but our insight - even that part which is collectively shared, is only partially expressed in the report and in the crucial part of it where the explanations should be given, not in a way that makes it easy for people to understand our messages. But most of all, I am disappointed that after eight years work the goal I personally for a long time envisioned - *the proposal of a proper professional language for our area* - still, as I see it, is not within reach.

Now, in order to be loyal to the lesson: "Always be explicit about what you mean", I must shortly describe my sense of 'proper professional language'. First, I consider a language not to be what the report laconically claims: "A non-empty set of permissible arrangements of symbols taken from an alphabet". For me a language is a social convention - an area of relevant concepts and rules that constitute a (semiotic) mapping from instances of these concepts to a potential domain of perceivable and sensible utterances about them. This means that in order to have any importance in practice, the language users (here those active in a certain profession) must *understand* the concepts behind the possible utterances. Then 'proper' in this context means that the concepts should cover - i.e. enable the description of - all relevant professional phenomena on the intended level of generality. Furthermore that applied terms, phrases, symbols or whatever should be *well in accordance* with common language sense and grammar and with other general conventions of the society. The reason for the latter should be obvious: Learning a professional language in practice requires that one can talk about it in some other language, and the only one that *everybody* knows is common language.

The most important issue in this interpretation of language is the establishment of the *inter-subjective understanding of the relevant concepts* that is a necessary prerequisite for practical use of the language. The question one must raise, then is: What does it mean to understand something, and how can descriptions best support the understanding? Again here, there is only limited consensus about the sense of the words. However, as I see it, it is necessary to distinguish between knowing that one thing is different from another thing and really understanding the involved things. Accordingly it is necessary to realise the difference between *definitions* and *explanations*. Both are descriptions, but they serve quite different purposes: The potential of a definition is that it, in a concise phrasing, can put focus on something that is relevant to know of, that it declares a name for it, and that it can state enough about the thing to enable people to distinguish it (or instances of it) from other things. An explanation of a thing, on the other hand, requires much more, because the purpose of it is to provide enough knowledge of *every relevant aspect* of the thing, such that it is possible really to understand it and thereby apply it in a sensible and useful way.

This is where I am disappointed with the result of our work - essentially with chapter 3 and 4, since they are declared as "the core of the report". It is exactly in that core, the report in my opinion fails by not properly enabling the communication of our insight to people outside the project. Chapter 1 gives a fair outline of the situation behind the FRISCO work and some crucial aspects of the task we have dealt with. Chapter 2 in an excellent way provides a primary argumentation for why it is relevant to include and seriously consider concepts like those presented in the report. However in the chapter 3/4 core I suspect many readers after having sensed from chapter 2 the importance of our messages will be left in serious troubles trying to understand what it really is we suggest as relevant professional concepts. Primarily they will have difficulties in anchoring them with concepts already known from common language and to see how many of them are just professional specialisations of common

phenomena that are relevant when things are regarded in a broader perspective. In some cases it will also be difficult to see a justification for the chosen terminology. In my opinion it is not enough just to list a number of possible/traditional synonyms.

I believe that FRISCO failed twice in the process that lead to the present core of the report: The first major error occurred when it was decided that a *formalised set of axiomatic definitions* should be a main goal. The "rationale" - which I opposed - was the opinion that without such a formalism the report would not be regarded as a proper scientific document measured by the standards usually expected from work carried out under IFIP.

In every respect I (like probably everybody else in FRISCO) support a scientific approach within IFIP. However, my argument against the decision is that providing a set of axiomatic definitions neither is a necessary nor a sufficient condition to ensure a scientific result. This, in fact is valid for any kind of formalism. To express something in a formal way is a powerful tool to clarify certain aspects and to expose possible omissions and inconsistencies. But any formal tool is limited in semantic power (it is the price paid for precision), and applying it to enforce the qualities requires that there is something to make complete and consistent.

As regard the chosen axiomatic approach I argued against it for four reasons:

- It imposes a hierarchical structure on our set of concepts regardless of the fact that the concepts - like all other things in the world - generally are related as a complex network.
- There are many aspects relevant for understanding our concepts (which therefore must be described as a necessary part of the explanations) that cannot be expressed properly by the formalism (or at least only in a very awkward manner).
- To be manageable it requires a very limited set of "given" or "assumed" basic / axiomatic concepts, which is in contradiction with a broad anchoring of the majority of our concepts with commonly known concepts.
- Although based on rather simple mathematical concepts many readers in practice are untrained in the notation - also among those scientifically educated - and accordingly they will have severe difficulties in grasping what is expressed by the formal statements.

Acknowledging the usefulness of formalisms in the realising process I suggested *to describe relevant aspects of our concepts in an information model* with an entity type for each concept and with a description of all attributes and all roles with cardinalities which the entities are playing in respect to each others. This could be illustrated in a proper set of ER-diagrams and further documented in a detailed lexical specification of all relevant properties.

Anticipating the latter of the four above mentioned problems FRISCO decided to incorporate a *tutorial chapter 3* before the more difficult chapter 4 with the axiomatic definitions.

Now, a tutorial can be a good idea, when a message is comprehensive and complicated and/or if it requires a new way of thinking. But its use is limited, if there isn't a reference document containing the full story - all the difficult stuff a reader assumedly must be tutored about to understand. The problem with the chosen approach, however, is that chapter 4 is no such a reference document. It contains definitions, but primarily because of the minimal set of concepts to anchor an understanding to, it does not provide sufficient explanations. Then, in my opinion FRISCO made its second communicative error: It was declared that chapter 3 and 4 should have the same overall structure, and that the understanding of chapter 3 also mainly should be based of the minimal set of concepts from chapter 4. A tempting - although not completely fair - way to express the outcome would be to say that chapter 3 is a version of chapter 4 flavoured by a considerable amount of "syntactical sugar" and allowing a few of the referential loops that are necessary for the understanding, but prohibited in chapter 4.

I argued for the alternative approach, that chapter 3 should provide the *full explanations* and be structured in a way that best *supported a gradual understanding* of our concepts and anchoring to already more general concepts. An indication of the explanatory "style" I intended with such an alternative chapter 3 and of the amount of details necessary to incorporate can be found in the three explanatory sections of chapter 6. Properly formulated in such a way chapter 4, then, could be regarded as providing a supporting formal "proof" of the "correctness" of a part of what was stated in chapter 3 and an alternative, more precise specification of this part. Furthermore, besides the explanations chapter 3 should provide *arguments for the direct or indirect relevance of the concepts* in the organisational and communicative context outlined in chapter 2. Finally for each concept there should be *proper arguments for the suggested terminology*, and as the understanding of the concepts could be expected to grow with the reader, examples should be given of applying the terms in meaningful phrases and rules for proper concatenation of the terms to denote more specialised concepts.

Well, I did not succeed in convincing my colleagues of the need for a different approach with the core of the report. (Or my ambition were too high given the general conditions under which our work was carried out.) Anyway, although I don't like it, I must repudiate chapter 3 and 4 of the report. They do not live up to my own professional intentions with written communication. I don't like it, because there are so many important messages to our professional community from the FRISCO work - also in chapter 3 and 4. Therefore I urge everybody who have reached so far in reading the report, to try to grasp and judge the details of our messages and even more important - but also more demanding as regard the readers patience - *to try to derive the systemic properties of the system of concepts* which, in fact, is the main result of the FRISCO work.

Reflections from a Practitioner on the Information Concept

by Björn E. Nilsson

a) THE TOPIC

The most intriguing concept of the report is labelled *information*. Information is certainly a term with a lot of different interpretations in different subcultures of our profession as well as in ordinary common language. In the report, the term is used in accordance with the Scandinavian tradition which, in turn, is founded on the definition Børje Langefors provided in the early sixties.

My reflections will only cover two limited aspects of the definition of the concept. I will, contrary to my normal inclinations, neither dwell on the conceptual framework of semiotics nor dwell on old questions of philosophy but, rather, first concentrate on how well FRISCO filled its task as seen from its conceptual criterion of relevance and, secondly, to give a short historical commentary to the use of the term information as introduced by Langefors.

b) THE FRISCO DEFINITION OF INFORMATION AND THE BASIC CRITERION OF RELEVANCE

b1) The Criterion of Relevance

The FRISCO group formulated a fundamental criterion of relevance by which to accept or reject concepts. The criterion is based on the idea that the usage of a concept in analysis and design should ultimately contribute to value creation in the target business processes. In my view, the current definition of information does not fully take advantage of the power of this criterion of relevance to construct the most pregnant definition.

b2) The Relevance Trace Behind the Current Definition

The criterion of relevance is based on the notion of value addition - which is assumed to be achieved by conscious action. Coherent action in large scale undertakings is enabled by communication - i.e. the basic function of communication is to get things done. Communication, finally, is accomplished by the exchange of messages - where the information in a message is defined as the new knowledge gained by an interpreter thereof. The gained new knowledge is enabling proper action.

Clearly, this reasoning follows a proper traceable chain which links the concept to value addition in the enterprise - but does the definition encompass the full consequences of properly applying the criterion of relevance? In my view, the answer is no. Let us start with the structure of the mentioned definitions.

b3) The Structure of the Definitions

Communication is a balanced concept involving two actors with alternating roles interacting by messages. The message concept is balanced involving two actors in the different roles of sender and receiver. The concept of information is unbalanced and only involves the receiver. Now, is that sufficient?

Yes. It is, as far as I can see, sufficient from a formal point of view provided knowledge

encompasses all aspects relevant. The question is rather - is it good from an analysis standpoint?

According to my view a balanced definition with defined roles would make for a richer and more useful concept during analysis.

b4) What is Lacking and What is There

To be able to criticise the definition, I will have to utilise the underlying concept of a message rather than the information concept itself. Otherwise the things I want to say become inexpressible.

To start, the most relevant part of a proper definition, is already in place. The concept of new knowledge means that the first link of a proper value chain, the enabling part, i.e. the basis for action, is in place.

The most relevant missing parts of a definition relating to value creation are:

- A** The intention, the very reason for producing the message by the sender
- B** The inter-subjective agreement on meaning, the generally accepted meaning of a message
- C** The effect of the message in terms of action or effect

Moreover an important inexpressible parts of the definition is

- D** The truth value of a message

Let us start with the missing parts.

b5) Information Intent - A: The Intentional Aspect

Following the way of the relevance criterion, the most important aspect of information is the enabling of value creating action. The essence is that action in a business environment is enabled not only by making state of affairs, in the classical meaning, known, but, rather, by making intentions known. By explicitly introducing the intention part in a definition, utterances like - I want you to - you did not understand me - that was not what I meant - by this I mean - become a natural part of the discourse. Now, would this introduce problems?

Certainly. The most serious problem is that intentions are not observable, they are internal. On the other hand, so is knowledge. This means that the difficulty is not of a new kind. Even if the observability is not direct, indirect observation or interaction is possible.

b6) Information Content - B: The Inter-subjective Aspect

In the context of co-ordinated action over time, similar interpretation over a defined population has to be established. So has a certain time permanence of interpretation. The most important feature of information-technology-based communication, as via databases, is to ascertain that messages are produced and interpreted using roughly the same grammar. Grammar is here used in the wide sense and taking both syntactic and semantic aspects into account. When a certain message is produced and introduced to the information support system we do not know by whom and when the message is to be used. This means that we have to establish control over the interpretation process. Normally this is done with the help of a conceptual schema in a database environment, with form types in a form based system and so on. This is the meta-aspect of data.

b7) Information Effect - C: The Effectiveness Aspect

This is really very closely coupled to the intentional aspects. However, intentions are not always honoured, even if the "correct" interpretation is at hand. If we introduce effect, the simple discourse would be able to handle assertions like - in harmony with paragraph 2 of this contract I decided to - as the intention behind the order was unclear, I made the following assumptions - under standing orders I opened fire when - and so on.

Another aspect which links to effect is time. A good illustration is to tell a good story with the components in the wrong order. This problem, however, is taken care of by the communication concept.

The effect reasoning, by the way, leads to the conclusion that an empty message - still listening by being silent - is also information carrying. This is merely to give a flavour of the kind of second order reasoning that has to be applied when changing in a set of concepts.

b8) Information Reliability - D: The Truth Aspect

With the approach of the FRISCO group the classical correspondence theory of truth is not applicable. This means in practice that the truth concept is not developed. I will not dive into the swamp of truth concepts but rather give some hints on reasoning.

A constructivist view implies that the normal definition of truth becomes a fuzzy concept based on inter-subjective consensus. In practical analysis, the merits of a constructivism-based reasoning is so strong that this seems to be a very low price to pay. In popular terms: We are not modelling the world - we are not even modelling conceptions of the world - we are modelling conceptions of the interpretation of utterances concerning conceptions of the world. As such utterances are model denotations, we are modelling models.

The consistency notion of truth might be used formally if restrained to the data level rather than the information level. Consistency on the information level has to do with the established knowledge as being a reference base for judging consistency of new knowledge, or, to judge the complex of knowledge as a whole.

The pragmatic notion of truth which might be expressed as: What is judged as useful for the purpose is true. This truth concept has never been seriously considered within the group.

c) A MODEST HISTORICAL OUTLOOK

Langefors defined the concept of information in essence in the same way as FRISCO, i.e. as new knowledge. For this decision he had three major but rather different reasons.

The first reason was to separate the analysis of the business system and its information demands from the analysis of the computerised data system. To make a clear distinction between data, as the carrier of information, and the information as such meant that the analysis could reason about necessary information for proper decision without any regard to how and in what form information was provided. Langefors was not without history, though. In logic, linguistics and philosophy some schools made the same kind of separation between sentence and proposition, term and meaning.

The second reason was to be able to use the backbone of signal theory, which still is called information theory, in the analysis of information and data systems. This motivated, in essence, the introduction of the conception of novelty, new knowledge. To introduce new knowledge meant, moreover, that while data could be redundant, information could not. Still, it was possible to talk about requisite variety on the information level of analysis.

The third reason was that Langefors mainly thought of information as the basis for decision. This meant that novelty is crucial in the definition. To have the same data repeated gave no information - if not the context was altered. To have different data leading to the same interpretation gave no information either.

During my early, and very mild, criticism of the information concept by Langefors, he expressed the thought that actually information was not used primarily for decision making but, rather, to positively reinforce already made decisions or decisions underway. With some insights into my own behaviour when driving a car and reading the map, I fully supported this view. Reinforcement or reassurance seemed to play a very important role, and the observations of the map were frequent and not, in essence, giving new knowledge. Still it affected my well being and my decisions. Based on our talks I came up with a new definition, with which Langefors did never fully agree: The information in a certain message is the effects over time on a receiver from that message [Nil79]. This happened to include most of the properties we wanted from a definition, but it was very hard to keep the signal theory aspects intact. I think we should have changed that theory instead.

The definition had some merits in that time and sequence could be taken into account. From the thoughts on effect, it was also clear that "non-action" must be considered as action. This happened to correspond nicely with the view of existentialist philosophy concerning these matters. Change is a concept with certain connotations. It is easily seen, however, that effect, action and value are not always linked to change at all. The keeping of a certain desirable state which is threatened by change also may create value. An umbrella could as a resource be used to withhold the resource of dry property if properly used - thus creating value.

To sum it up, the thinking of Langefors, concerning the effects on analysis by making clear a fundamental differentiation of data and information, was at least fifteen years ahead of the prevailing praxis. Today, we can slowly see this way of looking at things being accepted world-wide and finding its way into methods of analysis.



A Dissenting Position

by **Ronald K. Stamper**

To the shame of our profession a high proportion of computer systems fail. They fail not for technical reasons – most delivered software performs efficiently to specification – but for organisational reasons – they do not relate correctly to the world of business reality. The sad facts that, in general technical people do not understand business problems and business-oriented people do not understand the need for detailed, formal precision contribute to the problem. We need to link the two sides of the problem together with a scientific framework of information systems concepts that will allow us to develop the subject in a formal and precise way but on a sound experimental basis.

FRISCO begins to achieve the necessary synthesis in chapters 1, 2, 6 and part of 3. But the formality in chapters 3, 4 and 5 abandons the synthesis and builds on a foundation that cannot support an empirical treatment of information systems. Sadly, therefore, I have to express my dissent.

Scientific method and the sociology of science

I find no pleasure in dissenting from the majority of my colleagues over the final FRISCO report, but honesty compels me. Having expressed my position consistently, I cannot withdraw now. Science only progresses through our attempts to refute accepted theories, as Popper has demonstrated, but Kuhn has taught us to expect no quarter if we have the temerity to object to the established paradigm.

Popper [Pop72] characterised the work of a good scientist thus: “[He] is consciously critical of his theories which, for this reason, he tries to formulate sharply rather than vaguely.” (p.25) He demonstrated that as we cannot attain absolute truth, “all theories therefore remain conjectural” (p.80), however “every time we succeed in falsifying a theory ... we make a new important discovery.” (p.196) Thus he says: “The method of science is the method of bold conjectures and ingenious and severe attempts to refute them.” (p.81) He draws the demarcation line between science and non-science, to separate refutable hypotheses from those “immunised” against refutation. (Popper’s emphases throughout.) One might express his position in proverbial form: No criticism or possibility of criticism = no science.

We proceeded this way in our discussions and happily for my colleagues they can broadly agree with the ideas presented in this report. However, for the whole ten years of our collaboration I have held a view quite incompatible with their formal consensus in chapters 3, 4 and 5. My dissent has no personal basis and I presume that, as scientists, they will welcome every sincere attempt to test their theories by critical argument, including mine. Kuhn [Kuh62] pointed out that a single paradigm normally dominates a science and its practitioners will not treat kindly anyone attempting to undermine a consensus in which they have made personal investments of time and reputation. Kuhn cites the historical reality of science to show that we can forget Popper’s ideal of science proceeding by conjecture and refutation. Politics and the exercise of power ensure that the isolated outsider will have a hard time. In my case this consists of having to express my objections to approx. 130 pages of chapters 3, 4 and 5, in the same allocation of six pages as my colleagues who agree with them. I can offer you in the ordained space little more than an invitation to contact me for an adequate account of my position.

Of course, I do not claim to know the truth, I merely advance my criticism and offer another view which I invite others to refute. I have some confidence in this alternative because it

works well in practice, making systems easier and cheaper to develop, support and maintain than orthodox methods.

Assumptions

Remember my serious objections only to the formal part of this report, so let me go straight to my disagreement with the assumptions in section 3.1 which I replace by:

- [a'] No known "world" exists independently of our existence, though we may hypothesise that it does.
- [b'] Individual beings can discover ways of behaving that help them to survive, more or less comfortably, and from these invariant repertoires of behaviour they construct all the things that constitute their separate worlds.
- [c'] By participating in a society, human beings can recognise and adopt the behaviour patterns of others when they seem effective, and so develop socially a far richer world than they can individually; moreover, they learn to use one thing (as a sign) to stand for another (an object) so they can develop repertoires of behaviour beyond the capability of any individual and so enrich their socially constructed reality using language; these behaviour patterns, including their language constitute a culture.
- [d'] We can study information systems without using such mentalistic notions as perceptions, conceptions, senses, mental states, cognitive and intellectual processes; we need only study overt regularities of behaviour, but with special emphasis upon the uses made of some things to signify others.

The alternative assumption [a'] I prefer because we have no need to rely on the immunised hypothesis in [a] that "the 'world' exists, independent of our own existence", which also misleads us into believing that we can deposit in our computer systems some absolute truths or data for which no individual, group or profession need take responsibility. Later I shall explain further the notion in [b'] of behavioural repertoires, or affordances, which enable us to define things in a manner that allows anyone to observe and subject them to empirical testing which one cannot say for "percepts". [c'] makes explicit our almost total dependence on our culture for a rich world view as well as our capacity to use information in more sophisticated ways than animals. [d'] makes sure that we base our theories not upon what we can only imagine takes place within a million skulls but upon empirically observable behaviour including sign-behaviour.

I shall show how these sets of assumptions make the difference between a purely formal, technical, abstract treatment of information systems and one that can also handle the social, pragmatic and semantic aspects of information. First I must demonstrate the deficiencies of assumptions [a, b and c] then demonstrate how [a', b' and c'] can handle the neglected, organisational aspects of information systems.

Constructivism and what we construct with: Definitions and primitive notions

Chapter 3 begins by specifying the FRISCO position as constructivist, one which builds models on the basis that 'we only have access to our own (mental) "*conceptions*"' from which we 'negotiate an agreed "*inter-subjective reality*"'. The term 'constructivist' emphasises the process of construction, disregarding the quality of the building bricks. I admit in [c'] that we construct our 'world' by negotiation but I replace FRISCO's "*conceptions*" with repeatedly observable *repertoires of behaviour*.

We construct definitions. These explain one term using several other, preferably simpler terms. The formal chapter builds a chain of approx. 50 definitions starting with:

'A thing is any part of a conception of a domain (being itself a "part or "aspect" of the "world").'

building on the primitive notions:

conception, world

plus English as a natural, informal meta-language. Construction certainly takes place but notice that all the information concepts defined depend on these basic building blocks which we have to take on trust. We cannot observe the 'conceptions' locked inside our skulls; and a 'world' that exists independently of any of us also defies observation because the observer involved always gets in the way.

Take any FRISCO definition and trace it back to its source and you will find it relies on the notion of conceptions which one may accept in an act of faith but which deny operational treatment and so rule themselves out as the foundation of science. Take, for example, information defined (E36) as an increment of personal *knowledge* defined, in turn (E33), as a stable and consistent set of *conceptions*. What a pity! The informal treatment of our subject in the other sections (especially 1.2c, 3.4, 6.1) explains that information has many different exact operational meanings.

Operation meaning: The ostensive definition of "sign"

We can define words in terms of other words which also define in terms of other, simpler words, and so on. But how can we escape the verbal trap? Easy! Given enough concrete examples, a person can learn the meaning of many terms without recourse to other words except possibly as a commentary to accompany the demonstrations. As children we all learn the meaning of such words as "table", "chair", "cup", "saucer" and "friend". This method of definition by ostention does not work for the inaccessible conception which one can only "see" in a metaphorical sense by introspection. As a basis for a scientific discipline we must have primitive notions that do not rely on such untestable procedures as introspection. Fortunately for the discipline of information systems, the notion of a sign lends itself to definition by ostention; because we can define it by demonstration it will serve us well as a primitive term on the basis of which we can construct a chain of definitions that escapes from the verbal trap.

Just show a person some thing-A standing for some other thing, thing-B, in the sense that someone can interpret A (as a sign) in order to act appropriately towards thing-B for which it stands; for example a red traffic light standing for the dangerous nature of the road ahead, or the list of household items standing for the shopping one must do. I could take you into the town and show you enough examples to make sure that you have grasped the meaning of "sign", and I could easily test that you do understand. At that point where we demonstrably share the same repertoire of behaviour when interpreting the word "sign", you can begin to use that word to get useful things done, for example, in the field of information systems. We don't need to use the vague term "information", we just need to get on with the job of engineering ways to do things using signs.

Just as we have rooted physics in the simple, demonstrable primitive notion of a physical object so we can root our study of information in the notion of a sign. We then find that signs have many precisely definable properties some of which correspond to various aspects of our common sense idea of information. Just as physics studies separately the clusters of static, dynamic, hydrodynamic, elastic, magnetic, electrical, optical and other properties of objects, so we can remove confusion from and simplify the study of signs (one meaning of information) by dealing separately with each of the six levels of the Semiotic Framework [Sta71, Sta96]. This report does so in 6.1 (for more also see [HS70, Nau72, Sta73] and their citations) demonstrating many precise, operational definitions of "information" in terms of various properties of signs. Thus, in the alternative theory of information systems which I propose elsewhere, "information" has at least a dozen precise operational meanings.

Information systems as computer systems or social systems

Isolated individuals do not develop language of any kind. Language depends upon the members of a community acquiring the same set of behavioural repertoires each linking a thing-A to a thing-B, so that, for example, thing-A (the ink patterns you now see in quotation marks "bank") means thing-B (the upward sloping land you see out of your window). We can also link signs to signs in this way, for example: "bank" means "declivity". If we confine ourselves to meanings of this second and simpler kind, we can occupy ourselves totally with the world of signs. Computer science does that; but crucially information systems must also concern meanings of the first kind. Computer science would lose nothing by relating its signs to conceptions or any other imaginary primitive notion, but information systems must relate signs to physical and social reality and someone must take responsibility for each interpretation. Therefore the FRISCO definitions suffice for theories of computer science but I would not like to trust an information system based on such a theory.

From a formal perspective, a well constructed game of Dungeons and Dragons does not differ from a system to control a nuclear reactor or run a social security service, because software structures have nothing to do with the difference between fiction and fact. Formal systems of symbols need not have any meaning outside themselves. In which case the ultimate building blocks of the tower of definitions does not matter, so I do not object to using FRISCO's formal framework for formal purposes. Well-written software for Dungeons and Dragons should meet the same formal criteria as programs for controlling a nuclear reactor or running a social security service. Certainly we should demand internal consistency of any model for an application system and all of those software systems should meet that criterion. As my home in England lies a few miles down the prevailing wind from a nuclear power station, the mathematically proven correctness of its control programs provides my family scant comfort if our profession uses little more than intuition for linking a computer's statements to reality. FRISCO's definitions lack the essential criteria for relating signs to reality. Consequently, they treat the subject of information systems as a minor branch of mathematics concerned with the internal consistency of its models. But to test the operational meaning of a program to control a nuclear power station requires us to construct it on the basis of notions with operational meanings, not conceptions to which we have no empirical access, and must take as a matter of faith. As Popper pointed out ([Pop72], p.122): "Scientists try to eliminate their false theories, they try to let them die in their stead. The believer – whether animal or man – perishes with his false beliefs." When I read that, the nuclear power station looms before me. For the software controlling it I should like to know that all the code has correct meanings (in terms of the reactor, not the in the formal, set theoretical sense). I should also like to know on whom to pin responsibility for its program statements, parameters and data and moreover that I trust their sense of value.

Meaning, intention and social value

We have ignored difficult problems: those of semantics, or how to justify the implied claim that our data relate to reality; of pragmatics, or how to justify acting on the statements our systems generate; or of the social consequences that give information its value. Only by attending to these issues can we turn the data outwards to ensure that the technical system fits correctly into the organisation and its affairs.

In their different spheres, science and the law concern themselves with linking statements to reality. Businessmen must operate simultaneously in both the domains of fact and obligation and our professions should make sure that high standards apply to the information we deliver to them. For this we need a "well-justified terminology". Only if we can add criteria stringent enough to assure people that our own body of professional knowledge relates reliably to the real world can we bring information systems into the domain of science, where I think it should belong. The factual data our systems generate should meet justifiable standards of

truth. In addition, for data representing, values, judgements and decisions, our systems should place the locus of responsibility correctly, as in a good legal system.

Realising these goals would ensure that software serves its business purposes by functioning correctly as a component of an information system in the broad, social sense. We cause systems to fail by our preoccupation with their formal properties while ignoring the issues of information quality in terms of its meaning, pragmatic justification and social effects. The issues may seem rather philosophical but they have huge economic consequences.

These difficult problems have no place in computer science, nor should they have. Computer science concerns the mechanical manipulation of symbols regardless of their meaning or purpose. However, designing information systems concerns the micro-structure of a social world, where people interact to form relationships, teams, companies and communities of all kinds by means of the signs and tokens they exchange. Computer science need have nothing to say on these topics which I believe lie at the core of our professional competence.

Ontology

If we define meaning as the relationship established by people in a language community between thing-A (sign) standing for thing-B (object), we see that meanings always depend on the interpreter(s). Moreover to explain meaning we have to answer the ontological question about what fundamental sort of things exist. Note that the FRISCO report does not use "ontology" in this philosophical sense but in the vague way adopted by the Artificial Intelligence community for marketing purposes. Meaning, a foundation stone of our subject, presupposes an ontological position. Most professionals in our field would answer the question of ontology by saying that entities, attributes and relationships comprise all the kind of things we find in the world. But I favour a simpler approach based on Gibson's Theory of Affordances.

James Gibson [Gib68, Gib79], trying to discover why wartime pilots often failed to land correctly, argued that we do not perceive a ready-made reality (FRISCO's assumption [a]), but that we perceive useful invariants in the flux of signals that bombard our sense organs as we do things. Discomfort or even death face any organism which fails to do so. Perceptions arise not from sensations alone but from their combination with our actions; and we do not perceive objects that "exist independently of our own existence", rather, we perceive invariant repertoires of behaviour that the world and our own psychological constitutions afford us. In fact, without exception we can regard everything (entity, attribute, relationship, object, numbers, logical connective ...) as a repertoire of behaviour. Try it! So far I have found no exceptions. Gibson's Theory of affordances deals with physical reality.

Our interest in organisations demands that we also take social reality into account. To do this, I have suggested [Sta85] that we recognise that the regularities constituting organised behaviour define the affordances of our social reality. These social affordances result from the norms we share with other people, including the rules we agree upon and the legal norms we enact to bring order in our lives. (Note, this tallies with Searle's views [Sea95] on the construction of social reality but goes further by admitting a social role in the construction of our known physical reality.)

We can easily observe repertoires of behaviour, whether physical or social. The experiments a natural scientist performs to explore some new phenomenon (a neutrino, mad cow disease or superconductivity) always aim to explore the relevant repertoires of behaviour; similarly for social scientists exploring price elasticity, pressure groups or money laundering. Many social repertoires of behaviour we explicitly construct by writing laws or rules to regulate our behaviour: companies and contracts, for example.

I have just presented you with my ontology, my preferred answer to the question about what exists. In practice, I assert that, to all intents and purposes, nothing exists without our doing things and discovering in the flux of experience invariants worth recognising because they help us to survive and prosper physically and socially. The FRISCO ontology contained in assumptions [a, b and c] in section 3.1, I would characterise as mentalistic with a touch of naïve realism in the assumption that the world exists independently of our own existence.

Voltaire first led me to this way of thinking by his novel *Candide* or *Optimism* [Vol1795], when the hero had had his fill of Dr. Pangloss and his philosophy, he stated that we can do no more than "cultivate our garden", and there, I presume, discover stable repertoires of behaviour (sowing, cultivating, harvesting, eating ...) that we signify with other stable repertoires of sign-behaviour we call word; so that "cabbage", (the ink mark before you) denotes neither a conception in our minds nor things in the world existing independently of you and me but a repertoire of behaviour in which we can both visibly and publicly engage.

You may have noticed that I completely avoided using the verb "to be" in writing this contribution. Why? It helps to purge the mind of philosophical rubbish. For Dr. Pangloss to have denied himself the use of that verb would have saved *Candide* from the dreadful consequences of learning his philosophy "This is the best of all worlds." It would also have prevented FRISCO from repeatedly using the formula "X is Y", such as "Information is knowledge." and "Knowledge is a ... set of conceptions." and it would have compelled us to talk about doing things, such as one thing as a sign for another, and then led us to demonstrate the many properties of signs based on performing operations with them. We would have written chapters 3 and 4 in the spirit of engineering or science rather than mathematics. We would have linked our terminology to everyday, easily understandable things such as tables, chairs, cups and saucers and friends, rather than to the inaccessible conceptions and perceptions of chapters 3 and 4. By avoiding the verb "to be" I have tried to demonstrate that one can talk about anything of practical significance provided that one does so in terms of someone acting or using some instrument to do something.

“You may prove the pudding by eating it”

If you find my arguments rather theoretical, I apologise, but space does not permit me to present any remotely adequate account of my position, in particular I had reworked the case-study in chapter 5 to demonstrate its effectiveness in engineering terms, using a meta-model based on a logic of social action which always forces the user to specify the responsible agent and the structure of repertoire of behaviour through which that agent relates her word to reality. For a copy please contact me (e-mail: r.k.stamper@sms.utwente.nl). Meanwhile many of the ideas you will find in [Sta96]: "Signs, Information, Norms and Systems".

FRISCO Reflected Upon: So What or Aha?

by Alexander A. Verrijn-Stuart

Some readers of the final FRISCO Report may initially experience a “So What” feeling. Sure, the problem statement is relevant, the argument is developed consistently, appropriate elements have been borrowed from other disciplines, the number of definitions is kept limited, helpful links are provided to the world of those whose thinking differs. But what is new, and is any “Aha Erlebnis” triggered? The answers are: *lots* (of new insights) and *yes* (there are a number of reasons to say Aha!).

Firstly, let me assert that the FRISCO observation (that there is harmful confusion) is correct and would seem to apply even more strongly today than it did when the task group was set up in 1988. On the other hand, I contend that the current terminological fuzziness did not cause any confusion but merely is a result of it. FRISCO correctly gauges a climate of interested parties not understanding what other interested parties are interested in. The question is whether a more precise language is a comprehensive answer to the problem. It should help, but discussing what that language is in aid of is of greater help. Which is what FRISCO is all about.

An information-technology specialist may think of him- or herself as someone who always tries to understand the client. Yet, forgetting one’s upbringing and training is impossible. The specialist’s inbred preference for formal expression will be lost on the client. A manager may say that he intends considering all issues, but cannot divorce himself from a natural tendency to think in terms of goals and objectives (rather than specifications for the provision of information). The information user may fully comprehend an organisational task and all its implications, yet fail to take in the subjective personal knowledge aspects (as opposed to any generally accepted more or less “objective” knowledge). I would like to suggest that the description of that climate and the expose of its consequences is FRISCO’s strong point.

To underpin my thesis, let me review what I see as the triple basis of FRISCO’s innovative thinking: the “constructivist” position, the recognition of semiotic levels, and the nature of the models employed in describing organisationally relevant communication. The fourth element, FRISCO’s view on systems, follows on from its metaphysical and ontological position selected¹.

a) CONSTRUCTIVISM VS. OBJECTIVISM [REF. CHAPTER 3]

The natural sciences have no deep philosophical problem talking about “reality”. A theory is a model that provides a capability for prediction. If some experiment comes up with fully contradictory results, then the theory must be revised. Modern theories predict (practically) deterministic results, or produce outcomes that may be taken as levels of probability.

Scientists feel happy with such pronouncements. While no person is capable of reading all scientific papers in which all theories are set out, most will accept that the documents in

¹The original FRISCO “temple” was characterised by a base, representing its metaphysical approach, three pillars (systems, semiotics, models) and a crowning pediment standing for the FRISCO “framework” synthesis. In the view presented here, the base is common sense, whereas the pillars of wisdom now are *Constructivism-cum-systems*, *Semiotics* and *Models*, respectively. Rather than constituting an ever shaky temple, all blend perfectly in to the glory of a solid pyramid.

question are convincing to the initiated. The fact that a theory is couched in probabilistics - such as statistical mechanics, quantum mechanics, worm holes, chaos theories and the like - does not mean that the scientific community has given up its belief in “reality”. It merely expresses the extent to which one can use predictions. For practical purposes, the theory denotations (i.e. the scientific papers) may be considered representations of “shared knowledge” and what is stated in them may be treated as if it referred to an objective truth.

The social sciences have greater difficulties. The “scientific method” still applies (even if proponents of “alternative” approaches preach rejection of “inadequate” paradigms), but three problems are felt more strongly here, viz. (1) the objects of study are far more subject to variability, (2) the observers are part of the study object and have an influence which is of the same magnitude as the phenomenon under study¹, and (3) generally it is impossible to repeat significant experiments (especially in economics) or to run true control experiments (especially in life critical situations). However, these are merely problems of degree, not of principle. With reservations, one may still talk about findings as if they are associated with a real existence.

In the information systems domain, there is an additional problem, viz. the fact that its representations (data, messages, etc.) are essentially based on interpretation (i.e. personal judgement). That is not to say that a stock level would not be acceptable to someone who did not measure it himself, nor that the recorded balance of some account should never be trusted. What it does mean is that the information one tries to obtain (when querying, say, a database) or the message by which one tries to influence an intended receiver (when communicating), from the point of view of decision making, may be quite insufficient to generate the same level of acceptance and meaning for both originator and ultimate receiver. The simple example I always like is the use of a road map. We all accept that it is a considerable abstraction of the knowledge one might possibly gain from a professional cartographer or travel writer, but worse, we must admit that abstractions of this kind are open to a wide variety of interpretations (who has not got lost at least once as a result of mis-interpreting some map?). Thus, the information technology specialist and the information user must both be aware of the fact that what is represented must not be identified so much with “reality” as with intentions of triggering action (or inaction).

Acceptance of the FRISCO constructivist view as a basis of what may be known (in other words, to assume that “knowledge” is essentially personal²) has most important consequences for the understanding of what information systems (may) do for us. In connection with the systems aspects per se, it does mean that a system only exists in so far as it is perceived by the system viewer. We will revert to these points when we discuss their implications.

b) THE SEMIOTIC LADDER [ref. Chapters 3, 6]

In a way, recognition that there are different levels at which things may be addressed is an extension of the same argument. The information systems domain is concerned with

¹This also holds for the natural sciences, of course. It gives rise, for instance, to the Heisenberg uncertainty principle. However, the natural science observer may be treated as a standard interference. In social sciences, it is very hard to divorce the individual person from the experiment.

²FRISCO’s version of “constructivism” does not imply that nothing exists outside the mind (that would have constituted “mentalism”), but that what is important in information systems (arriving at mutual agreement about objects of action by means of communication) is not as easily reduced to “shared knowledge” as is the case in natural science theories.

representations, the expression of which requires a language with syntax, terms and productions. In order to get across at all, they must be carried physically. To be understood, they must also have (more or less) agreed semantics. And to be effective, they must be accepted according to some pragmatics. This multifaceted nature resembles the two characteristics form and content, in speech or art. It is like the need to distinguish data, processes and events in system development. Similarly, the meaning triangle links the various aspects of the sign concept - extended under FRISCO usage by including the sign interpreter so as to become a tetrahedron. Any “expression” requires consideration of multiple angles.

The same applies, a fortiori, to the interpretation of perceptions and to the ways of passing on the conceptions one has formed. What is important is not only the split between personal (mental) conceptions and interpersonal (physical) messages, but even more so the fact that “conceiver” and “perceiver” almost by their very nature associate different aspects with otherwise jointly available representations.

It is interesting to see that, when a statement is made, a “meaning” may be associated at each of the six levels of the semiotic ladder:

- Physical level: cause-and-effect of a physical token stream
- Empirical level: code equivalence of symbol strings
- Syntactical level: formula equivalence of expressions
- Semantical level: (constructivist) agreement arranged, (objectivist) agreement assumed
- Pragmatic level: intention (of sign creator, sign interpreter)
- Social level: fitting in or altering norm structure

FRISCO’s contribution has been to highlight this multiplicity of potential angles. Especially, the ever-present danger that two persons subconsciously associate different semiotic aspects with the same representation is recognised. Where some system designer may be concerned primarily with the efficiency of some information system feature, the user for whom the work is undertaken may think that effectiveness is being discussed. Where one person identifies meaningfulness with correctness and completeness (of records, say), the other may strive towards ease of use and clarity.

The recognition of semiotic multiplicity has important consequences for the understanding of the (kinds of) services that are or may be rendered by information systems, in the broadest sense. We will revert to the implications this has for users and designers of computerised systems.

c) THE NATURE OF COMMUNICATION [ref. chapters 2, 3]

The FRISCO “line of reasoning” (in Chapter 2) advocates a shift from the customary “how-perspective” to the broader “why-and-which-effects-perspective”. Elsewhere, it calls for a change of viewpoint from “data and communication” (the latter in the sense of data transfer) to “human communication and ... value addition”. In other words, the objective is not the technology as such, but the rendering of assistance to decision making and other organisational benefits. Now, what kind of communication is possible and what would be its uses?

Although subscribing to all question-answer pairs in Chapter 2, let me single out the fourth one: “*what is actually communicated - exclusively, conceptions or models are communicated*”. Acts of communication involve the exchange of signs. These are physical representations of what is intended (i.e. they can stand for some thing but they do not equal it). At this (the *physical*) level of the semiotic ladder, the main limitation to what may be expressed lies in time

and channel constraints. However, and more importantly, their representational power can only extend to part of what the communicator “has in mind”. Even if, at the *semantic* level, there is no end to what might be added, at some point, one has to be precise, meaning that one is limited in expressing one’s conceptions. A sign may “represent” a whole thing, but nevertheless constitutes no more than an associated abstraction. The FRISCO term for that deliberately abstracted description is “model”.

While the idea that representations - computerised or not - can cover only some of the domain one wishes to say something about is fairly generally accepted, the implication *is not* that one should endlessly refine one’s models and collect as much data as possible. The proper conclusion *is* that one must - creatively - restrict oneself to the most effective (value adding) formulation for the task or problem in question. As I shall demonstrate in a moment, small is beautiful. We will revert to this lesson in a moment.

CRITICISM

As an interlude, a few words of (self) criticism. One of FRISCO’s aims has been to achieve full consistency and, if possible, a (basis for a) professional language. In this, success has not been complete. As a compromise, a number of alternative terms had to be accepted. Some useful concepts had to be left out. Where FRISCO’s preferred term had to deviate from the most common usage this was almost always because FRISCO introduced a deeper analysis. These oddities - mostly resulting from the constructivist approach - should not pose problems to a reader with a positive attitude. And those with a negative attitude can never be satisfied, anyway.

One concept that remains open ended is the “system”. Early on, it was declared a “difficult” word (along with knowledge, information and communication). The difficulty derives from the choice of constructivism as a philosophical point of departure. Its logical consequence is that a system, treated as a special model, is as personal as knowledge. This sounds attractive, for there are many ways to look upon something complex - the system viewer determines in what sense it must be seen as a system. This is not a problem when one views (parts of) an organisation, i.e. when one considers the “information system in the broader sense” nor in the early “analytical” stages of developing a computerised sub-system. However, it is once a design is beginning to take shape, or already before that, as soon as the first formalised description of the domain is discussed.

Strictly speaking, an extensive negotiation would be necessary to get acceptance of any such system - not because of what is put into it (the user-designer dialogue), but because any future user of its implementation would have to be convinced over and over again that it must be seen as a system with the appropriate systemic properties!

If this is an interpretation of what teaching the use of a computerised sub-system is, so be it. It certainly covers the understanding of the information system in the broader sense, which is subject to continual change and which does require a lot of reflection and study. However, if FRISCO insists that the term “system” can only be used to cover something that is the result of a personal interpretation, one will find that the common usage is at variance with it. What is really meant, of course, is that the intelligent user of the word is (supposed to be) aware of the personal interpretation aspects and knows that it is short for longer expressions such as “computerised implementation of the conception of ... ”¹.

¹In slightly different words, the same observations are made in chapter 6, section 6.2 on system concepts, where it is remarked that the use of the term 'system' in the more specific sense of data-processing system or similar is merely a conveniently short version of the more elaborate expression otherwise needed.

As a reasonable person, I will accept, or at least consider, reasonable criticism from others. But this should work both ways. As a member of the FRISCO Task Group, I remember all too well how long it took before we became fully aware of the distinction between information systems in the broader and narrower senses. By now this would seem to be expressed loudly and clearly, throughout, as in the opening paragraph of Chapter 3:

“Information systems” concern the use of “information” by persons and groupings of persons in organisations, in particular through computer-based systems. A proper understanding of the nature of such systems requires viewing them in the context of the organisation that employs them.

It is amazing that some critics are still under the impression that the FRISCO framework exclusively applies to technical aspects of computerised systems and not to their organisational fit and significance. But, as remarked above, readers with a negative attitude can never be satisfied, anyway.

LESSONS

Now, for the lessons. A number of findings and formulations of FRISCO have a direct bearing on the practical world of information systems development and usage. In the preamble, the report is (correctly) said to be “particularly useful for everyone who is concerned in some way or another with the problems and issues of communication and information in organisations, enterprises and business environments ...”. Well, these are the three points I should like to make:

Aha 1: FRISCO & Constructivism

In FRISCO’s view, knowledge is essentially personal. Information is viewed as the “personal knowledge increment brought about by a receiving action in a message transfer” i.e. as a result of communication (mostly for the purpose of decision making). The lesson for the systems designer, therefore, is not just to ask for the “information” the user should like to get out of a computerised system. Instead, one must *ask for the way in which* the user’s *decisions* - in the organisational context - *might be supported by* additional and/or completed (personal and shared) *knowledge*.

This change of approach is in line with the thrust of chapter 2.

Warning:

If it is already difficult for the information-technology specialist not to interfere too much with the user when asking for the information desired, querying a client on his/her decision making procedures requires even more tact! But, ultimately, it will lead to more effective specifications.

Aha 2: FRISCO, Semiotics & the Message

In the information system in the broader sense, there is a continuous exchange of messages. Their physical evidence is in bit streams and sound bites, their semantics and pragmatics are naturally absorbed by the members of the organisation. Such a culturally acquired pattern of actions and reactions is not as easily established when one uses computerised systems.

Computer *literacy instruction* should, for instance, stress the *distinction* between

- (1) what is in a record (the bits & bytes of a database, document, message, and so on) and what linguistic constraints may apply (i.e. the *representational* and *syntactical* aspects), as opposed to
- (2) what interpretations and intentions can and cannot be associated with it (i.e. the *semantical*, *pragmatic* and *social* aspects).

Understanding the various semiotic angles makes information users more mature!

Warning:

Since computerised systems, in the minds of those exposed to them, constitute a form of “automation”, it will be hard to induce interpretation rather than automatic reaction. The understanding required is for the thrust of the information system rather than the technology as such.

Aha 3: FRISCO, Models & Abstractions

One of the most important insights to be gained regarding the role of information is that any *model* employed has a *limited coverage*. This point may be reversed to great advantage: If only restricted abstractions can be represented, then look especially for those that are most informative! Also, the more straightforward the message content, the stronger the message. In a full analysis of information economics, one finds that the first step contributes considerably to knowledge acquisition, but subsequent steps invariably show diminishing returns.

For information systems, especially of the computerised kind, *small is beautiful*.

Warning:

Do not confuse the contents of an administrative system with “information”. There may be compelling reasons for extending refining a system, but capturing more “meaning” is not one of them. Multi-purpose tools, as you know from DoItYourself practice, always perform more poorly than the small specialised set you have built up over the years.

CONCLUSION

All in all, FRISCO has covered a lot of ground that others visited before. But where its predecessors got lost or sidetracked, FRISCO did find a way around numerous traps. Its findings may not all be new, but their juxtaposition is. Where similar ideas had been known before, they have been either consistently ignored or else preached by an isolated few only.

In spite of some unavoidable “So What” (all that detail), there are several refreshing experiences to be had (some lateral thinking, some simple idea combinations) : “*Aha*” *indeed* !

REFERENCES

- Abr74 J.R. Abrial: Data Semantics, in J.W. Klimbie and K.L. Koffeman (eds.): Data Base Management, North-Holland, Amsterdam, 1974
- Ack71 R.L. Ackoff: Towards a System of System Concepts, Management Science Vol.17, July 1971
- AE72 R.L. Ackoff and F.E. Emery: On Purposeful Systems, Aldine-Atherton, 1972
- AHL92a E. Auramäki, R. Hirschheim and K. Lyytinen: Modelling Offices Through Discourse Analysis: The SAMPO Approach, in The Computer Journal, 35, No. 4, 1992
- AHL92b E. Auramäki, R. Hirschheim and K. Lyytinen: Modelling Offices Through Discourse Analysis: A Comparison and Evaluation of SAMPO and OSSAD and ICN, in The Computer Journal, 35, No. 5, 1992
- ALL88 E. Auramäki, E. Lehtinen and K. Lyytinen: A Speech-Act-Based Office Modeling Approach, in ACM Transactions on Office Information Systems, 6(2), 1988
- Ama87 The AMADEUS Project, Final Report, Interprogram, Amstelveen, The Netherlands, June 1987
- And90 P.B. Andersen: A Theory of Computer Semiotics, University Press, Cambridge, 1990
- ANN96 P.B. Andersen, M. Nadin and F. Nake: Informatics and Semiotics, IBFI - Schloss Dagstuhl, University of Saarland, Saarbrücken, Germany, 1996
- Aus62 J.L. Austin: How to Do Things with Words, Harvard University Press, 1962
- Bar64 Y. Bar-Hillel: Language and Information: Selected Essays on their Theory and Application, Addison-Wesley, Reading, Mass., 1964
- Ber50 L. von Bertalanffy: An Outline of General Systems Theory, British Journal of the Philosophy of Science, Vol.1, pp. 134-164, 1950
- BFM79 B. Breutmann, E.D. Falkenberg and R. Mauer: CSL: A Language for Defining Conceptual Schemas, in G. Bracchi and G.M. Nijssen (eds.): Data Base Architecture, North-Holland, Amsterdam, 1979
- BL66 P. Berger and T. Luckmann: The Social Construction of Reality, Doubleday, New York, 1966
- Bou56 K. Boulding: General Systems Theory - The Skeleton of Science, Management Science, Vol. 2, April 1956
- BS88 G. Bergheim and E. Sandersen: A Taxonomy of Concepts for the Science of Information Systems, 1988
- Bun77 M.A. Bunge: Ontology I: The Furniture of the World, Vol. 3 of Treatise on Basic Philosophy, D. Reidel Publishing Company, The Netherlands, 1977
- Bun79 M.A. Bunge: Ontology II: A World of Systems, Vol. 4 of Treatise on Basic Philosophy, D. Reidel Publishing Company, The Netherlands, 1979
- Car47 R. Carnap: Meaning and Necessity, University of Chicago Press, Chicago, 1947
- CB53 R. Carnap and Y. Bar-Hillel: Semantic Information, British Journal of the Philosophy of Science, Vol. 4, pp. 147-157, 1953
- Che57 C. Cherry: On Human Communication, MIT Press, Cambridge, Mass., 1957
- Che75 P.P. Chen: The Entity-Relationship Model – Toward a Unified View of Data, Proceedings of the First Conference on Very Large Data Bases, Framingham, Mass., September 1975
- Che81 P.B. Checkland: Systems Thinking, Systems Practice, Wiley, 1981
- Cho68 N. Chomsky: Language and Mind, Harcourt, Brace, Jovanovich, New York, 1968

- Col67 Colliers Encyclopedia, Crowell Collier and Macmillan Inc., 1967
- Daf86 R.L. Daft: Organisation Theory and Design, West Publishing Co, Minnesota, 1986
- DAI88 DAISEE Report No. 97, University of Trondheim, Norway, 1988
- Deu81 Peter Deutsch: In Records of a Discussion on the Type Concept, Proceedings of the Workshop on Data Abstraction, Databases, and Conceptual Modelling, Pingree Park, Colorado, Co-published in SIGART Newsletter No.74, SIGMOD Record Vol. 11, No. 2, and SIGPLAN Notices Vol.16, No. 1, ACM, New York, 1981
- Dru92 P.F. Drucker: Managing for theFuture: The 1990s and Beyond, ISBN 0-452-26984-9, 1992
- DS90 Th.H. Davenport and J.E. Short: The New Industrial Engineering: Information Technology and Business Process Redesign, Sloan Management Review, Summer 1990
- Ear89 M.J. Earl.: Management Strategies for Information Teohnology, Prentice Hall, ISBN O-13-551656-O, 1989
- Fal76 E.D. Falkenberg: Concepts for Modelling Information, in G.M. Nijssen (ed.): Proceedings of the IFIP TC 2 Conference on Modelling in Data Base Management Systems, Freudenstadt, Germany, North-Holland, 1976
- Fal77 E.D. Falkenberg: Co-existence and Transformation of Data, in Proceedings of the 3rd International Conference on Very Large Data Bases, Tokyo, 1977
- Fal93a E.D. Falkenberg: An Approach to Deterministic Event-Tuned Information Analysis, in Proceedings of the International Conference on NIAM-ISDM'93, Utrecht, The Netherlands, 1993
- Fal93b E.D. Falkenberg: DETERM: Deterministic Event-Tuned Entity-Relationship Modeling, in R.A. Elmasri, V. Kouramajian and B. Thalheim: Proceedings of the 12th International Conference on the Entity-Relationship Approach, Arlington, Texas, December 1993, Lecture Notes in Computer Science, No. 823, pp. 230-241, Springer-Verlag, Berlin, Heidelberg, 1994
- FGHW88 F. Flores, M. Graves, B. Hartfield and T. Winograd: Computer Systems and the Design of Organizational Interaction, in ACM Transactions on Office Information Systems, 6 (2), 1988
- FHO95 E.D. Falkenberg, W. Hesse and A. Olivé (eds.): Information System Concepts: Towards a Consolidation of Views, Proceedings IFIP WG8.1 Conference ISCO3, Marburg, Germany, March 1995, Chapman & Hall, 1995
- Fil68 G.J. Fillmore: The Case for Case, in F. Bach and R. Harms (eds.): Universals in Linguistic Theory, Holt, Rinehart & Winston, New York, 1968
- FL87 E.D. Falkenberg and P. Lindgreen: Proposal to TC8 for the establishment of the FRISCO Task Group, IFIP WG 8.1 Memo, February 1987
- FL89 E.D. Falkenberg and P. Lindgreen (eds.): Information System Concepts: An In-depth Analysis, Proceedings IFIP WG 8.1 Conference ISCO1, Namur, Belgium, October 1989, North-Holland, 1989
- FOP91 E.D. Falkenberg, J.L.H. Oei and H.A. Proper: A Conceptual Framework for Evolving Information Systems, in H.G. Sol and R.L. Crosslin (eds.): Dynamic Modelling of Information Systems II, pp. 353-375, North-Holland, Amsterdam, The Netherlands, 1992
- FOP92 E.D. Falkenberg, J.L.H. Oei and H.A. Proper: Evolving Information Systems: Beyond Temporal Information Systems, in A.M. Tjoa and I. Ramos (eds.): Proceedings of the Data Base and Expert System Applications Conference (DEXA 92), pp. 282-287, Valencia, Spain, September 1992, Springer-Verlag, 1992
- FRE92 E.D. Falkenberg, C. Rolland and E.N. El-Sayed (eds.): Information System Concepts: Improving the Understanding, Proceedings IFIP WG 8.1 Conference ISCO2, Alexandria, Egypt, April 1992, North-Holland, 1992
- Fuc92 K. Fuchs-Kittowski: Reflections on the Essence of Information, in C. Floyd et al. (eds.): Software Development and Reality Construction, pp. 416-432, Springer, Berlin, 1992

- FW88 E.D. Falkenberg and Th.P. van der Weide: The TOP Model: A Meta-model for Conservative Data Bases, Technical Report, Department of Information Systems, University of Nijmegen, The Netherlands, 1988
- Gar65 G.N. Garmonsway: The Penguin English Dictionary, Penguin Books Ltd., 1965
- GD85 C.F. Gibson and Th.H. Davenport: Systems Change: Managing Organizational and Behavioral Impact, Information Strategy, The Executives Journal 2:1, 23-27, Fall 1985
- Gib+84 C.F. Gibson et al.: Strategies for Making an Information System Fit Your Organization, Management Review, 8-14, January 1984
- Gog92 J. Goguen: The “dry” and the “wet”, Invited paper, in FRE92
- Hab84 J. Habermas: The Theory of Communicative Action, *Social Science Journal*, Vol. 16, No. 3, MIT Press, Boston, 1984
- HAKP96 B. Holmqvist, P.B. Andersen, H. Klein and R. Posner (eds.): Signs at Work, De Gruyter, Berlin, 1996
- Hal91 T.A. Halpin: A Fact-oriented Approach to Schema Transformation, in B. Thalheim, J. Demetrovics and H.-D. Gerhardt (eds.): Proceedings of MFDBS 91, Rostock, Germany, Lecture Notes in Computer Science Vol. 495, Springer-Verlag, 1991
- Hal93 T.A. Halpin: What is an elementary fact?, in Proceedings of the International Conference on NIAM-ISDM'93, Utrecht, The Netherlands, 1993
- Hal95 T.A. Halpin: Conceptual Schema and Relational Database Design, 2nd edition, Prentice Hall Australia, Sydney, 1995
- Ham90 M. Hammer: Reengineering Work: Don't Automate, Obliterate, Harvard Business Review, July-August 1990
- HBO94 S. Brinkkemper, F. Harmsen and J.L.H. Oei: Situational Method Engineering for Information Systems Project Approaches, in Proceedings of the IFIP WG 8.1 CRIS Conference, Maastricht, The Netherlands, September 1994
- Hes92 W. Hesse: Can Machines Think? Reflections on Computers, Mind and Responsibility, Keynote address, Tandem Users' Journal, Chicago, Ill., March/April 1992
- HKL95 R. Hirschheim, H.K. Klein and K. Lyytinen: Information Systems and Data Modelling: Conceptual and Philosophical Foundations, Cambridge University Press, 1995
- HO92 T.A. Halpin and J.L.H. Oei: A Framework for Comparing Conceptual Modelling Languages, Technical Report 92-29, Department of Information Systems, University of Nijmegen, The Netherlands, 1992
- Hof93 A.H.M. ter Hofstede: Data Modeling in Data Intensive Domains, Doctoral Thesis, University of Nijmegen, The Netherlands, 1993
- HPW92 A.H.M. ter Hofstede, H.A. Proper and Th.P. van der Weide: Data Modeling in Complex Application Domains, in P. Loucopoulos (ed.): Proceedings of the Fourth International Conference on Advanced Information Systems Engineering (CAiSE'92), Manchester, U.K., Lecture Notes in Computer Science Vol. 593, Springer-Verlag, 1992
- HS70 J. Hintikka and P. Suppes: Information and Inference, Reidel, Dordrecht, 1970
- HW31 C. Hartshorne and P. Weiss (eds.): Collected Papers of Charles Sanders Peirce, Harvard University Press, Cambridge, Mass., 1931
- IFIP88 The FRISCO Manifesto, IFIP WG 8.1, 1988
- Iiv83 J. Iivari: Contributions to the Theoretical Foundations of System Engineering Research and the PIOCO Model, Acta Universitatis Ouluensis, Ser. A, No. 150

- ISO82 ISO TC97/SC5/WG3, J.J. van Griethuysen (ed.): Concepts and Terminology for the Conceptual Schema and the Information Base, Publ. No. ISO/TC 97/SC5 – N 695, 1982
- ISO95 ISO/IEC JTC1 SC21/WG3/CSMF SOU 5, as transmitted by R. Hotaka in April 1995
- Kee81 P.G.W. Keen: Information Systems and Organizational Change, Communications of the ACM, 28 (1), 24-33, 1981
- Khi57 A.I. Khinchin: Mathematical Foundations of Information Theory, Dover, New York, 1957
- Kli69 G.J. Klir: An Approach to General Systems Theory, Van Nostrand Reinhold, New York, 1969
- KLR96 S. Kelly, K. Lyytinen and M. Rossi: MetaEdit+: A Fully Configurable Multiuser and Multitool CASE Environment, in Proceedings of the 8th Conference on Advanced Information Systems Engineering (CAiSE'96), Springer-Verlag, Berlin, 1996, pp. 1-21
- Kol84 D. Kolb: Experiential Learning, Prentice-Hall, Englewood Cliffs, N.J., 1984
- Kos76 C.H.A. Koster: Visibility and Types, Sigplan Notices, Vol. 8, No. 2, 1976
- KW92 K. Kumar and R.J. Welke: Methodology Engineering: A Proposal for Situation-specific Methodology Construction, in W.W. Cotterman and J.A. Senn (eds.): Challenges and Strategies for Research in Systems Development, Wiley, 1992
- Lan63 B. Langefors: Some Approaches to the Theory of Information Systems, BIT 3, pp. 229-254, 1963
- Lan66a B. Langefors: Samband mellan object och verklighet (Model Field and Object Field Correspondence), TB-ADB 66, Stockholm University, Sweden, 1966
- Lan66b B. Langefors: Theoretical Analysis of Information Systems, Studentlitteratur, Lund, 1966
- Lan71 B. Langefors: Editorial Notes to: Computer Aided Information Systems Analysis and Design, Studentlitteratur, Lund, 1971
- Lan77 B. Langefors, Information Systems Theory, Information Systems 2, 207, 1977
- Law88 D. Law: Methods for Comparing Methods: Languages in Software Development, NCC Publications, 1988
- Lee50 D. Lee: Codifications of Reality; Lineal and Non-lineal, Psychosomatic Medicine 12, No. 2, USA, 1950
- Lee76 G Leech: Semantics, Penguin Books, Edition printed in Great Britain, 1976
- Leo88 D. Leonard-Barton: Implementation Characteristics of Organizational Innovations, Communication Research 15:5, pp. 603-631, October 1988
- Lin90a P. Lindgreen (ed.): A Framework of Information System Concepts, Interim Report of the IFIP WG 8.1 Task Group FRISCO, May 1990 (out of print)
- Lin90b P. Lindgreen: The Sub-system Relationship, FRISCO Working Paper PL.15, December 1990
- Lin92a P. Lindgreen: Pillar one: Systemic Concepts. Position Paper for the IFIP WG 8.1 Review Meeting, Alexandria, April 1992
- Lin92b P. Lindgreen: A General Framework for Understanding Semantic Structures, in [FRE92], 1992
- LL86 E. Lehtinen and K. Lyytinen: "An Action-based Model of Information Systems", Information Systems, Vol. 13, No. 4, pp. 299-318, 1986
- Mar83 L. Mark: What is the Binary Relationship Approach? in C.G. Davis et al. (eds.): Proceedings of the 3rd International Conference on the Entity-Relationship Approach, Anaheim, California, North-Holland, 1983
- McC91 J.V. McCee: Implementing Systems Across Boundaries: Dynamics of Information Technology and

- Integration, Ph.D. diss., Harvard Business School, 1991
- McF84 F.W. McFarlan: Information Technology Changes the Way You Compete, Harvard Business Review, May-June 1984
- Mea72 D.L. Meadows: The Limits to Growth, Universe Books, New York, 1972
- MH79 E. Mumford and D. Henshall: A Participative Approach to Computer Systems Design, Associated Business Press, London, 1979
- Mil73 G. Miller (ed.): Communication, Language, and Meaning, Psychological Perspectives, Edition USA 1973, ISBN 0-465-09721-9
- Moo+82 E.C. Moore et al. (eds.): Writings of Charles S. Peirce: A Chronological Edition, Indiana University Press, Bloomington, Indiana, 1982
- Mor46 C. Morris: Signs, Language and Behaviour, Prentice Hall Braziller, New York, 1946
- MSW91 J.B. Michael, E.H. Sibley and R.L. Wexelblat: A Modeling Paradigm for Representing Intention in Information Systems Policy, Proceedings of the First Workshop on Information Technologies and Systems, Massachusetts Institute of Technology, Sloan School of Management, 1991
- Mum83 E. Mumford: Designing Human Systems, Manchester Business School, U.K., 1983
- MWFF93 R. Medina-Mora, T. Winograd, R. Flores and F. Flores: The Action Workflow Approach to Workflow Management Technology, in Proceedings of the Third European Conference on Computer Supported Co-operative Work, Milano, Italy, 1993
- Nau72 D. Nauta: The Meaning of Information, Mouton, The Hague, 1972
- NH89 G.M. Nijssen and T.A. Halpin: Conceptual Schema and Relational Database Design: A Fact-oriented Approach, Prentice-Hall, Sydney, Australia, 1989
- Nil79 B.E. Nilsson: On Models and Mappings in a Data Base Environment - A Holistic Approach to Data Modeling, Statistics, Sweden, 1979, ISBN 91-38-05221-0
- Nis92 H.-E. Nissen: Let's Not Devise Another Phlogiston Theory; in [FRE92], pp. 193-208, 1992
- Oco75 D.J. O'Connor: The Correspondence Theory of Truth, Hutchinson, U.K., ISBN 0-09-123200-7, 1975
- Oei94 J.L.H. Oei: The Meta Model Hierarchy Approach: A Formalisation, Technical Report, University of Twente, The Netherlands, 1994
- Oei95 J.L.H. Oei: A Meta-model Transformation Approach Towards Harmonisation in Information System Modelling, in [FHO95], 1995
- OF94 J.L.H. Oei and E.D. Falkenberg: Harmonisation of Information System Modelling and Specification Techniques, in A.A. Verrijn-Stuart and T.W. Olle (eds.): Methods and Associated Tools for the Information System Life Cycle, Maastricht, The Netherlands, pp. 151-168, IFIP/North-Holland, 1994
- Oll+88 T.W. Olle, J. Hagelstein, I.G. MacDonald, C. Rolland, H.G. Sol, F.J.M. Van Assche and A.A. Verrijn-Stuart: Information Systems Methodologies: A Framework for Understanding, Addison-Wesley, Reading, Massachusetts, 1988
- OPF92 J.L.H. Oei, H.A. Proper and E.D. Falkenberg: Evolving Information Systems: Meeting the Ever-changing Environment, Technical Report 92-24, Department of Information Systems, University of Nijmegen, The Netherlands, 1992
- Pet81 J.L. Peterson: Petri Net Theory and the Modelling of Systems, Prentice-Hall, Englewoods Cliffs, New Jersey, 1981
- PM85 M.E. Porter and V.E. Millar: How Information Gives You Competitive Advantage, Harvard Business Review, July-August 1985

- Pos73 R. Posner: *Cognition: An Introduction*, Scott Foresman, Glenview, 1973
- Pro94 H.A. Proper: *A Theory of Conceptual Modelling of Evolving Application Domains*, Doctoral Thesis, University of Nijmegen, The Netherlands, 1994
- Qui63 W.V.O. Quine: *From a Logical Point of View*, Library of Congress, No. 6115277, USA, 1963
- Rol87 C. Rolland (ed.): *Proceedings of the TAIS Conference*, Sophia Antipolis, France, May 1987
- Rus76 B. Russell: *An Inquiry into Meaning and Truth*, 8th ed., George Allen and Unwin, London, 1976
- SB93 K. van Slooten and S. Brinkkemper: *A Method Engineering Approach to Information Systems Development*, in C. Rolland, N. Prakash and B. Pernici (eds.): *Proceedings of the IFIP WG 8.1 Conference on Information Systems Development Process*, 1993
- Sca82 G. Scarrott: *Some Consequences of Recursion in Human Affairs*, IEE Proc., Vol. 129, Pt. A, No. 1, January 1982
- Sea69 J.R. Searle: *Speech Acts - An Essay in the Philosophy of Language*, Cambridge University Press, Cambridge, 1969
- Sea79 J.R. Searle: *A Taxonomy of Illocutionary Acts*, in *Expression and Meaning: Studies in the Theory of Speech Acts*, Binghamton, New York, Cambridge University Press, 1979
- Sea95 J.R. Searle: *The Construction of Social Reality*, Simon and Shuster, New York; Allen Lane, The Penguin Press, London, 1995; Paperback in *Penguin Philosophy*, ISBN 0-14-023590-6, 1996
- Sha73 R. Shank: *Identification of Conceptualizations Underlying Natural Language*, in Shank and Colby: *Computer Models of Thought and Language*, W.H. Freeman, USA, 1973
- SL94 R.K. Stamper and K. Liu: *Organisational Dynamics, Social Norms and Information Systems*, Proc. HICSS-27, Hawaii, 1994
- SMW91 E.H. Sibley, J.B. Michael and R.L. Wexelblat: *An Approach to Formalizing Policy Management*, Proceedings of the Second International Conference on Economics and Artificial Intelligence, Pergamon Press, Oxford, U.K., 1991
- SS77 J.M. Smith and D.C.P. Smith: *Data Base Abstractions: Aggregation and Generalisation*, ACM TODS 2, June 1977
- Sta71 R.K. Stamper: *Some Ways of Measuring Information*, Computer Bulletin, Vol.15, pp. 432-436, 1971
- Sta73 R.K. Stamper: *Information in Business and Administrative Systems*, Batsford, London, 1973, and Wiley, New York, 1973
- Sta85 R.K. Stamper: *A Logic of Social Norms for the Semantics of Business Information*, in T. Steel and R. Meersmann (eds.): *Knowledge and Data*, North Holland, 1985
- Sta94 R.K. Stamper: *Social Norms in Requirements Analysis - an Outline of MEASUR*", in M. Jirotko and J. Goguen (eds.): *Requirements Engineering: Technical and Social Aspects*, Academic Press, 1994
- Sta96a R.K. Stamper: *Signs, Information, Norms and Systems*, in [HAKP96], 1996
- Sta96b R.K. Stamper: *Organisational Semiotics*, in F. Stowell and J. Mingers (eds.): *Information Systems: An Emerging Discipline?*, McGraw-Hill, London and New York, 1996
- Sun74 B. Sundgren: *Conceptual Foundation of the Infological Approach to Data Bases*, in J.W. Klimbie and K.L. Koffeman (eds.): *Data Base Management*, North-Holland, 1974
- SV85 J.R. Searle and D. Vanderveken: *Foundations of Illocutionary Logic*, Cambridge University Press, Cambridge, 1985
- SW49 C.E. Shannon and W. Weaver: *The Mathematical Theory of Communication*, University of Illinois Press, USA, 1949
- Tho74 R.H. Thomason (ed.): *Formal Philosophy: Selected Papers of Richard Montague*, Yale University

- Press, New Haven and London, 1974
- TKO97 I. Tervonen, P. Kerola and H. Oinas-Kukkonen: An Organizational Memory for Quality-Based Software Design and Inspection – a Collaborative Multiview Approach with Hyperlinking Capabilities, To appear in Proceedings of 30th HICSS Conference, 1997
- Ver87 A.A. Verrijn-Stuart: Themes and Trends in Information Systems, Computer Journal 30, pp. 97-109, 1987
- Vos96 K. Voss: Modelling Information Systems and Database Systems by Petri Nets, Working Paper 1004, GMD, Germany, June 1996
- Web47 M. Weber: Theory of Social and Economic Organization, Free Press, New York, 1947
- Who78 B.L. Whorf: Language, Thought and Reality, Edition USA 1978, ISBN 0-262-73006-5
- Wil91 Willars: Amplification of Business Cognition through Modelling Techniques, Proc. of the 11th IEA Congress, Paris, 1991
- Wit22 L. Wittgenstein: Tractatus Logicus Philosophicus, 1922, Edition UK 1977, ISBN 0-7100-7923-0
- Wit53 Ludwig Wittgenstein: Philosophical Investigations, 1953, Edition UK 1976, ISBN 0631-11900-0
- Wri63 G.H. von Wright: Norm and Action, Routledge and Kegan Paul, London, 1963
- You89 E. Yourdon: Modern Structured Analysis, Prentice-Hall, Englewood Cliffs, New Jersey, 1989

GLOSSARY¹

Absolute time [E11, D14]

Absolute time may be determined by a clock that issues (assumedly) regular pulses (transition occurrences of the clock, or clock events). An absolute time value may be assigned to some transition occurrence, by comparing that transition occurrence with the successive (absolute-time-determining) clock events. A strictly ordered set of time points can be defined on the basis of these clock events, called time axis.

Act

See action.

Actand [E15, D20]

An **actand** is a thing involved in the pre-state or post-state of an action, not considered as an actor for that action.

Action [E14, D17]

An **action** is a transition involving a non-empty set of actors in its pre-state, and, if not "destroyed" or "consumed" by the action, in its post-state as well, and involving a non-empty or empty set of other things (actands) as part of its pre-state, and having a non-empty or empty set of other things (actands) in its post-state.

Action context [E16, P5, D22]

The **action context** of an action is a special, optional part of the pre-state of that action, qualifying the context or situation in which that action is performed, and determining or modifying at least one of its output actands.

Action occurrence [E14]

An **action occurrence** is a transition occurrence with the same conditions as applying for the notion of action.

Active system [E31, D43]

An active system is conceived as capable of doing something, i.e. some of the system components are actors performing actions on actands.

Activity

See action.

Actor [E13, P4, D19]

An **actor** is a special thing conceived as being "responsible" or "responsive" and as being able to "cause" transitions, and is therefore part of their pre-states, and, if not "destroyed" or "consumed" by the transitions, also part of their post-states.

Aggregate

See composite thing.

Alphabet [E22, D28]

An **alphabet** of a language is a non-empty and finite set of symbols.

Area

See domain.

Category

See type.

Closed System [E31, D44]

¹The following conventions are used in the glossary:

Glossary entry in bold:	A defined concept
Glossary entry not in bold:	Synonym of a defined concept, or a notion related to a defined concept in some other way
Term in bold:	A concept defined in this glossary entry
<u>Term underlined:</u>	A concept defined in another glossary entry

The numbering [En] refers to the explanatory definitions in chapter 3, and the numbering [Pn], [Dn] or [Fn] refers to the primitives, definitions or functions in chapter 4, respectively.

A system is called a **closed system** if the system environment cannot cause any transitions within the system.

Co-action [E14]

A **co-action** is a special action performed by more than one actor in a co-ordinated way, pursuing a common goal.

Co-ordinated action

See co-action.

Collection of things

See composite thing.

Communication [E37, D37]

Communication is an exchange of messages, i.e. a sequence of mutual and alternating message transfers between at least two human actors, called communication partners, whereby these messages represent some knowledge and are expressed in languages understood by all communication partners, and whereby some amount of knowledge about the domain of communication and about the action context and the goal of the communication is made present in all communication partners.

Composite action [E14, D18]

A **composite action** is a composite transition with the same conditions as applying for the notion of action.

Composite thing [E4, D7]

A **composite thing** is a thing, not being an elementary thing.

Composite transition [E9, D11]

A **composite transition** is a state-transition structure with a unique pre-state and a unique post-state.

Computerised information sub-system (CISS) [E41, D49]

A **computerised information sub-system** is a sub-system of an information system, whereby all actions within that sub-system are performed by one or several computer(s).

Conceiver [E20, D26]

A **conceiver** is a human actor involved in a conceiving action.

Conceiving action [E20, D26]

A **conceiving action** is a special action of a human actor having a perception and possibly some action context as input actand(s) and a conception as output actand.

Conceiving context [E20, D26]

A **conceiving context** is the action context of a conceiving action.

Conception [E20, P10, D26]

A **conception** is a special actand resulting from an action whereby a human actor aims at interpreting a perception in his mind, possibly in a specific action context.

Data [E34, D34]

The term **data** denotes any set of representations of knowledge, expressed in a language.

Domain [E18, P8, D25]

A **domain** comprises any "part" or "aspect" of the "world" under consideration.

Domain component [E18]

A **domain component** is any "part" or "aspect" of that domain.

Domain element

See domain component.

Domain environment [E18]

A **domain environment** is the "world" without that domain.

Dynamic system [E31, D42]

A **dynamic system** is conceived as capable of undergoing change, i.e. some of the system components are transitions.

Elementary sign

See label.

Elementary thing [E4, D5]

An **elementary thing** is a thing, not being a relationship and not being characterised by the special predicator called 'has-element'.

Entity [E5, D6]

An **entity** is a predicated thing as well as an elementary thing.

Event

See transition occurrence.

Extensional model [E28]

An **extensional model** is that part of a model containing a specific population of the types in the corresponding intensional model, whereby this population must obey all rules determined in that intensional model.

Field

See domain.

Goal [E17, P6, D23]

The **goal** of an action is a special input actand of that action, pursued by the actors of that action and stating the desired output state intensionally.

Goal-pursuing actor [E17]

A **goal-pursuing actor** is an actor performing an action, who deliberately aims at a specific goal when involved in that action.

Human actor [E19, P7, D24]

A **human actor** is a responsible actor with the capabilities and liabilities of a normal human being, in particular capable of performing perceiving actions, conceiving actions and representing actions.

Information [E36, D36]

Information is the personal knowledge increment brought about by a receiving action in a message transfer, i.e. it is the difference between the conceptions interpreted from a received message and the personal knowledge before the receiving action.

Information system [E40, D47]

An **information system** is a sub-system of an organisational system, comprising the conception of how the communication- and information-oriented aspects of an organisation are composed (e.g. of specific communicating, information-providing and/or information-seeking actors, and of specific information-oriented actands) and how these operate, thus describing the (explicit and/or implicit) communication-oriented and information-providing actions and arrangements existing within that organisation.

Information system denotation [E40]

An **information system denotation** is a precise and unambiguous representation of an information system.

Information system in the broad sense

See information system.

Information system in the narrow sense

See computerised information sub-system.

Input actand [E15, F8]

An **input actand** is a part of the pre-state of an action, excluding the actors.

Instance [E6, D15]

An **instance** of a type of things is an element of a population of that type.

Intensional model [E28]

An **intensional model** is that part of a model comprising the possibilities and necessities of a domain only, i.e. the types and rules.

Interpreter [E21, D27]

An **interpreter** is a human actor performing an interpreting action.

Interpreting action [E21, D27]

An **interpreting action** is the sequence of a perceiving action performed on a domain, resulting in a perception of that domain, followed by a conceiving action performed on that perception, resulting in a conception.

Interpreting context [E21, D27]

An **interpreting context** is the action context of an interpreting action.

Interpreting process

See interpreting action.

Knowledge [E33, F11, D33]

Knowledge is a relatively stable and sufficiently consistent set of conceptions possessed by single human actors.

Label [E24, D29]

A **label** is a special entity being an elementary representation and used for referring to some conception in an elementary way.

Language [E22, D28]

A **language** is a non-empty set of permissible symbolic constructs. The permissible symbolic constructs in a language are determined either extensionally by enumeration or intensionally by a set of rules. The rules of a language may be syntactic ("grammar") as well as semantic ("semantic rules").

Law

See rule.

Message [E35, D35]

A **message** is data, transmitted by one actor (the sender) via a channel (a medium), and intended for a non-empty set of other actors (the receivers).

Message transfer [E35, P14, D35]

A **message transfer** is a sequence of actions, the sending action by the sender and the receiving actions by the receivers, whereby the input actand of the sending action is the message to be sent, whereby the output actand of the sending action, being equal to the input actand of the receiving action, is the message on the channel, and whereby the output actand of the receiving action is the message received.

Meta-model [E29]

A **meta-model** is a model of the conceptual foundation of a language, consisting of a set of basic concepts, and a set of rules determining the set of possible models denotable in that language.

Model [E26, P13, D30]

A **model** is a purposely abstracted, clear, precise and unambiguous conception.

Model denotation [E26, D31]

A **model denotation** is a precise and unambiguous representation of a model, in some appropriate formal or semi-formal language.

Modeller [E27, D32]

A **modeller** is a human actor performing a modelling action.

Modelling action [E27, D32]

A **modelling action** is the sequence of a perceiving action performed on some domain, followed by a conceiving action on that perception, resulting in a model, and followed by a representing action on that model, resulting in a model denotation.

Modelling process

See modelling action.

Name

See label.

Norm [E39]

Norms are socially agreed rules affecting and to a large extent directing the actions within an organisational system.

Object

See thing.

Object of action

See actand.

Open system [E31, D44]

An **open system** is conceived as one which may respond to external messages or triggers, i.e. there may be transitions within the system due external causes coming from the system environment.

Organisational system [E39, D45]

An **organisational system** is a special kind of system, being normally dynamic, active and open, and comprising the conception of how an organisation is composed (i.e. of specific actors and actands) and how it operates (i.e. performing specific actions in pursuit of organisational goals, guided by organisational rules and informed by internal and external communication), where its systemic properties are that it responds to (certain kinds of) changes caused by the system environment and, itself, causes (certain kinds of) changes in the system environment.

Output actand [E15, F9]

An **output actand** is a part of the post-state of an action, excluding the actors.

Passive System [E31, D43]

A system is called a **passive system** if it is not active.

Perceiver [E19, D25]

A **perceiver** is a human actor involved in a perceiving action.

Perceiving action [E19, D25]

A **perceiving action** is a special action of a human actor having a domain as input actand and a perception as output actand.

Perception [E19, P9, D25]

A **perception** is a special actand resulting from an action whereby a human actor observes a domain with his senses, and forms a specific (static, non-time-varying, or dynamic, time-varying) pattern of visual, auditory or other sensations of it in his mind.

Personal knowledge

see knowledge.

Population [E6, D15]

A **population** of a type of things is a set of things, each one fulfilling the characterisation determining that type.

Post-state [E7, F6]

The **post-state** of a transition is the state valid after that transition, and is characterised by the special predicator 'after'.

Pre-state [E7, F5]

The **pre-state** of a transition is the state valid before that transition, and is characterised by the special predicator 'before'.

Predicated thing [E2, D3]

A **predicated thing** is a thing being characterised or qualified by at least one predicator.

Predicator [E2, D2]

A **predicator** is a thing, used to characterise or qualify other things, and assumed as being "atomic", "undividable" or "elementary".

Process

See state-transition structure.

Qualifier

See predicator.

Receiver [E35, D35]

A **receiver** is an actor receiving a message.

Reference [E24, P12, D29]

A **reference** is a special binary relationship between a conception and a representation used to describe that conception.

Relationship [E3, D1]

A **relationship** is a special thing composed of one or several predicated thing(s), each one associated with one predicator characterising the role of that predicated thing within that relationship.

Relative time [E11, D13]

The strict partial order imposed on the sets of all transition occurrences is called **relative time**.

Representamen

See representation.

Representation [E23, P11, D28]

A **representation** is a special actand describing some conception(s) in a language, resulting from an action whereby a human actor aims at describing his conception(s), possibly in a specific action context.

Representer [E23, D29]

A **representer** is a human actor involved in a representing action.

Representing action [E23, D29]

A **representing action** is a special action of a human actor having a conception and possibly some action context as input actand(s) and a representation as output actand.

Representing context [E23, D29]

A **representing context** is the action context of a representing action.

Resource [E15, D21]

The pre-state of an action, i.e. the union of the set of actors and the set of input actands of that action, is called its **resources**.

Role

See predicator.

Rule [E12, D16]

A **rule** determines a set of permissible states and transitions in a specific context. In other terms, a rule governs a non-empty set of types of things by determining their permissible populations.

Semiotic level [E25]

The **semiotic level** of a representation is the aspect considered in representing it. The semiotic levels are: physical, empirical, syntactical, semantical, pragmatic, and social.

Sender [E35, D35]

A **sender** is an actor sending a message.

Set membership [E4, P2, D4]

A **set membership** is a special binary relationship between a thing (the set), characterised by the special predicator called 'has-element', and another thing, characterised by the special predicator called 'is-element-of'.

Set of things

See composite thing.

Shared knowledge [E38, D38]

Shared knowledge is that knowledge of the individuals in a group of human actors, which they assume to be identical (or at least similar) to that of the others, as resulting from the negotiation process implicit in some communication.

Sign

See representation.

Sign carrier

See representation.

Sign relationship

See representation, reference.

Sign token

See representation.

Signifiant

See representation.

State [E7, D9]

A **state** is a composite thing, involved as pre-state or as post-state in some transition. No element of a state may be a transition, itself.

State-transition structure [E8, D10]

Given are the transitions $t_x: s_1 \Rightarrow s_2$ and $t_y: s_3 \Rightarrow s_4$. The following basic **state-transition structures** exist in this case:

(1) Sequence:

sequ (t_x, t_y) is a sequence of transitions if s_3 is a subset of s_2 .

The resulting state-transition structure has s_1 as pre-state and s_4 as post-state.

Longer sequences are defined as follows:

sequ (t_x, t_y, t_z) follows from sequ (t_x, t_y) and sequ (t_y, t_z)

(2) Choice:

choice (t_x, t_y) is a choice of transitions if the intersection of s_1 and s_3 is not empty.

The result is either transition t_x or t_y , but not both.

(3) Concurrency:

concur (t_x, t_y) are concurrent transitions if the intersection of s_1 and s_3 is empty.

The result is $(s_1 \cup s_3) \Rightarrow (s_2 \cup s_4)$.

Static system [E31, D42]

A system that is not dynamic is called a **static system**.

Sub-system [E32, D41]

Any **sub-system** S' of a larger system S is a system, itself. The set of all sub-system components of S' is a proper subset of the set of all system components of S .

Symbol [E22, D28]

A **symbol** is a special entity used as an undividable element of a representation in a language.

Symbolic construct [E22, D28]

A **symbolic construct** is a non-empty and finite "arrangement" of symbols taken from an alphabet. In the one-dimensional case, an arrangement is just a sequence of symbols (a "sentence"). In the n-dimensional case ($n > 1$), it may be any arrangement of its constituting symbols in the n-dimensional space. Provided one considers the elements of arrangement (such as sequence) belonging to the alphabet, a symbolic construct is a non-empty and finite set of symbols.

Symbolic form

See symbolic construct.

System [E30, D40]

A **system** is a special model, whereby all the things contained in that model (all the system components) are transitively coherent, i.e. all of them are directly or indirectly related to each other form a coherent whole. A system is conceived as having assigned to it, as a whole, a specific characterisation (the so-called "systemic properties") which cannot be attributed to any of its system components.

System component [E30]

A **system component** is a non-empty set of things being contained in that system.

System denotation [E30]

A **system denotation** is a precise and unambiguous representation of a system.

System element

See system component.

System environment [E30]

The **system environment** of a system is the set of all things not being contained in that system.

System representer [E30]

A **system representer** is a human actor representing a system in some language.

System viewer [E30]

A **system viewer** is a human actor perceiving and conceiving a domain as a system. A system viewer recognises the system, by its distinction from the system environment, by its coherence, and because of its systemic properties.

Thing [E1, P1]

A **thing** is any part of a conception of a domain (being itself a "part" or "aspect" of the "world"). The set of all things under consideration is the conception of that domain.

Transition [E7, P3, D8]

A **transition** is a special binary relationship between two (partially or totally) different composite things, called the pre-state and the post-state of that transition, whereby at least one thing is element of the pre-state, but not of the post-state, or vice versa.

Transition occurrence [E10, D12]

A **transition occurrence** is a specific occurrence of a transition. A set of transition occurrences is subject to strict partial ordering.

Type [E6, D15]

A **type** of things is a specific characterisation (e.g. a predicate) applying to all things of that type.

Universe of discourse

See domain.

The rest is silence.

William Shakespeare, Hamlet

INDEX

The following conventions are used in the index:

- A Term, first letter capital: Concept informally and/or formally defined
- A term in small letters: Concept mentioned or discussed only informally
- A **page number in bold**: Concept informally explained or defined there
- Page number, plain layout: Concept mentioned or discussed there
- *Page number, bold italics*: Concept formally introduced or defined there

- Absolute time**, **40**, **99**, 129
act, 5, 41
Actand, 18, 30, 33, 41, **42**, 45, 47, 48, 64, 71, 73, 77, **102**, 103, 110, 111, **115**, 124, 130
Action, 5, 15, 16, 18, 28, 30, **41**, 45, 47, 48, 52, 64, 66, **101**, 103, 110, **115**, 124, 129, 152
Action context, **42**, 47, 48, **102**, **115**, 130
Action occurrence, **41**, 130
Active system, **62**, **109**, **119**, 134
activity, 5, 41
Actor, 18, 30, **41**, 68, **101**, 103, 110, **115**, 124, 129, 152
adaptive information sub-system, 79
addressee-specific communication, 78
addressee-unspecific communication, 78
aggregate, 36
Alphabet, **47**, **104**, **117**, 131
arbitrariness, 158
area, 46
autonomous behaviour, 80
autonomous system, 151
behaviour, 19
behaviour modelling, 22
bit, 9
building processor, 77
byte, 9
category, 37
channel, 66, 67
channelling, 67
Closed system, **110**, **119**, 134
Closed System, **62**
Co-action, **41**, 70, 130, 152
cognitive science, 6, 12
Coherence, **109**, **119**, 133
collection, 36
Communication, 5, 9, 14, 15, 20, 21, 28, 30, 45, 53, 64, **70**, **108**, 135, 140, 142, 143, 144, 145, 152
communication partner, 70
Composite action, **41**, **101**, **115**, 130
Composite thing, **36**, 37, 38, **96**, **113**, 128
Composite transition, **39**, **98**, **114**, 129
computer science, 12
Computerised actor, **111**, **120**, 136
Computerised information sub-system, **73**, **111**, **120**, 136
Conceiver, **47**, **104**, **116**, 131
Conceiving, 131
Conceiving action, **47**, **104**, **116**
Conceiving context, **47**, **104**, **116**, 131
Conception, 21, 22, 23, 30, 33, 34, 44, **47**, 48, 49, 51, 53, 66, **104**, **116**, 131
conceptual model, 5
conceptual system, 5
constructivism, 29
constructivist, 8, 11, 26, 31, 33, 44, 45, 46, 47, 52, 53, 56, 65, 71, 72
co-ordinated action, 14, 15, 20, 41
Data, 5, 21, **66**, **107**, **118**, 134
data modelling, 22
database, 14, 20, 77
database system, 11, 154
decision, 19, 20, 21
decision process, 15
decision support system, 7, 154
deep structure, 32
degeneration, 161
derivation, 81
development of information systems, 16
dis-information, 68
document retrieval system, 11
Domain, 29, 34, **46**, 51, 61, **103**, **116**, 130, 147
Domain component, **46**
domain element, 46
Domain environment, **46**
Dynamic system, **62**, 134
Dynamic System, **109**, **119**
effect, 20
elementary sign token, 49
Elementary thing, **36**, **95**, **113**, 128
empirical layer, 9, 15, 140
Entity, 5, **36**, 47, 49, **96**, **113**, 124, 128
epistemology, 32
event, 5, 39
evolution of first order, 82
evolution of second order, 82
expert system, 7, 154
expressiveness, 158
Extensional model, **57**, 82, 133
false information, 68
field, 46
first-order-logic language, 92
Goal, 18, 30, **43**, **102**, **116**, 130
Goal-pursuing actor, **43**, 130
government information system, 7
grammar, 23, 48
granularity, 36, 39, 42, 70, 95
homonym, 49
Human actor, **46**, 47, 48, 52, 53, 56, 61, 66, 70, **103**, **116**, 130

- Human being, 29
- Information, 5, 21, 22, 26, 28, 64, **68**, **107**, **119**, 135, 140, 143, 144, 145
- Information system, 5, 26, **73**, **110**, **120**, 136
- Information system denotation**, **73**, 136
- information system in the broader sense, 73
- information system in the narrower sense, 73
- informing, 68
- Input actand**, **42**, 130
- input handling information sub-system, 79
- input processor*, 77
- Instance**, **37**, **99**, **114**, 128
- Intensional model**, **57**, 82, 133
- intention**, 20
- interest group, 7
- internal mental inspections*, 32
- interpretant, 51, 139
- interpretation conflict, 30
- Interpreter**, **47**, **104**, **116**, 131
- Interpreting action, **47**, 52, **104**, **116**, 131
- Interpreting context**, **47**, **104**, **116**, 131
- interpreting process, 47
- inter-subjective reality, 8, 26, 32, 45, 71
- Knowledge, 5, 9, 22, 28, 30, 45, **66**, 70, **106**, **118**, 134, 145, 152
- Knowledge/data-processing action**, **110**, **120**, 135
- knowledge-based system, 154
- Label**, **49**, 51, **105**, **117**, 132
- Language, 23, 31, **47**, 48, 70, 82, **104**, **117**, 131, 158, 162
- law, 40
- layered approach, 32
- library system, 7
- linguistics, 6
- maintenance-oriented information sub-system, 79
- management information system, 7
- managerial information sub-system, 79
- meaning**, 22, 142, 143, 145
- mechanist*, 8
- medical expert system, 7
- Message, 5, 9, 28, **66**, **107**, **118**, 134
- Message transfer**, **66**, **107**, **118**, 134
- Meta-model, 11, **58**, 133, 154, 155
- Meta-model transformation, 157, 158, 159, 162, 164
- metaphysical position, 8
- model, 154
- Model, 21, 22, 23, **55**, 60, **105**, **117**, 133, 145, 153
- Model denotation, 23, **55**, **106**, **117**, 133, 145
- Modeller**, **56**, **106**, **118**, 133
- Modelling action**, **56**, **106**, **118**, 133
- modelling approach, 11, 154
- modelling concept, 11
- modelling process, 56
- naïve realist*, 8, 24, 26, 71
- name, 49, 51
- network, 14
- noëma, 51
- noise**, 24
- nominalist*, 8
- Norm, 18, 19, **71**, 135
- object, 5, 34, 51
- object of action, 42
- object orientation, 14
- objectivist*, 8, 26, 52, 53
- object-oriented method, 21
- observation**, 20
- office information system, 7
- onoma, 51
- ontological view, 11, 12, 32, 33, 58, 92, 154, 156
- ontology, 6, 32, 137, 153
- Open system**, **62**, **110**, **119**, 134
- ordering and transformation scheme*, 11, 158, 159
- organisation, 5, 14, 16, 18, 64
- organisation science, 6, 12
- organisational culture, 19
- Organisational system**, **71**, **110**, **119**, 135
- Output actand**, **42**, 130
- output handling information sub-system, 79
- Passive system**, **109**, **119**, 134
- Passive System**, **62**
- pattern*, 23, 24
- Perceiver**, **46**, **103**, **116**, 131
- Perceiving action**, **46**, 47, **103**, **116**, 131
- Perception, 29, **46**, **103**, **116**, 130
- permanently stored message**, 78
- philosophical position, 8, 11, 12, 18, 26, 27, 52
- physical layer, 9, 15, 140
- physical token**, 24
- planning system, 11
- Population**, **37**, **99**, **114**, 128
- Post-state**, **38**, 129
- pragma, 51
- pragmatic layer, 9, 15, 143
- Predicated thing**, **35**, 36, 37, **94**, **112**, 128
- Predicator**, **35**, **94**, **112**, 127
- preferred synonym, 50
- Pre-state**, **38**, 129
- probabilistic behaviour**, 80
- process, 5, 38
- process modelling, 22
- production-oriented information sub-system, 79
- professional language, 26
- protocol**, 20
- public administration system, 7
- qualifier, 35
- reactive behaviour, 80
- reactive system, 151
- Receiver**, **66**, **107**, **118**, 135
- Receiving action**, **107**, **118**, 134
- redundancy**, 24
- Reference, 23, **49**, **105**, **117**, 132
- Relationship**, **35**, 36, 38, 49, **94**, **112**, 124, 128
- Relative time**, **40**, **99**, 129
- representamen, 48, 51, 139
- Representation, 21, 23, 28, 31, 44, 47, **48**, 49, 66, **104**, **117**, 132
- Representer**, **48**, **105**, **117**, 132
- Representing action**, **48**, 52, **105**, **117**, 132

- Representing context**, 48, 105, 117, 132
- Resource, 18, 42, 102, 115, 130
- responsibility**, 19
- responsible actor, 28, 46
- responsive actor, 28
- responsive system, 151
- restriction, 161
- retrieval processor*, 77
- role, 35
- Rule, 18, 19, 31, 40, 100, 114, 129
- rule-based system, 11
- semantic rule, 48
- semantical layer, 9, 15, 142
- Semiotic level**, 54, 132, 139
- semiotic tetrahedron, 51, 62
- semiotics, 6, 12, 137
- Sender**, 66, 107, 118, 134
- Sending action**, 107, 118, 134
- sending information, 68
- sentence**, 23, 48
- set, 36
- Set membership**, 36, 95, 112, 128
- shared conception*, 32
- Shared knowledge**, 71, 108, 135
- sign, 48, 50, 51, 139
- sign carrier, 48
- sign object, 51, 52, 139
- sign relationship, 49
- sign token, 48
- signal, 28
- signifiant, 48, 139
- signifié, 139
- social layer, 9, 15, 144
- social system, 5, 18, 19
- sociology, 6
- specialisation, 160
- speech act**, 20, 21, 70, 144
- State, 18, 19, 30, 38, 97, 113, 128
- State-transition structure**, 38, 98, 114, 129
- Static system**, 62, 109, 119, 134
- subjectivist*, 8
- Sub-system**, 63, 109, 119, 134, 152
- suitability, 159
- Symbol, 31, 47, 104, 117, 131
- Symbolic construct**, 47, 104, 117, 131
- symbolic form, 47
- synonym, 49
- syntactical layer, 9, 15, 141
- System, 5, 18, 28, 60, 109, 119, 133, 146
- System component**, 60, 133
- System denotation**, 60, 133
- System environment**, 60, 133
- system exposition, 148
- system of concepts*, 61
- system of meta-models, 163
- System representer**, 60, 134
- system science, 6, 12, 137
- system type, 148
- System viewer**, 60, 133, 147
- systemeering, 149
- systemic property, 60, 147
- technical system, 5
- Thing, 26, 30, 33, 34, 94, 112, 124, 127
- time-oriented system, 11
- transient message, 78
- Transition, 5, 19, 30, 38, 41, 97, 113, 129
- Transition occurrence**, 39, 98, 114, 129
- Type**, 37, 99, 114, 128
- universe of discourse, 5, 46
- value**, 19
- value addition, 14, 15, 16, 18
- value of information, 16
- variety**, 24
- world, 29