“Applied and Computational Category Theory (ACCAT)”

Prof.Dr.Jochen PFALZGRAF
Fachbereich Computerwissenschaften
Universität Salzburg
jochen.pfalzgraf@sbg.ac.at

TUTORIAL (Preliminary Version)

www.cosy.sbg.ac.at/∼jpfalz/ACCAT-Tutorial-KI2004.html

Originally Presented at

Workshop WS12 at KI 2004

27th German Conference
on Artificial Intelligence - KI 2004
and INFORMATIK 2004

September 20 - 24 (2004), Univ. of Ulm, Germany

1 REMARKS on the Workshop "ACCAT" (2004) "Applied and Computational Category Theory"

Subsequently, the text of the announcement of the ACCAT Workshop at KI-2004 on the homepage of the author (`www.cosy.sbg.ac.at/~jpfalz/ACCAT-Tutorial-KI2004.html`) is presented as brief introductory information.

Important NOTE: The KI-2004 ACCAT Workshop was the starting point of later ACCAT Workshops that I organized jointly with my colleague Hartmut Ehrig and his group (TU Berlin) as Satellite Events of the big European Joint Conferences on Theory and Practice of Software, **ETAPS**. The 1st ACCAT Workshop took place at ETAPS-2006 in Vienna, Austria, followed by the 2nd ACCAT Workshop at ETAPS-2007 in Braga, Portugal, and the 3rd ACCAT Workshop at ETAPS-2008 in Budapest, Hungary. The 4th ACCAT Workshop **ACCAT'09** is scheduled as satellite event at **ETAPS-2009**, York, England.
For further information on ACCAT'09 at ETAPS-2009, cf.
`http://www.cosy.sbg.ac.at/ jpfalz/ETAPS-2009.html`.
For a report on the origins of ACCAT, cf.
`http://www.cosy.sbg.ac.at/ jpfalz/ACCATabstract-2006.pdf`.
We also refer to the remarks in `http://www.cosy.sbg.ac.at/ jpfalz/ACCAT.html`.

NOTA: The sections following Section 2 represent an Extended Version of the original tutorial of 2004.

NOTE: CAT abbreviates Category (Theory)

Nachfolgend ist die Veranstaltung beschrieben in Form eines Tutorials. Die einzelnen Punkte sind zur thematischen Orientierung des Workshops gedacht, aber nicht bindend. Der Workshop soll offen sein. Der Umfang des Tutorials wird den lokalen Gegebenheiten flexibel angepasst.
The following material corresponds to a short version (summary, compact course) of an extended tutorial, based on a university course given by the author several times.
This ACCAT Tutorial is the basis for future extended versions (work in progress).
Supplementary text materials can be obtained by request.

OBJECTIVE: To offer a compact introduction to elementary notions in category theory. CAT is a powerful mathematical modeling language (containing universal construction principles) with a wide area of potential applications, especially in computer science and AI. The workshop will discuss aspects from theory and practice (applications). Depending on participants and submissions, the meeting can offer a brief tutorial on ACCAT by the organiser.

MOTIVATION: Introductory and motivating remarks: CAT provides a unifying formal (mathematical) language with constructive elements, of increasing importance in computer science and AI. The concept of a CAT, by its very nature, contains OO aspects. There are obvious trends in interdisciplinary modeling with CAT, algebra, geometry, topology, logic, as can be observed in activities and events, taking place in large areas like Neurobiology, Cognitive Sciences, Brain Research, Topological Psychology, Music Theory. (Further motivating remarks and information: cf. later sections).

CONTENTS (of an Extended Tutorial - a Guideline)

- Introduction to basic notions and notation of category theory, examples.
- Implementational aspects (in ML syntax). Categorical description of LIST handling.
- Categorical modeling of general relational structures (with many examples); towards a general semantics for relations. Arrow diagrams interpreted categorically; in particular, this aspect applies to UML diagrams.
- Example: Categorical treatment of forming the greatest common divisor and least common multiple as (co-)limit constructions (the universal property).
- Remarks on the textbook Bird/de Moor: Algebra of Programming (published by Prentice Hall, 1997).
- Towards a generic CAT based model for multiagent systems (MAS).
- On the Base Diagram of a MAS
- Report on doctoral theses dealing with CATs to solve open problems of relevance in computer science (former PhD students in my ACCAT group at RISC-Linz).
- ACCAT and Theoretical Neurobiology
- Remarks on a CAT based structure modeling in connectionist networks and towards a theory of "homomorphic learning". An industrially relevant application of CAT based modeling in simulation of neural networks (hints to the Salzburg "FlexSimTool").
- New trends in interdisciplinary modeling with CAT, Algebra, Geometry, Topology, Logic, e.g. in Neurobiology, Cognition, Brain Research, Topological Psychology, Music Theory.

Some LITERATURE Hints

Renowned classical textbook by one of the inventors of CAT theory:

- Saunders Mac Lane: Categories for the Working Mathematician (2nd ed.). (Springer Graduate Texts in Mathematics, Vol.5, 1998)
- Lawvere/Schanuel: Conceptual Mathematics. A first introduction to categories. (Cambridge University Press, 2000)
- Rydeheard/Burstall: Computational Category Theory. (Prentice Hall, 1988)
- Adamek/Herrlich/Strecker: Abstract and Concrete Categories. (J.Wiley Interscience, 1990)
- Goldblatt: TOPOI. The categorical analysis of logic. (North-Holland, 1986)
- Mazzola: The Topos of Music. (Birkhäuser Verlag, 2002)
- Pierce: Basic Category Theory for Computer Scientists. (The MIT Press, 1991)
- Pfalzgraf: ACCAT Tutorial, held at TU Munich, Sept. 3-5, 1997
- Pfalzgraf: Modeling connectionist network structures: some geometric and categorical aspects. (Annals of Math. and AI, 36 (2002), pp.279-301)

- Pfalzgraf: Modeling Connectionist Networks: Categorical, Geometric Aspects (Towards "Homomorphic Learning").
Presentation at CASYS'2003, Liège, Belgium, Aug.2003. Received a Best Paper Award.
In: CASYS'2003, American Institute of Physics, AIP Proceedings, Vol. 718 (2004), D.M.Dubois (Ed.)

NOTA: ACCAT Lecture Notes (by J.Pfalzgraf), worked out with details, exist as materials for a two semester course.

2 Introductory Remarks and Motivations

Kategorien und gefaserte Strukturen [Faserbündel (fiber bundles), Garben (sheaves)] sind schon seit vielen Jahren etabliert als sehr ausdrucksstarke Modellierungssprachen innerhalb der Mathematik und haben tiefliegende Anwendungen in der (mathematischen) Physik und der Systemtheorie. Kategorientheorie gewinnt auch zunehmende Bedeutung in der Informatik. Wir werden anschliessend dazu einige Themen ansprechen.

Eine Bemerkung am Rande: Insider der Kategorientheorie kürzen das Wort *Categories* oft liebevoll ab durch *CATs*, also CAT ist das Kürzel für Category (Theory) – wir werden dies auch verwenden.

Wir werden sehen, dass die Pfeil-Notation für Abbildungen (Funktionen), also z.B.

$$f : A \longrightarrow B \quad \text{bzw.} \quad A \xrightarrow{f} B$$

eine ganz fundamentale Rolle in der Kategorientheorie spielt. Dazu gibt es eine historisch sehr interessante Anmerkung von einem der Begünder der Kategorientheorie, Saunders Mac Lane, der sinngemäß sagte: The invention of a new notion, the *arrow*, formed the basis for the development of a new theory, *categories*. Dies ist besonders zu sehen vor dem Hintergrund der Tatsache, dass die Pfeil-Notation noch gar nicht so lange existiert. Sie ist erfunden worden innerhalb des Gebietes der Topologie (möglicherweise von Hurewicz) in den 1940er Jahren und ersetzte die früher übliche mengentheoretische Notation für eine Abbildung: $f(A) \subset B$. Die Pfeil-Notation $f : A \longrightarrow B$ ist dagegen ein erheblicher Fortschritt (auch wenn man dies auf den ersten Blick nicht ersehen mag), denn sie ermöglicht es, mit den Abbildungspfeilen einen grundlegenden "Calculus of Arrows" aufzubauen. Dieser kommt tiefgehend zum Tragen in Disziplinen wie Homologische Algebra, Algebraische Topologie, Algebraische Geometrie, K-Theorie und natürlich in ganz elementarer Weise in der Kategorientheorie. Wir werden dies ausführlich kennenlernen und sehen, wenn wir mit sog. kommutativen Diagrammen und Sequenzen von Morphismen, etc., zu tun haben. Man spricht im Rahmen der CATs oft vom sog. "diagram chasing" und auch von "abstract nonsense". Lassen wir es auf uns zukommen und einwirken. Es ist berechtigt zu sagen: Kategorientheorie beinhaltet ganz wesentlich "arrow handling", "arrow calculus".

Axel Poigné schreibt im *Handbook of Logic in Computer Science, Vol.1*, über Kategorientheorie:

"If asked for a single reason for the attention that category theory, at least as a language, enjoys in some areas of computer science, I would guess that its attraction stems from being a foundational theory of functions which provides a sound basis for (functional) programming and programming logic. If asked for more reasons I would recollect the familiar arguments namely that category theory

- formalizes otherwise vague concepts,
- provides a language that brings to the surface common basic concepts in ostensibly unrelated areas,
- allows us to translate problems from one area to another where a solution may be more easily achieved, or more specifically with regard to computer science, category theory
- allows easier access to various areas of mathematics in that it provides a core of properties to be looked for,
- offers a rich language in which to axiomatize, differentiate and compare structures in computer science and mathematics."

2.1 Selected REFERENCES

Adámek, J., Trnková, V.:
Automata and Algebras in Categories. Kluwer Academic Publishers (1990).

Adámek, J., Herrlich, H., Strecker, G.E.:
Abstract and Concrete Categories. John Wiley & Sons (1990).

Barr, Wells:
Categories for Computing Sciences. Prentice Hall (1990).

Geiger, Pfalzgraf:
Quality Control Connectionist Networks Supported by a Mathematical Model.
Proceedings of EANN'95, Helsinki (1995).

Goldblatt, R.:
Topoi. The Categorial Analysis of Logic. North-Holland, Elsevier Publishers,
Amsterdam, Studies in Logic and the Foundations of Mathematics **98** (1984).

Gray, J.W., Scedrov, A. (eds.):
Categories in Computer Science and Logic.
Contemporary Mathematics Vol. 92, American Math. Soc. (1989).

Hermann, R.:
Geometric Computing Science: First Steps.
Interdisciplinary Mathematics, Vol.25, Math Sci Press (1991).

Herrlich, H., Strecker, G.E.:
Category Theory: An Introduction. Heldermann Verlag Berlin (1979).

Lawvere, F.W., Schanuel, S.H.:
Conceptual Mathematics. A First Introduction to Categories.
Cambridge University Press, 2000, [and Buffalo Workshop Press (1993)].

Mac Lane, S.:
Categories for the Working Mathematician.

Springer Verlag, Graduate Texts in Mathematics **5** (1998), 2nd edition.

Mac Lane, S., Moerdijk, I.:
Sheaves in Geometry and Logic. Springer Verlag, Universitext (1992).

Pfalzgraf, J.:
On a category of geometric spaces and geometries induced by group actions.
Ukranian Jour.Phys. **43,7** (1998), 847-856.

Pfalzgraf, J.:
A note on modeling connectionist network structures: geometric and categorical aspects. Proceedings AISC'2000, July 17-19, 2000, Madrid. Springer Lecture Notes in Artificial Intelligence, Vol.1930 (2001), pp.184-199. J.A.Campbell, E.Roanes-Lozano (Eds.).

Pfalzgraf, J.:
On mathematical modeling of networks and implementation aspects. (With R.Bernhaupt). Proceedings AISC'2002 (AI and Symbolic Computation), July 1-3 (2002), Marseille, France. Published in: Artificial Intelligence, Automated Reasoning, and Symbolic Computation. Springer Lecture Notes in Artificial Intellligence, LNAI 2385. J.Calmet, B.Benhamou, O.Caprotti, L.Henocque, V.Sorge (Eds.), Springer-Verlag 2002.

Pfalzgraf, J.:
Modeling connectionist network structures: some geometric and categorical aspects.
Annals of Mathematics and Artificial Intelligence **36 (3)** (2002), 279-301 (special issue on "AI and Symbolic Computation", Editors: J.Calmet and E.Roanes-Lozano). Kluwer Academic Publishers.

Pfalzgraf, J.:
On a general notion of a hull. In: *Automated Practical Reasoning, J.Pfalzgraf and D.Wang (eds.). Texts and Monographs in Symbolic Computation*, Springer Verlag Wien–New York (1995).

Pierce, B.C.:
Basic Category Theory for Computer Scientists.
The MIT Press Cambridge, Massachusetts and London, England (1991).

Pitt et al.:
Category theory and Computer Programming. Springer Lecture Notes in Computer Science (LNCS) 240 (1986).

Pümplin, D.:
Elemente der Kategorientheorie.
Spektrum Akademischer Verlag, Hochschultaschenbuch (1999).

Rydeheard, Burstall:
Computational Category Theory. Prentice Hall (1988).

van Oosten, J.:
Basic Category Theory. BRICS Lecture Series LS-95-1 (January 1995), University of Aarhus, Denmark. ISSN 1395-2048, (ftp ftp.brics.aau.dk (cd pub/BRICS)).

Schubert, H.:

Kategorien I, II. Springer Verlag (1970).

For Your Information:

♣ Of course, in the WEB you can find a variety of information about categories and working groups (just type "category theory", for example).

♣ A web address where one can find further information about *categories and semantics* (a group working in the interdisciplinary fields of Computer Linguistics and AI):

`http://www.cl-ki.uni-osnabrueck.de/cl-ki/skripte/`

♣ An electronic journal where you can get publications via the WEB:

Theory and Applications of Categories (TAC)

to subscribe you can send e-mail to `tac@mta.ca`.
Cf. also the WWW server at `http://www.tac.mta.ca/tac/`.

2.2 Introductory Remarks (cf. university course)

In various titles of the preceding literature list we can find close links of categories to *computer science* and *logic*. Let us mention, for example, Pitt and Gray/Scedrov: Both titles contain many very interesting aspects of applications of mathematical results, mainly of category theoretical nature, to computer science. Of special interest are the articles in LNCS 240 about **implementing categorial notions** (e.g. *universal properties*) in Functional Programming ML. The book Rydeheard, Burstall: "Computational Category Theory". Prentice Hall (1988) provides plenty of **implementation examples** of pure categorical notions – it can be considered as a textbook on basic **category theory along with ML implementations**. Later we will say more about the role of categories in the foundations of functional programming.

The notion of a category appears in many disciplines in a natural way. Let us cite Robert Hermann in his book on *geometric computing science* where he points out that category theory has been successfully used in computer science in the *algebraic formulation of the logical foundation* and in the theory of *semantics of computer programs*. Additionally, he mentions links to the theory of *formal languages*. Or let us look at the title of the book by Adámek and Trnková (*automata and categories*). The famous computer scientist and mathematician Dana S.Scott first used lattice theory in computer science in the development of the theory of *denotational semantics*. The further evolution of this work has been in categorical directions – this can be read in R.Hermann's book (cf. pp. 387, 388). Hermann even states that category theory is the appropriate mathematical language and tool to systematize *artificial intelligence* (AI) ! We make an interesting quotation (cf. page 388, loc.cit.) here: "Further, the very 'distributed' nature of AI systems is captured mathematically very well by the notion of a Category, which unifies the theory of mathematical structures (ordered sets, groups, fiber bundles, ...) which have already proved to be useful in the physical and engineering sciences."

I speak about these notions in the courses "Formale Systeme II" and "Wissensbasierte Systeme", respectively. Thus, you have already met special instances of a category before. In fact, the very nature of category theory is to be a *unifying language*, coming from mathematics, where the typical notion of a "structured space" and the "functions" or "mappings" between spaces which preserve these structures are formalized in a general manner. Later, we will meet well known examples of categories. Actually, the language aspect is of central interest. And, as I try to convey in my "Wissensbasierte Systeme Vorlesung", finding the right language is of crucial importance in modeling knowledge based systems.

Later, I mention work of former PhD students of mine in the ACCAT group which I built up during my time at RISC-Linz, University of Linz, where categories (and fiber bundles, sheaves and logical fiberings) were in the center of interest, among others. In 1992, we organized a joint seminar with Dana Scott, who was a professor at RISC then, on sheaves in geometry and logic (cf. the corresponding textbook by Mac Lane and Moerdijk). Dana Scott strongly propagated the *importance of categories for the development of computer science*. Our ACCAT group became associated to the **European TEMPUS project** on "Applications of Categories in Computer Science" and we participated in the **European COST Action 15** on "Many-valued Logics for Computer Science Applications".

Later, when I was contacted by "AUI" (the Austro-Ukrainian Institute for Science and Technology), I heard that our ACCAT group was the only one in the country dealing with category theory and applications. We were invited by a group of mathematical physicists to cooperate with them. They are interested in catgeories in mathematical physics (!) – actually, category theory appears in quantum theory. So, for example, one can read in the article by Joyal and Street (in the Springer Verlag Lecture Notes in Mathematics, Vol.1488 (1990): "Category Theory") on page 414 a remark about the connection between knot theory, Feynman diagrams, category theory, and quantum groups.
As an additional information we mention that the AUI is in close contact with the well established research institute for mathematical physics, "Schrödinger Institute", of the famous physicist Walter Thirring in Vienna.

2.3 Brief Report on ACCAT Theses Work (at RISC-Linz)

Subsequently, a brief report is given on doctoral theses dealing with CATs to solve open problems of relevance in computer science (it refers to work of former PhD students in my ACCAT group at RISC-Linz).

With the help of category theory as a unifying language, Karel Stokkermans, in his dissertation, was able to solve an older problem, raised by Bruno Buchberger. The basic question was whether there is a common formal basis for the notion and method of *Gröbner Bases (Buchberger Algorithm), Resolution, and Critical Pair Completion (Knuth-Bendix).* Category theory helped to find a satisfactory natural answer. His results are internationally published in the Journal of Symbolic Computation (1999).

As you will see in our ACCAT course, a substantial part of technical work will be the handling of diagrams of "arrows". An arrow stands for a morphism, functor, natural transformation, respectively. The so-called "diagram chasing" is an every day task of a CAT worker. In particular, it is a common problem to check whether a given diagram of morphisms (arrows) is "commutative". I proposed this problem to Wolfgang Gehrke: to develop a method to check commutativity of diagrams automatically (with the help of a computer). Of course, one has to restrict the study of this problem to "tractable" cases, since the general problem certainly will be undecidable. He concentrated his studies on very interesting (and for computer science important) classes of categories, namely *monads* and *comonads*. Translating the original question into a corresponding problem of equational rewriting resulted in a decidable solution. The results of his thesis are internationally published.

In the framework of a cooperation with other scientists in Linz and Spain, Josef Schicho learned about problems around the proof of a result called a theorem of Fried and MacRae, where specific algebraic proofs had been given so far. He was able to translate the corresponding notions and problems into the language of category theory and thus was able to find a general solution to the original problems (published in the international journal Archiv der Mathematik (1995)).

Viorica Sofronie-Stokkermans worked very successfully with categorical language and sheaves to obtain general semantical models in the field of systems of cooperating agents (especially applicable to cooperating robots). This work on sheaf semantics, among others, also appeared in international publications.

2.4

2.5 Remarks on ACCAT and Connectionist Networks

At RISC-Linz I established another working group called ANNig (artificial neural network interest group) – what has this to do with ACCAT ? There was an old very fruitful cooperation contact with Dr.Hans Geiger and his coworkers. Since many years, he is very active in ANN research and development and he founded an own business in industrial ANN applications in various areas. An important outcome of this cooperation contact was the design of an own flexible ANN simulation tool "NeuroTools 6.0" which we built on the basis of "NeuroTools" of the Geiger group. It has a considerable improved software concept (this work started at RISC-Linz and continued later at Salzburg). All our former work has been permanently performed in close cooperation contact with Dr.Geiger and his group. It has to be underlined that the NeuroTools simulator has proved to be very successful in industrial applications as well as in academia. "Nonclassical" ANN paradigms of H.Geiger are implemented including time coded ANNs (single spike models, SSM) which are designed following biological information processing principles. In a personal cooperation period with H.Geiger it turned out that the net structures which he is permanently using can be mathematically modeled in the sense of a certain *geometric net theory* which we could introduce (based on so-called *noncommutative geometric spaces* which I contributed based on work done in former years on "pure" mathematics). Actually, this became the interface to category theory in the following sense. To an ANN we can associate a geometric net which is deduced from a suitable geometric space. Geometric spaces form a category, therefore we are inspired to speak about the *category of geometric nets*. But what is then the basic feature of ANN theory, namely "LEARNING"? Well, it turns out that a LEARNING STEP can be interpreted as a corresponding MORPHISM in the category under consideration. Thus, we will speak about the instance of "homomorphic learning". We refer to the corresponding articles in the reference list.

In the course of this cooperation we made first modeling approaches using categorical thinking in the previously indicated way. The exploitation of this mathematical, categorical approach for modeling the ANNs in a concrete simulation (which H.Geiger immediately applied in an industrial project on optical quality control) led to surprising results and effects. The *economy of the mathematical model* became visible in a considerable increase of performance in the whole simulation that even could be measured in terms of concrete reductions of the overall project costs !

2.6 Remarks on CAT Semantics for Relational Structures

As already mentioned, semantical modeling is a major aspect of category theory applications. This concerns logic: it is possible to translate (and interpret) the basic logical operations into the language of categories. For example, the classical logical connective AND (conjunction) corresponds categorically to a "product" and the connective OR (disjunction) corresponds to a "coproduct", respectively. The consequent translation of basic logical operations leads to the notion of a *cartesian closed category*, and in extension to that, to a *topos*. Topos theory plays a fundamental role. In its elementary setting it is the categorical "picture" of classical set theory. Bill Lawvere, in his PhD thesis, showed that category theory can be used as the theoretical foundation of the building of mathematics (he was a PhD student of one of the pioneers of category theory, Saunders Mac Lane). W.Lawvere and Schanuel's book (cf. references) is the most elementary treatment and introduction to CATs that I know. It is written very skillfully. Furthermore, we point here to the interplay between CATs and logic (categorical semantics of logic).
I made own experiences in the field of semantic modeling using the language of category theory. During the EU Project MEDLAR, Dov Gabbay raised a question concerning a *hull operation* in general deductive systems. Translating the corresponding problem into the language of CATs opened the view to a natural way of constructing a hull in terms of a suitable colimit (cf. the article "on a general notion of a hull", in the references). Another application of the notion of (co-)limit was the categorical formulation of the process of calculating the greatest common divisor (gcd), and, similarly, least common multiple (lcm). The typical property of gcd can be "rendered" categorically in terms of the "universal property" of the corresponding limit (cf. the previous citation). In the course of this work we came to the idea of a possible program of future research that we name "towards a categorical semantics of general relational structures". In fact, we shall see that (reflexive, transitive) relations lead to concrete examples of CATs. □

Concluding, we mention that category theory plays a basic role in the mathematical foundations of *functional programming*. We point to the interesting article by Philip Wadler: "Monads for Functional Programming", in Lecture Notes in Computer Science, Vol.925 (Springer Verlag, 1995), "Advanced Functional Programming", J.Jeuring and E.Meijer (Eds.). Subsequently, we quote the abstract of that article:
"The use of monads to structure functional programs is described. Monads provide a convenient framework for simulating effects found in other languages, such as global state, exception handling, output, or nondeterminism. Three case studies are looked at in detail: how monads ease the modification of a simple evaluator; how monads act as the basis of a data type of arrays subject to in-place update; and how monads can be used to build parsers."
In the introduction of that article by Ph.Wadler we can read the following remarks: *"It is doubtful that the structuring methods presented here would have been discovered without the insight afforded by category theory. But once discovered they are easily expressed without any reference to things categorical. ..... The examples will be given in Haskell,*

but .... The languages refered to are Haskell, Miranda, Standard ML, and Scheme."

Furthermore, we mention here that CATs play a role in the foundations of the functional programming language ML.

> It is the common curse of all general
> and abstract theories that they have to
> be far advanced before yielding useful
> results in concrete problems.
>
> HERMANN WEYL

3 Some Basic CAT Notions and Notation

This section provides a brief introduction to elementary notions from category theory (CAT). It is not possible to go into details of theory and practice here. Subsequently, we restrict our interest only to the definitions of **category, functor, natural transformation, adjoint functor pairs, diagrams, (co-)limits**. Again we point out: It is an experience that "categorical modeling" can lead to the effect of a "formal economy" in the development of new notions providing a common "linguistic" basis and formal transparency and clarity.

Concerning **computer implementations of CAT notions and constructions** we mention again (cf. section 2) the important book Rydeheard, Burstall: "Computational Category Theory". Prentice Hall (1988). It provides plenty of **implementation examples** of pure categorical notions and it can be considered as a textbook on basic **category theory along with ML implementations**. In our university course "ACCAT" we discuss implementational aspects and material of the textbook. Once more, we point to the brief remark about the role of categories in the foundations of functional programming made at the end of the previous section.

3.1 Categories, Functors, Natural Transformations, Adjoints

Definition .1 *A Category (Cat)* $\mathbf{A}$ *consists of a class of objects, denoted by* A, B, C, $\ldots \in Obj(\mathbf{A})$ *(the objects of* $\mathbf{A}$*). For each pair of objects* A, B *there is a set of morphisms,* $Mor(A, B)$*, also denoted by* $\mathbf{A}(A, B)$ *(the "arrows" between* A *and* B*).* $\mathbf{A}(A_1, B_1)$ *and* $\mathbf{A}(A_2, B_2)$ *are disjoint unless* $A_1 = A_2, B_1 = B_2$*. (Note that* $Mor(A, B)$ *can be empty). There is a composition operation on morphisms: if* $f : A \to B$ *and* $g : B \to C$ *are morphisms, then there is a morphism* $g \circ f : A \to C$*, the composition of* f *and* g*. In a category the following axioms have to hold.*

(i) *The composition of morphisms is associative, that is* $h \circ (g \circ f) = (h \circ g) \circ f$.

(ii) *For every object* A *there is the identity morphism* id_A *with the properties* $f \circ id_A = f$ *and* $id_B \circ f = f$ *for all* $f : A \to B$.

We briefly emphasize that the **arrow notation** for morphisms is of basic importance. We shall use $f : A \to B$, as well as $A \xrightarrow{f} B$, to denote morphisms. The arrow notation is well suited to illustrate (to "visualize") a broad spectrum of modeling problems in a categorical sense (e.g. everything dealing with relational structures).
We refer to the interesting remarks in Mac Lane, S.: "Categories for the Working Mathematician". Springer Verlag, Graduate Texts in Mathematics 5 (1998), 2nd edition. Notes at the end of Chapter I.
Some typical examples of categories in mathematics are, among others: the category of *groups* (objects are groups, morphisms are the group homomorphisms), the category of *monoids* (objects are monoids, morphisms are monoid homomorphisms), the category of *topological spaces* (objects are topological spaces, morphisms are the continuous maps between spaces), the category of *vector spaces* over a field (objects are the vector spaces, morphisms are the linear maps), and of course the category of *sets* (sets as objects and set mappings as arrows) having all the previous ones as subcategories.

A further example is the notion of the *category of pointed sets*, $\mathbf{SET}_*$. A pointed set (set with base point) X_a is a set X together with a selected "base point" $a \in X$. If X_a, X_b are pointed sets, then a base-point-preserving map is a map $f : X_a \longrightarrow X_b$, s.th. $f(a) = b$. With pointed sets as objects, base-point-preserving maps as morphisms, and ordinary composition of maps we obtain the category $\mathbf{SET}_*$. The notion of a pointed space (space with base point) is basic in algebraic topology (homotopy theory).
As we will see later (next section), general relational structures can be interpreted in categorical terms.

Summarizing, one can say that category theory discusses the basic features of "everyday work" when dealing with spaces in a certain discipline and studying structure preserving functions (the morphisms) between spaces.

Definition .2 *The notion of a functor constitutes a concept of "function" between categories. Let* $\mathbf{X}$ *and* $\mathbf{Y}$ *denote two categories. Then a functor* $F : \mathbf{X} \longrightarrow \mathbf{Y}$ *assigns to every object* $A \in Obj(\mathbf{X})$ *an object* $F(A)$ *in the category* $\mathbf{Y}$ *and to every morphism* $f : A \rightarrow B$ *in* $\mathbf{X}$ *a morphism* $F(f) : F(A) \rightarrow F(B)$ *in* $\mathbf{Y}$ *such that the following holds for morphisms* $f : A \rightarrow B,\ g : B \rightarrow C$ *and* id_A *in* $\mathbf{X}$
(1) $F(g \circ f) = F(g) \circ F(f)$
(2) $F(id_A) = id_{F(A)}$

More specifically, such a functor is called covariant; it is called contravariant, if it reverses arrows and thus reverses the order of the arrows of a composition of morphisms (i.e. $F(g \circ f) = F(f) \circ F(g)$*).*

A very elementary, basic functor is the so-called *forgetful functor*. Let us consider for example the category SET of sets and Group of groups (as previously mentioned), then the forgetful functor assigns to every object G (group) of Group the underlying set G ("forgetting" its original group structure) which is an object of SET, naturally. It is clear how this functor operates on morphisms. We cannot present further material here and refer to the literature.

On the next higher level of abstraction the notion of a *natural transformation* is settled. It is a kind of a function between functors and is defined as follows.

Definition .3 *Let* $F : \mathbf{X} \longrightarrow \mathbf{Y}$ *and* $G : \mathbf{X} \longrightarrow \mathbf{Y}$ *be two functors. A natural transformation* $\alpha : F \longrightarrow G$ *is given by the following data.*
For every object A *in* $\mathbf{X}$ *there is a morphism* $\alpha_A : F(A) \rightarrow G(A)$ *in* $\mathbf{Y}$ *such that for every morphism* $f : A \rightarrow B$ *in* $\mathbf{X}$ *the following diagram (square) is commutative.*

$$\begin{tikzcd} F(A) \arrow[r, "\alpha_A"] \arrow[d, "F(f)"'] & G(A) \arrow[d, "G(f)"] \\ F(B) \arrow[r, "\alpha_B"'] & G(B) \end{tikzcd}$$

Commutativity means (in terms of equations) that the following compositions of morphisms are equal: $G(f) \circ \alpha_A = \alpha_B \circ F(f)$.
The morphisms α_A *are called the components of the natural transformation* α.

A simple example of a natural transformation in the category SET arises naturally in the following way. Let I denote the identity functor on SET and $\Delta : \mathsf{SET} \longrightarrow \mathsf{SET}$ denote the diagonal (cartesian product) functor, assigning to an object (set) A the (cartesian) product set $A \times A$, then there is a natural transformation $\delta : I \to \Delta$, given by components $\delta_A : I(A) = A \longrightarrow A \times A = \Delta(A)$, where $\delta_A(x) = (x, x)$, for every $x \in A$. Again, we cannot present further material here and refer to the literature.

The fourth elementary notion from category theory, *adjoint functor pairs (adjunctions)*, will not be introduced here — it can be found in the literature.

NOTA: It is possible to treat the **handling of LISTs** categorically where the previous notions arise in a natural way (we refer to our ACCAT course notes for the details).

3.2 Some Simple Examples (Exercises)

It is mentioned in a later section that every set X with a reflexive, transitive relation $R \subseteq X \times X$ leads to the category $\mathbf{X}$ with objects the elements $a \in X$ and arrows defined by $a \to b$ iff aRb (i.e. $(a, b) \in R$).

Subsequently, we consider categories $\mathbf{A}, \mathbf{B}$ defined as follows.

(1) $\mathbf{A} = (\mathbb{N}, |)$, where $|$ is the usual symbol for divisibility (i.e. $m \mid n$ iff $\exists k \in \mathbb{N} : n = km$). Thus, for $a, b \in \mathbb{N}$ (objects in $\mathbf{A}$) there is an arrow $a \to b$ iff $a \mid b$.

(2) Let $\mathbf{B} = (\mathbb{Z}, \leq)$. For $x, y \in \mathbb{Z}$ there is an arrow $x \Rightarrow y$ iff $x \leq y$.
NOTE: In general it holds: if $a \mid b$, then $a \leq b$. Therefore, if $a \to b$ is an arrow in $\mathbf{A}$, then $a \Rightarrow b$ is an arrow in $\mathbf{B}$.

We define two *covariant functors* $F, G : \mathbf{A} \longrightarrow \mathbf{B}$ in the following way.

(3) $F : A \longrightarrow B$, for $a \in \mathrm{obj}A$, we define $F(a) = a \in \mathrm{obj}B$.
It is easy to verify that F is a covariant functor.

(4) $G : A \longrightarrow B$, defined by $a \rightsquigarrow G(a) = a^2$.
It is easy to see that G is a covariant functor. (Note, $a \mid b$ implies $a \leq b$ implies $a^2 \leq b^2$.)

We introduce further examples.

(5) $G_x : A \longrightarrow B$, $a \rightsquigarrow G_x(a) = a + x$, for $x \in \mathbb{N}$.
Show that G_x is a covarinat functor.

(6) For $\alpha \in \mathbb{N}$ we define $G_\alpha : A \longrightarrow B$, $a \rightsquigarrow G_\alpha(a) = \alpha.a$.
Verify that G_α is a covariant functor.

(7) Note, that more general $G_{2n} : A \longrightarrow B$, $a \rightsquigarrow G_{2n}(a) = a^{2n}$, $n \in \mathbb{N}$, is covariant.

(8) Let $H : A \longrightarrow B$ be defined by $a \rightsquigarrow H(a) = -a$.
Verify that H is a *contravariant* functor.

We come to some *natural transformations* which arise in this context.

(9) Show that $\alpha : F \longrightarrow G$, given by the components $\alpha_a : F(a) \longrightarrow G(a)$ (i.e. $a \longrightarrow a^2$) is a natural transformation.

(10) $\beta : F \longrightarrow G_x$, given by the components $\beta_a : F(a) = a \longrightarrow G_x(a) = a + x$ is a natural transformation.

(11) $\gamma : F \longrightarrow G_\alpha$, defined by the components $\gamma_a : F(a) = a \longrightarrow G_\alpha(a) = \alpha.a$ is a natural transformation.

A basic and far reaching concept in category theory is the notion of **Adjoint Functors**. This expresses relationships between functors which are naturally related to each other, involving a corresponding universal property - this is characterized by adjointness. Such adjunctions arise in various disguises in category theory and applications. Rather often the "forgetful functor" arises in adjoint functor pairs. Subsequently, we give the definition of "Adjunctions / Adjoint Functors", also called "Adjoints".

Definition .4 *Let $F : \mathbf{A} \to \mathbf{B}$ and $G : \mathbf{B} \to \mathbf{A}$ be two functors.*
Let $\mathbf{A}^{op} \times \mathbf{B}$ be the product category (defined in the obvious way). For every pair of objects (X, Y) we have the sets of morphisms $\mathbf{B}(FX, Y)$ and $\mathbf{A}(X, GY)$. This leads to the following functors $\mathbf{A}^{op} \times \mathbf{B} \to \mathsf{SET}$:
$\mathbf{B}(F\_, \_) : \mathbf{A}^{op} \times \mathbf{B} \to \mathsf{SET}$ *and* $\mathbf{A}(\_, G\_) : \mathbf{A}^{op} \times \mathbf{B} \to \mathsf{SET}$. *With these preparatory considerations we can define the notion of adjoint functors as follows.*
F and G are called adjoint functors, if there is a natural equivalence
$\vartheta : \mathbf{B}(F\_, \_) \xrightarrow{\cong} \mathbf{A}(\_, G\_)$ *given by the components:* $\vartheta_{XY} : \mathbf{B}(FX, Y) \xrightarrow{\cong} \mathbf{A}(X, GY)$.

F is called left adjoint to G, denoted by: $F \dashv G$.
G is called right adjoint to F, denoted by: $G \vdash F$.
ϑ - or more precisely (F, G, ϑ) - is called adjunction (from $\mathbf{A}$ to $\mathbf{B}$).
The adjointness is a 1-1-correspondence between arrows: $X \to GY$ in $\mathbf{A}$ corresponds to $FX \to Y$ in $\mathbf{B}$. This is schematically expressed by

$$\frac{X \to GY}{FX \to Y}$$

An example of an adjoint situation is the diagonal functor in combination with the binary product functor. Let $\Delta : \mathbf{A} \to \mathbf{A} \times \mathbf{A}$ be the diagonal functor on a category $\mathbf{A}$, assigning $a \in Obj(\mathbf{A})$ to (a, a) and $f : a \to b$ to $(f, f) : (a, a) \to (b, b)$. If Δ has a right adjoint $G : \mathbf{A} \times \mathbf{A} \to \mathbf{A}$, $\Delta \dashv G$, then G is the binary product functor. Thus we can say: In $\mathbf{A}$ there are binary products, iff Δ has a right adjoint G.
We cannot present further material here and refer to the literature.

A collection of several examples of adjoints can be found in chapter IV, section 2 of the textbook Mac Lane, S.: "Categories for the Working Mathematician". Springer Verlag, Graduate Texts in Mathematics **5** (1998), 2nd edition.

3.3 Limits, Colimits

Subsequently, we define further basic notions which are of importance for constructive categorical modeling. The notion of a *diagram* is essential in the definition of *limit* and *colimit.* Actually, in the general framework of (co)limits it is possible to get back other notions like product, coproduct, terminal and initial objects, etc., by applying the (co)limit definition to a corresponding specified base diagram.

Definition .5 *A diagram $\mathbb{D}$ in a category A is defined by a family of objects $\{X_i\}_{i \in I}$ and a family of morphism sets $F_{ij} \subseteq Mor(X_i, X_j)$, for $i, j \in I$. We shall write $f_{ij} : X_i \to X_j$ for $f_{ij} \in F_{ij}$. It is explicitly mentioned that F_{ij} can be empty, i.e. there may not be an arrow between the "nodes" X_i and X_j in the diagram.*
A cocone $\mathcal{C}$ of the diagram $\mathbb{D}$ consists of an object X in A and morphisms $f_i : X_i \to X$, for $i \in I$, such that $f_i = f_j \circ f_{ij}$ for all $i, j \in I$. This can be illustrated by the following commutative triangle:

$$\begin{tikzcd} & X & \\ X_i \arrow[ur, "f_i"] \arrow[rr, "f_{ij}"] & & X_j \arrow[ul, "f_j"'] \end{tikzcd}$$

A colimit of the given diagram is a cocone $\mathcal{C}$ with the following universal property (defining $\mathcal{C}$ in terms of a "normal form"):
for every other cocone $\mathcal{C}'$, given by morphisms $f'_i : X_i \to X'$, $i \in I$, there exists exactly one morphism $f : X \to X'$, such that $f'_i = f \circ f_i$, for all $i \in I$.
In these terms $\mathcal{C}$ can be called a universal cocone of the diagram which is the colimit, denoted by: $\varinjlim(\mathbb{D})$. (In the literature, several other notations may be found, such as $\varinjlim_{i \in I} X_i$ or $\varinjlim_{\mathbb{D}} X_i$ or $\varinjlim X_i$.) For our purposes here we find it suggestive to write $\varinjlim(\mathbb{D})$ since the colimit as a limiting cocone depends on the given base diagram $\mathbb{D}$ and makes it "complete".
Reversing the arrows in the definition of a colimit of a diagram $\mathbb{D}$ results in the dual notion called limit of $\mathbb{D}$, denoted by $\varprojlim(\mathbb{D})$.

3.4 ML Description of a Category

Only very briefly we sketch the description of the datatype CAT in ML notation wich is the basis for all **implementational aspects**, we cannot go into the details here and refer to the excellent textbook Rydeheard, Burstall: "Computational Category Theory".
We have to consider the following two **types:** OBJECTS: $'\mathsf{o}$ and ARROWS: $'\mathsf{a}$ and four **functions** defining the basic structure of the category involving objects and arrows. These functions are

$$\begin{array}{ll} \left.\begin{array}{l} \textit{source}\ (\textbf{dom}) \\ \textit{target}\ (\textbf{codom}) \end{array}\right\} & '\mathsf{a} \longrightarrow '\mathsf{o} \\ \textbf{identity} & '\mathsf{o} \longrightarrow '\mathsf{a} \\ \textbf{composition} & '\mathsf{a} * '\mathsf{a} \longrightarrow '\mathsf{a} \end{array}$$

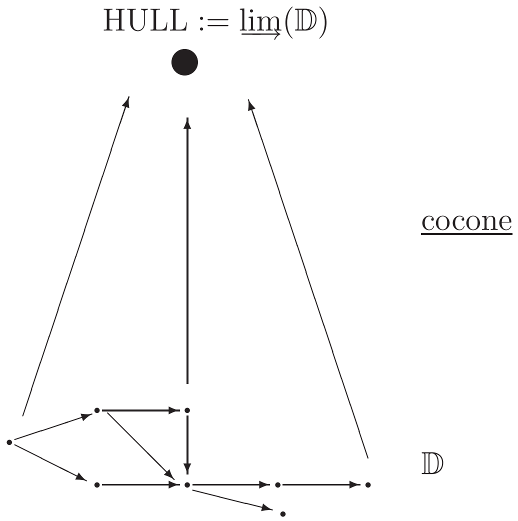
The definition of **type of categories** is a 4-tuple of functions with given type (as above):

$$\mathsf{datatype}('\mathsf{o}, '\mathsf{a})\mathsf{Cat} = \mathsf{cat\ of}\ \ ('\mathsf{a} \longrightarrow '\mathsf{o}) * ('\mathsf{a} \longrightarrow '\mathsf{o}) * ('\mathsf{o} \longrightarrow '\mathsf{a}) * ('\mathsf{a} * '\mathsf{a} \longrightarrow '\mathsf{a}).$$

This type declaration defines a **type constructor Cat** which gives the type of all categories with specified type for objects and arrows.

3.5 A Hull Construction. Greatest Common Divisor as Limit

The following diagram illustrates the categorical construction of a HULL in a deductive system as a colomit. It is taken from the article *J.P.: "On a General Notion of a Hull"* in the book Automated Practical Reasoning, J.Pfalzgraf and D.Wang (Eds.). Texts and Monographs in Symbolic Computation, Springer Verlag Wien-New York (1995).



Subsequently, we present a popular example from ARITHMETIC with which we would like to demonstrate and illustrate how we can apply our general approach to obtain a concrete "visualization" of notions always using the same concept which we established previously.
Let us consider the *Euclidean algorithm* for constructing the *greatest common divisor* (**gcd**) of two natural numbers a and b, $\gcd(a,b)$, where we assume that $b < a$. In the Euclidean algorithm two basic arithmetical relations are used on natural numbers $a, b, \ldots, u, v, \ldots$, namely the *division*, $a|b$, and the *less relation*, $u < v$. We are going to visualize these relations by the following two types of arrows $a \mapsto b$ (for $a|b$) and $u \to v$ (for $u < v$).
Now, let $r_0, r_1, \ldots, r_n$ denote the sequence of remainders produced by the algorithm, where we start with the initial division $a = q_0 \cdot b + r_0$ with remainder $r_0 < b$, and repeat this procedure until we reach the last nontrivial remainder, denoted by r_n, which divides all the other remainders and a and b. Actually, we have $\gcd(a,b) = r_n$. Applying our categorical modeling principle we can reinterpret the gcd in terms of a limiting cone with r_n as its "top". To this end we "visualize" all the relations which are the result of the algorithm leading to the following base diagram $\mathbb{D}$

$$0 \to r_{n-1} \to r_{n-2} \to \ldots \to r_0 \to b \to a$$

(note that this corresponds to the <-relation). Then we take this as the base diagram and form the limit with r_n as its top object. We obtain the same result when we use the completed base diagram including the smallest remainder r_n itself

$$0 \to r_n \to r_{n-1} \to r_{n-2} \to \ldots \to r_0 \to b \to a.$$

Having a look at the categorical definition of a limit we use the arrows $r_n \mapsto r_i$ ($i = 0, \ldots, n-1$) and $r_n \mapsto b$, $r_n \mapsto a$, as the "projection arrows". These correspond to the arrows $\{f_i : X \to X_i\}_{i \in I}$ in the general definition of a limit (cf. the definition above). Thus we have $\gcd(a, b) \stackrel{\mathrm{Lim}}{=\!\!\rightarrow} \{r_i, b, a\}$ as the limit of the given base diagram, cf. the figure below.

$$\begin{tikzcd}
 & & & & r_n = \gcd(a,b) = \varprojlim(\mathbb{D}) \arrow[dllll, mapsto] \arrow[dlll, mapsto] \arrow[dll, mapsto] \arrow[dl, mapsto] \arrow[d, phantom, "\cdots"] \arrow[dr, mapsto] \arrow[drr, mapsto] \arrow[drrr, mapsto] & & & \\
\perp = 0 \arrow[r] & r_n \arrow[r] & r_{n-1} \arrow[r] & r_{n-2} \arrow[r] & \ldots \arrow[r] & r_0 \arrow[r] & b \arrow[r] & a
\end{tikzcd}$$

We can observe that by dropping the arrows $r_i \to r_{i-1}$ in $\mathbb{D}$ and just taking the "discrete" set of objects $\{r_{n-1}, \ldots, r_0, b, a\}$ as a new base diagram (without arrows), we can write the gcd as the product of the objects $\{r_{n-1}, \ldots, r_0, b, a\}$ with the same projections as in the previous limit descriptions. Our final step in the application here is to extract in words the "universal property" of that limit (product). That means, knowing the general formulation of the universal property of a limit in category theory, we read this property in terms of our concrete arrows, that means we write down this property in our concrete semantical modeling context. This gives us back (as universal property) the well-known definition of $\gcd(a, b)$ as we learned it at school. To repeat it: "the $\gcd(a, b)$ divides all $r_{n-1}, \ldots, r_0, b, a$" (this is modeled by the projection arrows as mentioned above) and universality corresponds to the words "and every other number (object) g' dividing $r_{n-1}, \ldots, r_0, b, a$ (i.e. with projection arrows to these objects) must divide $\gcd(a, b)$, i.e. $g' \mapsto \gcd(a, b)$". This is the common schoolbook definition obtained back as universal property.

If we concentrate only on the relation $\gcd(a, b) \mapsto a$ and $\gcd(a, b) \mapsto b$ (and forget about the other remainders), then $\gcd(a, b)$ together with these two arrows again can be interpreted as a product in the categorical sense and its typical universal property yields the definition of the gcd in a similar way as we mentioned it in the previous considerations.

Now we apply another typical categorical principle, namely *duality* which, briefly spoken, means that we reverse all the arrows in the definition of the notion of a product. The result is the categorical *coproduct* of the two objects a and b. To be more precise we have an object which we denote by $\mathrm{lcm}(a, b)$ and two arrows (reversing the arrows of the $\gcd(a, b)$) $a \mapsto \mathrm{lcm}(a, b)$ and $b \mapsto \mathrm{lcm}(a, b)$. Then again interpreting the universal property of the coproduct in this specific context gives us the well-known definition of the *least common multiple* of a and b ($\mathrm{lcm}(a, b)$). If we interpret the arrow relation in terms of lattice theory then we can say that $\gcd(a, b) = \inf(a, b)$ and $\mathrm{lcm}(a, b) = \sup(a, b)$.

We recall that the categorical product (coproduct) is formed as a limiting cone (cocone) of the underlying "discrete" diagram $\{a, b\}$ given by the objects a and b. And thus $\mathrm{lcm}(a, b)$ can be considered to be the hull or closure of that diagram in the general meaning of this article.

We confined the previous considerations to natural numbers and make the remark here that, qualitatively, a similar "visualization" by a categorical diagram can be given for other (Euclidean) domains.

This example also shows that there is a natural motivation to speak about *morphisms of different type* in the underlying categorical model. This extends naturally the notion of a category but it is compatible with the categorical concepts.

3.6 Selected Literature

More about categories can be found in the extensive literature. We refer to the literature list above in section 2, and mention here some selected titles:
Mac Lane, S.: Categories for the Working Mathematician. Springer Verlag, Graduate Texts in Mathematics **5** (1998), 2nd edition. Adámek, J., Herrlich, H., Strecker, G.E.: Abstract and Concrete Categories. John Wiley & Sons (1990). Goldblatt, R.: Topoi. The Categorial Analysis of Logic. North-Holland, Elsevier Publishers (1986). Lawvere, F.W., Schanuel, S.H.: Conceptual Mathematics. A First Introduction to Categories. Cambridge University Press, (2000). Pierce, B.C.: Basic Category Theory for Computer Scientists. The MIT Press Cambridge, Massachusetts and London, England (1991). Pümplin, D.: Elemente der Kategorientheorie. Spektrum Akademischer Verlag, Hochschultaschenbuch (1999).

4 On CAT Modeling of General Relations

This section gives a brief summary of some considerations in the article *J.P.: "On a General Notion of a Hull"* in the book Automated Practical Reasoning, J.Pfalzgraf and D.Wang (Eds.). Texts and Monographs in Symbolic Computation, Springer Verlag Wien-New York (1995).
Starting point of our work was a question raised by Dov Gabbay (in the course of the Esprit project MEDLAR - "mechanising deduction in the logics of applied reasoning") about a suitable natural notion of a *hull operation* for deductive systems. Concretely, the problem is to find a construction which, for a given *deduction diagram* (a subset of formulas and deduction relations between them in an underlying deductive system), returns a (possibly larger) diagram which contains the original one and is "closed" under this operation. The construction we are proposing results from a look at this question from a categorical perspective interpreting a deductive system in terms of a corresponding *category*. In this setting, i.e. using category theory as our linguistic basis, we are led in a natural way to a notion of *hull operation as a colimit* in the underlying category. Note, the diagram at the end of the previous section 3 illustrates the result of such a HULL construction.

4.1 Some Technical Preparations

Let $X \neq \emptyset$ be a set and $R \subseteq X \times X = X^2$ a (binary) relation on X, sometimes denoted by (X, R). As usual, we express a relationship $(x, y) \in R$ between two elements $x, y \in X$ by xRy, symbolically. Having our "CAT intentions" in mind, we "visualize" a relation xRy by an arrow $x \to_R y$ or just $x \to y$, for short. Such an arrow will be interpreted as a morphism between objects x and y.
Now, let $(X, \leq)$ denote a *partially ordered set* ("poset"). We recall, that in a poset the relation $\leq$ is *reflexive, transitive, antisymmetric*. Then we can associate with this poset the following *category* X. The objects of X are the elements of the set X and for $a, b \in X$ there is a morphism $a \to b$, if and only if a and b are in relation, i.e. $a \leq b$. It is easy to verify that this defines a category. The transitivity of $\leq$ gives the composition relation in X and the reflexivity guarantees the existence of the identity arrows for each object of X. We note explicitly that there is at most one arrow between two objects (if there is one),

that means for the number of elements of the morphism sets we have $|\mathrm{Mor}(a, b)| \leq 1$. Thus these examples show that there might be empty morphism sets in categories. Associativity of the composition in X obviously holds. Note, that the antisymmetry property of the poset relation is not needed to verify that X becomes a category.
Summarizing, we can say that every reflexive, transitive relation leads to such a "small" category as introduced above.

Accordingly, for a *deductive system* $(X, \vdash)$ we translate $p \vdash q$ into the arrow (morphism) $p \to q$ between the objects (formulas) p, q. If we wish to emphasize the lattice theoretic aspects (which are naturally present), we may prefer to write $p \leq q$ instead of $p \vdash q$ or $p \to q$. This should show clearly enough how we establish our "translation table" in order to wander around in the different disciplines.

For practical reasons - in order to reach a large area of applications - we extend our CAT modeling approach to *arbitrary* relations (X, R). In such a case, we are not able to associate directly a category to the relation as we did it before since transitivity, reflexivity do not hold, in general. But from the categorical perspective again we interpret a relational structure as a certain diagram of arrows "visualizing" the given relations between the objects which form the "nodes" of the diagram. It turns out that we can always "embed" such a diagram in an associated *PATH-category* having comparable behavior as the category associated to a reflexive, transitive relation, although being a little "bigger" concerning the morphism structure. Having this environment we can proceed in a natural way and again introduce the previously mentioned hull operation via colimits.
We point out: The introduction of the *associated category* PATH allows to use and apply all the modeling principles and constructions provided by CAT in the corresponding situations.
Let us consider, for illustration, a simple practical example of real life: Looking at general relational structures is quite natural since transitivity and even reflexivity are not always existent in applications. As a practical example let us look at a road map where the nodes (objects) are towns and the arcs (arrows) are road connections, then not every pair of towns has a direct connection (arrow), in general. Therefore, generally, starting from a point we have to follow a path of direct road connections passing several nodes (towns) before we can reach a goal. Similarly, in deductive systems transitivity can be missing.

4.2 The Category PATH

For the technical definitions, again let $R \subset X \times X$ or (X, R) denote a (general) relation. We associate to it the following category denoted by $\mathsf{PATH(X,R)}$ or just PATH for short, if no confusion can arise.
The *objects* are the elements $x \in X$ and *arrows (morphisms)* are defined by sequences (paths) of adjacent arrows. That means, there is always a morphism $x \to y$, if $(x, y) \in R$ (or xRy), but if we have arrows (morphisms) $x \to y$ and $y \to z$, then, in general we do not have a "direct arrow" $x \to z$, since the relation need not to be transitive. But what we can always do is forming a sequence (path) of consecutive arrows, like $x \to y \to z$ in the previous case. This is then a morphism of a more general type between x and z. More generally, we can have (finite) sequences denoted for example by $x_0 \to x_1 \to \ldots \to x_n$ which is a morphism in $\mathrm{Mor}(x_0, x_n)$ in the new sense of our definition, but can also be

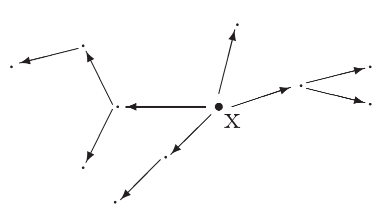
interpreted as the composition of other morphisms which will be represented by adjacent parts of that whole path. In a category normally we need the identity arrow id_x for each object x. We can add this as a requirement if there is really a necessity from a theoretical viewpoint; in practice this may be irrelevant. In PATH it can be the case that there exists more than one path between two nodes a and b, therefore in PATH we can have for the sets of morphisms $|Mor(a,b)| > 1$, in general, in contrast to the category which is associated to a reflexive, transitive relation as considered before. Based on these considerations we can see that PATH becomes a category.

In such cases for general relations it holds: The definition of a *closure operation* or *hull* is then defined analogously as above in terms of the *colimit* of a corresponding subdiagram in PATH. We should not forget to mention that the question of the existence of (co)limits in a given category is not our subject in this exposition here.

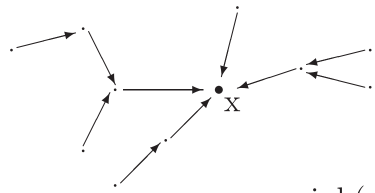
NOTA: We want to mention explicitly that associating the category PATH to a relation is somehow in contrast to forming the *transitive hull*. We deliberately emphasize this different point of view. Namely, concerning the modeling of relational structures an additional aspect is attracting our interest: What is the "minimal" necessary information describing a given structure ? Is it necessary to embed it in the (complete) transitive closure? Or can we come along with less information (partial diagrams or so)?
In our article cited above, we explain what we mean by having a look at *combinatorial group theory*. There, one describes a group by *generators and relations* without giving the complete group table (this would correspond to a complete diagram). We only give some generators and suitable relations among them and this already contains the whole information for describing the group. This can be depicted as a corresponding arrow diagram, the so-called *Cayley diagram*.
In this sense we think at a *generator and relations* point of view for encoding our diagram (representing the given relational structure). That means we look for a certain *skeleton* containing all the essential information about the original system (relation). Another aspect of interest is to consider a possible *spanning tree* of PATH as a certain "irredundant" representation of the original structure.

4.3 Prospects: Remarks on General CAT Semantics

Let a be a node of a given deduction diagram $\mathbb{D}$, i.e. an element/formula of the corresponding deductive system $(X, \vdash)$, for which we first assume again that it is a reflexive, transitive system. As previously done we visualize the $\vdash$ again by the arrow $\rightarrow$, then we put as a "*fiber* over a" the *principal filter* F_a (or *principal ideal* I_a) generated by a. Noting here that $F_a := \{x \in X | a \rightarrow x\}$ and $I_a := \{x \in X | x \rightarrow a\}$ it is suggestive to interpret F_a as a "source(a)" (all arrows and paths of arrows starting in a) and accordingly we can say that I_a is a "sink(a)" (cf. the figures included for illustration).



source(x)



sink(x)

These notions provide us with (local) algebraic (lattice theoretic) information about the system under consideration. The collection of all the fibers indexed by the elements of the given deductive system is called total space of the resulting *fibering*. The underlying diagram, more precisely the associated category, is called base space. Each particular fiber provides us locally with certain algebraic information centered around the node a. Moreover, in these fibers we can obtain a natural *Heyting algebra* structure. Then we use the filtering sets and ideals to introduce an induced topology on the base space (i.e. the underlying diagram). This finally gives us the possibility to define an associated (pre-)sheaf of Heyting algebras and thus opens the view to a general *sheaf semantics* and *topos* theoretic aspects for deductive systems.

Obviously analogous considerations can be done for more general relational structures for which we can introduce the associated category PATH in which we then can perform all the analogous categorical constructions. Moreover, this associated category is then the base space of the associated fibering.

Once we have established this general semantical modeling basis it will be natural to use it for working on *classification* results of relational structures interpreting the associated (local) algebraic structures, the base diagram which is involved and the associated fiberings and sheaves in the sense of *mathematical invariants* to classify given systems.

5 ACCAT and Theoretical Neurobiology

In this section we include the Abstract of a Talk which I presented at our 2nd Workshop "Applied and Computational Category Theory", ACCAT'07, Satellite Event at the European Conferences on Theory and Practice of Software, ETAPS-2007, March 24-April 1, 2007, Braga, Portugal.

In this short contribution, we make some comments on links of own work to Theoretical Neurobioloy. In December 2002, I obtained an invitation to give a plenary lecture at the "International Conference on Theoretical Neurobiology", February 24 - 26, 2003, New Delhi, India. Organized by National Brain Research Centre, India. The talk I intended to present there had the title "Modeling Connectionist networks: Categorical, Geometric Aspects (towards 'Homomorphic Learning')". I could not attend this conference, but I was invited to give a talk on this topic (same title) at the International Conference on Computing Anticipatory Systems CASYS'2003, Liège, Belgium, August 11-16, 2003 - cf. *J.Pfalzgraf: Modeling connectionist networks: categorical, geometric aspects (towards "homomorphic learning"). Proceedings CASYS'2003, "Computing Anticipatory Systems: CASYS 2003". August 11-16, 2003, Liège, Belgium. American Institute of Physics, AIP Conference Proceedings, Vol.718 (2004), D.M.Dubois (Ed.).* This contribution received a Best Paper Award.

It is interesting to read the announcement of the New Delhi Conference on Theoretical

Neurobiology. Scope of the Conference: "Algebra, Geometry, and Logic of Cognition. Artificial and Natural Intelligence. Cognitive Neuroscience. Computational Neuroscience. Dynamical Systems Theory. Functional Imaging of Brain. Neural Correlates of Consciousness. Neural Networks. Neuroinformatics. Neuropsychiatric Disorders".

Aims of the Conference: "....... One of the main goals of the conference is to provide a platform for experimental and computational neuroscientists to closely interact and exchange ideas with mathematicians working in the areas of **category theory and higher dimensional algebra** and explore the potential of these sophisticated mathematical methods, in view of their success in solving problems hitherto intractable within the point set theoretic framework, to meet the demands of cognitive neuroscience data. These interactions will also inspire mathematicians to develop new formal tools and techniques tailor-made to suit the unique nature of the brain and thereby accelerate the development of a comprehensive theory of brain function that provides a scientific account of not only photons and action potentials but also of percept, thoughts, emotions, intention, and action".

Some years ago, I had a long cooperation period with H.Geiger - a specialist in **Artificial Neural Network (ANN)** applications. It turned out that the net structures which he was permanently using can be mathematically modeled in the sense of a **geometric net theory** which I could introduce, based on so-called **noncommutative geometric spaces** - they form the category **NCG**. That rather new geometric discipline (introduced by Johannes André in the 1970ies in Saarbrücken) was a large part of my work done in "pure" mathematics. I found a new general approach to model André's original notion of a geometric space. Actually, these geometric aspects of ANN structure modeling formed the interface to category theory (CAT) in the following sense. To an ANN we can associate a geometric net which is induced by a corresponding geometric space. Geometric spaces form the category **NCG**, thus we are motivated to introduce a **category of geometric nets, GeoNET**. But what is then the basic feature of ANN theory, namely "LEARNING"? It turns out that a LEARNING STEP can be interpreted as a corresponding MORPHISM in the category **GeoNET**. This motivates us to speak about a novel concept called "Homomorphic Learning".

In the course of the previously mentioned cooperation we made an interesting experience: The exploitation of this categorical approach for modeling the ANNs in a concrete network computer simulation (which H.Geiger applied in an industrial project on optical quality control in his company) led to surprising results and effects. The *economy of the mathematical model* became visible in a considerable increase of performance in the whole computer simulation that even could be measured in terms of concrete *reductions of the overall project costs* ! Thus, CAT can help to save money.

An interesting topic of intended future work concerns **simplex configurations** and the **simplicial structure** of a network. I introduced the notion of a **simplex condition** ($\mathbf{Sim_q}$) in noncommutative geometry - cf. *J.Pfalzgraf: A note on simplices as geometric configurations. Archiv d. Math. 49 (1987)* - (Sim_q) deals with parallel shifting of q-simplices in a space leading to a rich geometric structure. A main theorem in noncommutative geometry (Hauptsatz by J.André) gives a geometric criterion of a finite group space on basis of the validity of the simplex conditions (Sim_q). A group space is a geometric space induced by a group action on the set of the points of the space, it has interesting geometric properties. It turns out that a regularly structured network fulfill-

ing the simplex conditions has an underlying symmetry group generating this space by a corresponding group operation.
Concluding, we quote an interesting remark by William C. Hoffman, Ph. D., Director of the Institute for Topological Psychology, Tucson, Arizona.
"Why Topological ?" : A thing is not just a thing – it has form and meaning. Form perception is inherently geometric – the things one sees consist, as they do, of geometric objects. And as for meaning, if one accepts the quaint fancy of the Connectionists that thought processes consist of point "neurons" and the paths connecting them, then simplicial topology enters as the lines, triangles, tetrahedrons, etc. that connect cognitive "chunks" by "trains of thought".

6 CAT Modeling of Neural Network Structures

In this section we give a brief outline of existing work and prospects of intended future work (in progress), following the article *J.Pfalzgraf: Modeling connectionist networks: categorical, geometric aspects (towards "homomorphic learning")*. Proceedings CASYS'2003, "Computing Anticipatory Systems: CASYS 2003". August 11-16, 2003, Liège, Belgium. American Institute of Physics, AIP Conference Proceedings, Vol.718 (2004), D.M.Dubois (Ed.). [Received a Best Paper Award]. Since we cannot go into technical details here we prefer to give a short description of the methods and results presented in our publication.

Work in interdisciplinary fields is very interesting and always a great challenge. We present work on applications of mathematical methods to modeling problems arising in the area of artificial neural networks (ANN).
In the article we focus on a mathematical approach for modeling the stucture of an artificial neural network (ANN). The approach is motivated by neurophysiological knowledge about a certain class of network structures (we called it Geiger paradigm). In the analysis of network structures, considering assemblies of cells (neurons) in biological nets, from a geometric point of view one can indentify and interpret, locally, what is called a **geometric configuration**. These configurations are composed by nodes (cells) and their connections. Following notions from algebraic topology, we are speaking about **simplicial configurations** (e.g. triangular, tetrahedral configurations, etc.).
An interesting hint to triangular configurations can be found in W.H.Calvin: The Cerebral Code. Thinking a Thought in the Mosaics of the Mind (MIT Press, Cambridge, Mass., 1996).
Based on our mathematical approach we introduce a general definition of ANN. This general definition served as guideline for implementation work of a powerful ANN simulation tool (FlexSimTool) which has been established in our group. Further development is intended by means of additional mudules for simulating multiagent systems, distributed logics (logical fiberings), robotics scenarios.
Since our interest focuses on the *net structure* of an ANN, neurons are abstractly modeled as *nodes* of a directed (and colored) graph, independent of their specific neuron types.
Analysis of the generic structure of the class of networks (ANNs) which we consider suggests the application of the following disciplines for general mathematical structure modeling, in a natural way. It turned out that aspects from **geometry (so-called "noncommutative geometric spaces")** and **category theory** arise, naturally.
It can be observed that a network is regularly structured in the following way. From the

view point of each node, the local structure of the network is the same. Mathematically, one speaks of a pointed space when selecting a distinguished point in a space and describing the structure locally. In this sense, the whole (global) network structure can be homogeneously described by its pointed spaces, i.e. the essential information for structuring the net is given locally. Thus, the networks can be interpreted as **directed colored graphs**. Input and output layers do not have to be distinguished. It is reasonable to associate an (abstract) "geometric net" to a given ANN, globally, and to introduce a smaller associated net that reflects the local network structure (containing the essential information). A (noncommutative) **geometric space** has a natural interpretation as a directed colored graph - we call it **geometric net**. This suggests interpretation of an ANN structure as a geometric net. Noncommutative geometric spaces form a **category (NCG)** with geometric spaces as objects and structure preserving maps as morphisms. This, in turn, leads to the **category of geometric nets** and to the interpretation of an ANN structure as the object of a corresponding category. This point of view suggests to interpret a **learning step as a morphism** between networks (objects) and **learning** as a finite **sequence of morphisms** in such a category.
Our mathematical model of an ANN is very general and formed the theoretical basis for the implementation work of our ANN-simulator FlexSimTool.

We briefly discuss the basic **role of simplices and simplex configurations**. Leading to a natural explanation how geometric and topological aspects and methods arise. In noncommutative geometric spaces the simplicial structure is an important aspect and a main result shows that a space with rich simplicial structure can be generated by the action of a (symmetry) group. In general, basic interdisciplinary aspects arise naturally.

An **industrial project** is mentioned with applications of ANN simulation (by H.Geiger) to problems in optical quality control where our new mathematical modeling approach for ANN structuring was industrially exploited for the first time. Concerning performance aspects it could be observed that the use of our mathematical ANN model as basis for simulation showed an **economic effect** (i.e. performance improvement). Indeed, the use of **category theory** even allows to explain why **"learning can be made cheaper"**.

Below, we show the commutative square which gives the explanation why "learning can be made cheaper".

$$\begin{tikzcd} X \arrow[r, "\sigma_X"] \arrow[d, "L"'] & X_{x_0} \arrow[d, "L"] \\ X' \arrow[r, "\sigma_{X'}"] & X'_{x_0} \end{tikzcd}$$

REMARK: This commutative square belongs to a corresponding natural transformation $\sigma : Id \longrightarrow P_{x_0}$, explained in the article.
Analyzing this diagram one can observe, that "learning on the right side is cheaper". This fact can be exploited to reduce the complexity of training (learning) in particular ANN applications (computer simulations).

In section 7 of the article *J.Pfalzgraf: "Modeling connectionist network structures: some geometric and categorical aspects".* Annals of Mathematics and Artificial Intelligence 36 (3) (2002), 279-301 (special issue on "AI and Symbolic Computation", Editors: J.Calmet and E.Roanes-Lozano). Kluwer Academic Publishers, an industrial application of ANN simulation is presented where the economic effect of cheaper learning, as previously described, could be observed even in terms of reduction of production costs. The formal mathematical proof of this economic fact is given by the commutative diagram above.

7 CAT Modeling of Multiagent Systems (MAS)

Subsequently, we give a brief report on first steps to model **multiagent systems (MAS) categorically** aiming at a unifying generic approach. We include the short exposition which we presented as an invited contribution at *16th International Conference on Systems Research, Informatics and Cybernetics (InterSymp-2004), July 29 - August 5, 2004, Baden-Baden, Germany*, with the title "On Categorical and Logical Modeling in Multiagent Systems" (to be published in the InterSymp-2004 proceedings). Continuation of work on these topics is part of our future working program.

Abstract: A general first approach of categorical modeling in multiagent systems (MAS) is presented. Looking in this direction it is natural to model agents as objects of the category to be defined and all kind of relations between agents as morphisms. This mainly concerns all kind of communication between agents. In order to achieve a general ("generic") definition for a category of MAS, it is indispensable to introduce typed objects ("agents") and typed morphisms ("arrows"). This suggests in a natural way to introduce certain subcategories characterizing typical multiagent systems as they are discussed in literature with corresponding types of agents and types of relations between agents (i.e. communication, cooperation). In the course of the whole project, finally, we can resort to our work on general categorical semantics for relational structures, with a view to sheaf semantics and topos aspects. Concerning logical modeling of MAS we are concentrating on logical fiberings, they provide systems of distributed logics for MAS. The basic idea is to assign to every agent its own local logic (logical fiber) and the collection of all the fibers over the base space of agents forms the fibering (global logical state space). We can resort to existing work on many-valued logics and logical fiberings on our way to propose a model of a (parallel) reasoning machine associated to a MAS or a (possibly small) group of agents. Many old and new aspects will provide the basis for an extensive program of work with many potential applications, including industrially relevant applications. MAS and ROBOT simulation will be an essential part of our work. □

Some general remarks on MAS follow. So far, in literature no generally accepted ("universal") definition of "Agent" exists. In the textbook M.J.Wooldrige: An Introduction to MultiAgent Systems (John Wiley & Sons Ltd, 2002) the definition is given: "An agent is a computer system that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives".

From so-called *intelligent agents* commonly the following list of capabilities, suggested by Wooldridge, is expected:

- **Reactivity**: Intelligent agents are able to perceive their environment, and respond in a timely fashion to changes that occur in it in order to satisfy their design objectives.
- **Proactiveness:** Intelligent agents are able to exhibit goal-directed behavior by taking the initiative in order to satisfy their design objectives.
- **Socialability:** Intelligent agents are capable of interacting with other agents (and possibly humans) in order to satisfy their design objectives.

In addition, **mobility** and **learning** aptitude are some further capabilities which are important for agents in several application areas.

MAS - System of Interacting Agents:
Typical structure of a MAS (*N.R.Jennings*): "System with a number of agents which **interact** with one another through **communication**".

For our new approach of essential interest:
All kind of **relationships between agents**.
For Example: **Logical constraints** and specific **dependencies** in cooperation "overlaps" (cf. "Generic Modeling Principle").
Specific **order relations** as basis for comparisons (e.g. "winner takes all", "dominant agents", "power criteria", "skills/qualification criteria").
Viewpoint of Physics: MAS = Many-particle System With Interaction (?)
Vielteilchensystem mit Wechselwirkung (?)

Communication. Technical Standards:
A basic theory of communication is **speech act theory** treating communication as action (i.e. speech actions performed like actions).
Some Technical Standards evolved from work on the development of **agent communication languages:**

- **KIF:** Knowledge Interchange Format

(closely based on 1st order logic - to express message content)

- **KQML:** Knowledge Query and Manipulation Language

(message-based language for agent communication, defines a common format for messages)

- **FIPA** Agent Communication Languages:

(1995, the Foundation for Intelligent Physical Agents (FIPA) started work on development of standards for MAS technology. Academic and industrial partners are cooperating in FIPA)

- **ACC:** Agent Communication Channel

(for communication between agents from different agent platforms)
Important organisations for standards in the area of **Software Agents** are OMG and FIPA.

- **OMG:** Object Management Group, a non-profit organisation for standards dealing with CORBA (standards for distributed software systems).

Introducing a Category "MAS" - first steps

Objects: AGENTS (Types of Agents described by properties)
Morphisms: RELATIONS between agents (Types of Relations are specified).

Resorting to Existing Work: CAT modeling of relations - every relational diagram is a CAT $\rightsquigarrow$ towards a **categorical semantics** for general relational structures.

Communication Between Agents:
A **morphism** (**typed arrow**) between agents, specifying a corresponding **"communi-**

cation type".

Subsystem of a MAS - i.e. subgroup of (cooperating) agents $\longleftrightarrow$ **Subcategory**.
Mapping between two MAS $\longleftrightarrow$ **Functor** between corresponding CATs.

Idea ("Vision"): Exploiting universal construction principles of CAT for **constructive MAS-modeling** $\rightsquigarrow$ Construction of Scenarios....

Concluding: Perspective - Prospects

Our **Categorical Modeling** approach is a first step towards a *general ("unifying")* approach to model all kinds of agents and communication, cooperation which are relevant for theory and practical applications. A variety of industrially relevant problems can be treated with MAS techniques.

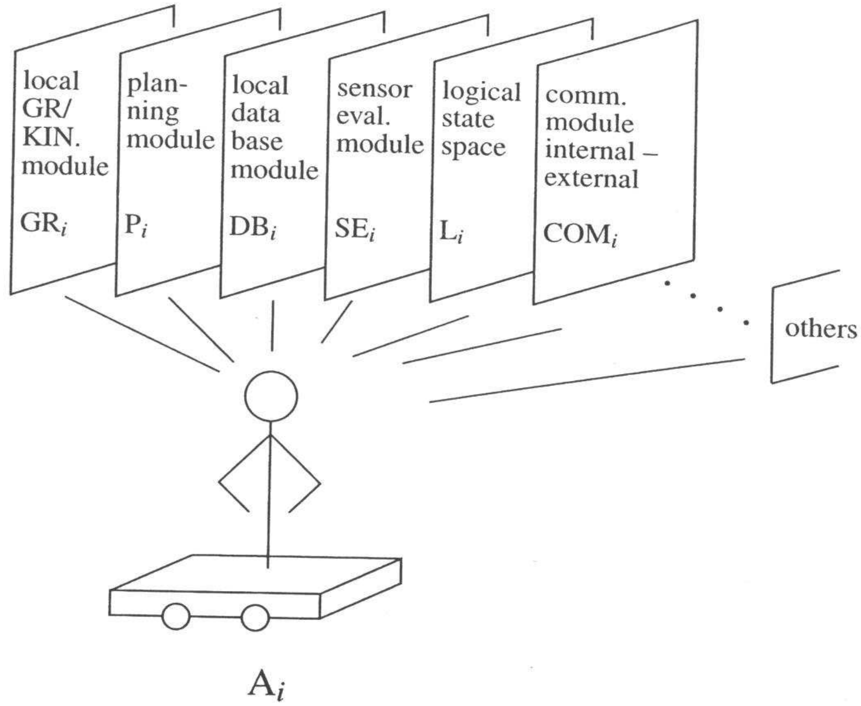
We distinguish between two main important classes of agents:
Technical and Artificial Agents (TAA) like ROBOTs, Working CELLs, Single Machines, Computing Devices, etc. (in particular for production automation and manufacturing models).

Human Agents (HA) are of basic importance for broad applications of MAS techniques, especially to industrially relevant problems, but also to general economically relevant services (e.g. eTourism). From a general point of view, human agents are an essential part of a MAS for modeling purposes in manufacturing. Industrial applications require the **integration of HA and TAA** in a MAS to model manufacturing/production scenarios realistically and optimally.

Finally, **Simulation of MAS** is an important aspect for research and development in science and industry. Computer simulation of MAS and ROBOTs is an essential part of our intended future work. It is of great relevance to be able to construct demonstrators of scenarios.

Concluding this section, we give a brief description of a schematic drawing of our model of a so-called "MEDLAR Practical Reasoner" as devised in our work for the MEDLAR Esprit Project and published in *J.Pfalzgraf, K.Stokkermans: "On robotics scenarios and modeling with fibered structures".* Automated Practical Reasoning, J.Pfalzgraf and D.Wang (Eds.). Texts and Monographs in Symbolic Computation, Springer Verlag Wien-New York (1995).
This model of an agent A_i consists of special modules ("fibers") forming the *reasoning state space* of the agent. In the picture are displayed, in particular: GR_i a geometric reasoning module, P_i a planning module, DB_i a data base for agent i, SE_i sensor evaluation units, L_i logical state space of agent i, COM_i a communication module.

In the framework of the introduction of the notion **space- and time dependent formulas** we developed the **generic modeling principle** for MAS. Very shortly speaking, this is a decomposition principle for complex scenarios into smaller local subparts. To an agent its individual (local) working domain is assigned. The domains have a (geometric) common overlap in all those areas of the working space where cooperation of agents can be performed. It is realistic to allow that semantics of logical formulas in the logical state space of an agent can change in dependence of the locations (coordinates) where the agent is moving around.



For more details we refer to the article mentioned above. Further information can also be found in *J.Pfalzgraf: "On geometric and topological reasoning in robotics".* Annals of Mathematics and AI 19 (1997) 279-318 (special issue on AI and Symbolic Mathematical Computing), Jacques Calmet, John A. Campbell (Eds.).
J.Pfalzgraf, W.Meixl: "A logical approach to model concurrency in multiagent systems". Proceedings 15th European Meeting on Cybernetics and Systems Research (EMCSR'2000), April 25-28, 2000, Vienna. Austrian Society for Cybernetic Studies, R.Trappl (ed.).

8 On the Base Diagram of a Multiagent System

In this section we include a report on recent work dealing with the new notion of a Base Diagram which I introduced with the aim to establish a model of the general communication structure of a MAS.
It is published as the following article (with Thomas Soboll as invited co-author):
"The Base Diagram of a Multiagent System:
A Categorical Model of the General Communication Structure".
In: ADVANCES in MULTIAGENT SYSTEMS, ROBOTICS and CYBERNETICS: Theory and Practice. (Volume II)
Edited by: George E.LASKER and Jochen PFALZGRAF
Published by: The INTERNATIONAL INSTITUTE for ADVANCED STUDIES in SYSTEMS RESEARCH and CYBERNETICS
(Tecumseh, Canada), 2008. ISBN 978-1-897233-61-0

The Base Diagram of a Multiagent System: A Categorical Model of the General Communication Structure

Jochen Pfalzgraf and Thomas Soboll
Department of Computer Sciences, University of Salzburg
[jochen.pfalzgraf,thomas.soboll]@sbg.ac.at

Abstract

The categorical modeling approach for a multiagent system (MAS) provides a very general, generic definition of MAS. Every agent is an object and all kinds of relations between agents - including classical agent communication - are interpreted as corresponding morphisms (arrows). We call it general communication. This leads naturally to the introduction of an arrow diagram associated with a MAS - called Base Diagram - encoding the general communication structure of the MAS. In a natural way, every base diagram can be interpreted as a corresponding category. Some structure modeling aspects for base diagrams are briefly discussed.

Keywords: Multiagent Systems, Categorical Modeling

8.1 Introduction

In the course of the years it became well-known that Category Theory (CAT) is a powerful mathematical modeling language and the notion of a category (Cat) is abound in mathematics and various other disciplines. CAT provides far reaching universal construction principles having a wide area of already existing and potential future applications, especially in computer science and artificial intelligence, including the semantical foundations of topics in software science and development. CAT provides a unifying formal (mathematical) language with constructive elements, of increasing importance in mathematics, computer science and AI and further disciplines (not to forget logic).

In this short presentation we give a brief report on recent work comprising a very general, generic categorical definition of the notion of multiagent system (MAS) where agents are the (typed) objects and all kinds of relations and communication types between agents are modeled as (typed) morphisms - the arrows of that Cat.

Of basic importance for our approach is the observation that every arrow diagram can be interpreted as a Cat where the nodes are the objects and every sequence of consecutive arrows, i.e. every path of arrows (not only a single arrow) in the diagram is a morphism. Such a category is called PATH.

On the basis of these considerations we associate with every MAS a corresponding Base Diagram modeling the communication structure in a very general way - every arrow represents a particular type of general communication - we speak of “communication arrows”. Every Base Diagram is a category PATH.

Briefly summarizing, the notion of Base Diagram is a model representing characteristic structure of a given MAS. This categorical structure model is of general nature and offers the possibility to apply existing and new approaches from other areas, as shortly mentioned in this article.

8.2 The Category MAS

For the convenience of the reader, we briefly recall the definition of a category (Cat) - we refer to some excellent textbooks, cf. [AHS90], [Lan98], [LS00] - and the introduction of our very general, generic definition of a MAS as a category.
A category $\mathbf{A}$ consists of a class of *objects*, denoted by $A, B, C, \ldots \in \mathrm{Obj}(\mathbf{A})$ (the objects of $\mathbf{A}$). For each pair of objects A, B there is a set of *morphisms*, $\mathrm{Mor}(A, B)$ (the "arrows" between A and B). There is a *composition operation* on morphisms: if $f : A \to B$ and $g : B \to C$ are morphisms, then there is a morphism $g \circ f : A \to C$, the composition of f and g. In a category the following axioms hold.
The composition of morphisms is associative, that is $h \circ (g \circ f) = (h \circ g) \circ f$, for corresponding morphisms f, g, h. For every object A there is the *identity morphism* id_A with the properties $f \circ id_A = f$ and $id_B \circ f = f$, for all $f : A \to B$.

We underline that the **arrow notation** for morphisms is of basic importance. We shall use $f : A \to B$, as well as $A \xrightarrow{f} B$, to denote morphisms (arrows).
One can say that category theory discusses and formalizes the basic features of "everyday work" when dealing with spaces in a certain mathematical discipline and studying structure preserving functions (the morphisms) between spaces.

We recall very briefly our new general definition of a MAS. A multiagent system is introduced as a category (cf. [Pfa05], [Pfa04a] for a first step). The Objects are the (typed) Agents (<u>Types</u> of Agents are introduced, specified by corresponding properties). The Morphisms are all kinds of Relations between agents (<u>Types</u> of Relations are specified). Composition of morphisms is described in a natural way as subsequently introduced in the category PATH. Sometimes we call this general form of a morphism between agents a *communication arrow, communication (cooperation) line*, in the general sense, also *(generalized) communication.*
In an abstract sense we are dealing with arrow-diagrams of agents where the existing arrows represent communication and cooperation lines (channels) in a very general formal way. This categorical model of a MAS is generic.
An interesting introductory ("classical") textbook for MAS is [Woo02].

8.3 The Base Diagram of a MAS

An important prerequisite for the introduction of the notion of **Base Diagram** is the **Category PATH** - for the first time discussed in [Pfa94], later in [Pfa04a], cf. also [PS07]. Starting point is the observation that every reflexive, transitive binary relation R on a set X can be illustrated by an arrow diagram and can be interpreted as a Cat where the objects are the elements (points) of X and the morphisms (arrows) are given by pairs of elements being in relation. More precisely, for all $x, y \in X$ there is an arrow $x \to y$, iff xRy (i.e. $(x, y) \in R$). Identity arrows exist since R is reflexive and transitivity of R leads to composition of arrows. All axioms of a Cat obviously hold. This observation can be generalized in a very natural way to arbitrary binary relations and corresponding arrow diagrams as follows.
Let $R \subset X \times X$ denote a general relation. We associate with it the category denoted by PATH(X,R), PATH(X) or just **PATH**.
<u>Objects</u>: Elements $x \in X$.

Arrows, Morphisms: Sequences (paths) of consecutive arrows.
This defines *composition of arrows*, in a natural way (concatenation of consecutive arrows) and this composition is associative. The identity arrow id_x, with respect to each object $x \in X$, will be assumed to exist ("tacitly") by definition.
There is a morphism $x \longrightarrow y$, iff xRy. In general, for arrows $x \to y$ and $y \to z$, we do not have a "direct arrow" $x \to z$ (the relation can be not transitive) - this causes no problem. We can always form a sequence (path) of consecutive arrows, like $x \to y \to z$. This is a morphism of a more general type between x and z. More generally, we can have (finite) sequences, for example $\mathbf{x_0} \to \mathbf{x_1} \to \ldots \to \mathbf{x_n}$ (a path), this is a morphism in $\mathrm{Mor}(x_0, x_n)$ in the new sense of our definition. It can also be interpreted as the composition of other morphisms being represented by adjacent parts of the long sequence. Thus, **PATH** is a category. The existence of the identity arrow for each object will always ("tacitly, trivially") be assumed by definition. Graph-theoretically it is a loop in the corresponding node.
Observation: An arbitrary binary relation R on X induces a corresponding arrow diagram $\mathbb{D}$, "visualizing" the given relations between objects by corresponding arrows. Vice versa, a given arrow diagram $\mathbb{D}$ induces (or defines) a corresponding binary relation R on the set of elements (nodes) of $\mathbb{D}$ in the obvious, natural way, i.e. a specific arrow $x \longrightarrow y$ in $\mathbb{D}$ defines xRy. In these situations there is always an associated Cat **PATH**.
This observation suggested to the first author the idea to deploy and further develop methods from the extensive work of Hartmut Ehrig and his group on (algebraic) graph transformations (cf. e.g. [EEPT06]) - we make a corresponding remark below.

The following figure shows an illustration of an arrow diagram interpreted as a Cat.

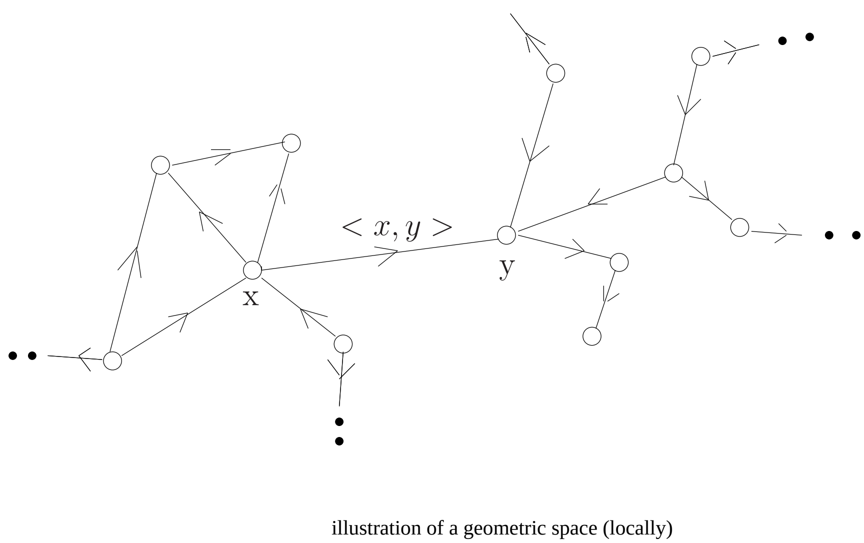


Figure 1: Arrow Diagram as a Cat (also: Geometric Net, locally)

The "abstract weight" $< x, y >$ on the edge from node x to y is the corresponding notion in use for noncommutative geometric spaces (and thus for GeoNet) denoting the direction, color of the line (the directed edge) joining x and y - it can also be interpreted as an abstract weight on the edge, this point of view is useful for MAS Base Diagram purposes (building a bridge to connectionist network structures). We point out that in the general case of an arrow diagram, usually there can be more than one arrow joining two points (nodes) x and y, in both directions. Depending on a particular modeling problem, we will consider corresponding arrow types.

Now we come to the definition of **Base Diagram** (as recently introduced by the first author). We recall that in our general Cat definition of a MAS the objects are the agents and the morphisms (arrows) are all kinds of relations (of specific type) between agents - represented ("visualized") by a corresponding arrow. This includes arrows representing **classical communication** relations like "KQML, KIF" and **specific comparisons** - agent to agent - concerning "capability, skill, power, degree of availability", etc., only to mention a few possibilities that can appear in real MAS modeling scenarios. We point out here that in our definition of agents human agents are included.

Considering and interpreting this, the **General Communication Structure** of a MAS is determined by the set of <u>all</u> **morphisms** (i.e. **"communication arrows"**) between agents, specifying corresponding **"communication types"**.

This leads to the following definition: The associated **Base Diagram** of a given MAS is a category **PATH** determined by the arrow diagram $\mathbb{D}$ where each node represents an agent (object) and where the arrows represent the morphisms between the agents - as previously described.

8.4 On Structure Modeling of Base Diagrams

The previously introduced notion of Base Diagram **PATH**($\mathbb{D}$) forms the formal fundament for the introduction and application of existing and new methods for structure modeling and system classification in MAS. Subsequently we mention a few ideas.

Base Diagrams and Graph Transformation Techniques. Based on the extensive work on (Algebraic) Graph Transformation of the Berlin group, cf. [EEPT06], we started own work with the idea to use such techniques to manipulate the base diagram of a MAS, first results are presented in [PS07].

General Network Aspects. As already mentioned, a useful observation is the fact that a base diagram is a network in the sense that every arrow (directed edge) has a type, i.e. a color, label or abstract weight in the sense of a general net (cf. Figure 1). This motivates to consider and deploy notions and methods from general net theory and especially artificial neural network structure theory - in former work we have introduced the Cat of Geometric Nets, **GeoNet**, cf. [Pfa04b]. Of particular interest, among others, is the **simplicial structure** of a geometric net coming from the notions of simplex and simplex configurations in topology and noncommutative geometry (cf. [Pfa04b], [Pfa87]). We point out here that these basic topological and geometric structure notions can be observed in a natural way in certain biological networks consisting of biological cell assemblies.

Concerning topological aspects, we quote an interesting remark by William C. Hoffman, Ph. D., Director of the Institute for Topological Psychology, Tucson, Arizona.

"Why Topological ?" : A thing is not just a thing – it has form and meaning. Form perception is inherently geometric – the things one sees consist, as they do, of geometric objects. And as for meaning, if one accepts the quaint fancy of the Connectionists that thought processes consist of point "neurons" and the paths connecting them, then simplicial topology enters as the lines, triangles, tetrahedrons, etc. that connect cognitive "chunks" by "trains of thought".

A further aspect of **intended future work** concerns construction principles from CAT like Limit and Co-lomit constructions that can be deployed, e.g. to extend a given MAS (using the Base Diagram) by a kind of universal communicator (coordinator) agent, cf. remarks in [Pfa06].
Simple scenarios of cooperating robot agents provided first motivating examples ("experiments") and first steps, cf. [Pfa97].

A further aspect concerns a MAS Semantics of "Commutativity of Arrow Diagrams", i.e. how can one exploit a commutativity condition in a given diagram with a view to "Redundancy of Communication Paths". Again, first steps can be found in [Pfa06].

The scenarios of cooperating robot agents were originally devised to demonstrate how the concept of a **Logical Fibering** can be used in a natural way to assign a system of **Distributed Logics** to a MAS, where every agent has an individual local logical state space (fiber), the collection of all the fibers forms the global state space (fiber bundle) of the MAS. Logical fiberings were originally introduced in [Pfa91].
This Fiber Bundle aspect can naturally be extended and generalized to introduce fibers of various structure types, modeling corresponding state space properties which are of relevance to model the complete state space of an agent, consisting of various modules (fibers) definig the complete type of an agent.
A first example in former work is the "MEDLAR Practical Reasoner" (cf. [Pfa94]) consisting of specific modules (local logic, kinematics, sensor evaluation, database, communication, etc.).
A proposal how to introduce an automated reasoning machine for agents and MAS - based on logical fiberings - is mentioned in [Pfa04c].

9 Some Remarks on "Algebra of Programming"

A very interesting and important book, obviously related to "Formal Methods in Software Engineering" aspects, is R.Bird and O. de Moor: Algebra of Programming, Prentice Hall 1997. Some information follows.

Editorial Reviews Book News, Inc.: Computer scientists from Oxford University describe an algebraic approach to calculating programs based on a categorical calculus of relations. They codify the basic laws of algorithmics and show how they can be used to classify many programs into families related by the algebraic properties of their specifications. An aid for software engineers to deriving individual programs or studying programming principles in general. – Copyright 1999 Book News, Inc., Portland, OR. All rights reserved.

From the back cover of the book: The purpose of this landmark text is to show how to calculate programs. Describing an algebraic approach based on a categorical calculus of relations, "Algebra of Programming" is suitable for the derivation of individual programs and the study of programming principles in general. The programming principles discussed are those paradigms and strategies of program construction that form the core of algorithm design. These include dynamic programming, greedy algorithms, exhaustive search, and divide-and-conquer. The fundamental concepts of the algebraic approach are illustrated by an extensive study of optimization problems. A wide variety of self-study exercises and bibliographical remarks reinforce the informative and educational nature of the text. Answers to most of the exercises can be obtained from a world-wide web site.

Critical Remarks (by J.P.): This book is certainly pointing to an important direction concerning formal aspects of software technology, in the spirit of: - towards (mathematical) software construction. Unfortunately, I see a serious drawback in the book. In chapter 2, on page 25, the authors introduce the definition of a category. They do NOT follow the standard way of introducing the notion of arrow (morphism) as it can be found in every good textbook on category theory. The shortcoming of their definition is that they introduce an arrow (morphism) in reversed direction, in contrast to the usual way (name of the arrow, f,:, domain A, rightarrow, codomain B). The reason why they do this is typed in parentheses "(pronounced 'f is of type A from B')". Actually, the textbook introduces the concept of "dual category". This fact should be pointed out to the reader with clear comments.

10 Concluding : Further Interdisciplinary Aspects

Subsequently, we include some material presented and discussed on our homepage under the title "On General Mathematical Modeling Aspects" (for more details we refer to http://www.cosy.sbg.ac.at/∼jpfalz/MATHmod.html).

Concerning mathematical modeling aspects, in the general interdisciplinary sense, the very interdisciplinary broad field of **Theoretical Neurobiology** seems to be an area of promising applications of ACCAT approaches.
NOTE: For a separate treatment of the subsequent remarks on this topic, cf. section "ACCAT and Theoretical Neurobiology".

In December 2002, I obtained an invitation to give a plenary lecture at: *International Conference on Theoretical Neurobiology, February 24-26, 2003, National Brain Research Centre, New Delhi, India.* In their invitation, the organizers made the following remark: "This conference provides a unique opportunity for interactions between eminent mathematicians working in the field of category theory and higher dimensional algebra, and computational and experimental neuroscientists."

It is interesting to note which topics were listed in *"Scope of the Conference"*:
Algebra, Geometry, and Logic of Cognition. Artificial and Natural Intelligence. Cognitive Neuroscience. Computational Neuroscience. Dynamical Systems Theory. Functional Imaging of Brain. Neural Correlates of Consciousness. Neural Networks. Neuroinformatics. Neuropsychiatric Disorders.

Subsequently, we quote from *"Aims of the Conference"* - these remarks perfectly fit the intention underlying this section on "General Methematical Modeling Aspects".

Aims of the Conference: The main aim of the conference is to facilitate development of theoretical tools and methodology required to further our understanding of the brain. Towards this end, the organizers invite papers discussing ...., in-depth examination of outstanding problems, and introduction of novel mathematical methods to elucidate the workings of the brain.

One of the main goals of the conference is to provide a platform for experimental and computational neuroscientists to closely interact and exchange ideas with mathematicians working in the areas of **category theory and higher dimensional algebra** and explore the potential of these sophisticated mathematical methods, in view of their success in solving problems hitherto intractable within the point set theoretic framework, to meet the demands of cognitive neuroscience data. These interactions will also inspire mathematicians to develop new formal tools and techniques tailor-made to suit the unique nature of the brain and thereby accelerate the development of a comprehensive theory of brain function that provides a scientific account of not only photons and action potentials but also of percept, thoughts, emotions, intention, and action.

I could not attend the New Delhi conference, but obtained an invitation to present a talk at the international conference "Computing Anticipatory Systems CASYS'2003, August 11-16, 2003, Liège, Belgium", with title: *Modeling connectionist networks: categorical, geometric aspects (towards "homomorphic learning").* Received a Best Paper Award. Published in: Proceedings CASYS'2003, American Institute of Physics, AIP Conference Proceedings, Vol.718 (2004), D.M.Dubois (Ed.).

NEWS: Gesellschaft für Informatik (GI).
Neue GI-Fachgruppe "Formale Methoden und Software Engineering für Sichere Systeme (FoMSESS)".
Am 22.3.2002 wurde im Fachbereich Sicherheit die Fachgruppe FoMSESS e.V. vom Fachbereichsrat gegründet. Zielsetzung dieser Fachgruppe ist es, in der Computer- und Informationssicherheit ein Diskussionsforum im deutschsprachigen Raum zu bieten, das sich mit der Grundlagenforschung und Anwendung formaler oder mathematisch präziser Techniken im Software-Engineering beschäftigt..... (more on my hompage).

In my opinion, it is very likely that CAT approaches are of theoretical and practical usefulness and importance for FoMSESS purposes in the sense of "towards CAT based software construction".

Mathematical modeling aspects with certain links to the previously mentioned topics can be found in the extensive area of work at the **Institute for Topological Psychology** Tucson, Arizona, William C. Hoffman, Ph. D., Director (click "Topology" on my homepage, cf. `http://home.att.net/ topologicalpsychology/index.htm`). It is fascinating to see how widely developed fields from Topology, Geometry lead to concrete, interdisciplinary modeling approaches. It can be observed that very abstract theories like Categories, Fibered Structures, Sheaves, Topoi are of increasing relevance in interdisciplinary generic mathematical modeling.

On the previously mentioned homepage one can read the following very illustrative and plausible introductory remark **'Why Topological?'**:
"A thing is not just a thing – it has form and meaning. Form perception is inherently geometric – the things one sees consist, as they do, of geometric objects. And as for meaning, if one accepts the quaint fancy of the Connectionists that **thought processes consist of point "neurons" and the paths connecting them**, then **simplicial topology** enters as the lines, triangles, tetrahedrons, etc. that connect cognitive "chunks" by "trains of thought"."

We note here, that this motivating explanation is closely related to a remark about the **role of simplices** in our **CAT and geometric modeling of neural network structures** (cf. section 5).

Another very interesting place where high level interdisciplinary work is going on and where **Categorical Modeling** plays a role, among others, is the group of Guerino MAZZOLA, Informatik Institut, Universität Zürich, Switzerland. There is a MultiMedia Laboratory dealing with **MusicMedia Science**, for detailed information we refer to the very interesting web site `www.encyclospace.org`. G.Mazzola is a leading scientist in **Mathematical Music Theory**, Computer Music and related Computer Science (**Music Informatics**). In 2002, his book "The Topos of Music - Geometric Logic of Concepts, Theory, and Performance" has been published. It is a real master piece, I highly recommend it (detailed information can be found on the previously mentioned homepage "EncycloSpace").

Concerning interdisciplinary mathematical aspects of systems modeling, say **"theory meets practice"**, in the general sense, I was impressed by the very extensive program of work of the mathematician Robert HERMANN who edited an own series of books **"Interdisciplinary Mathematics", MATH SCI PRESS** (53 Jordan Road, Brookline, MA. 02146), where he published and propagated his "vision"of the role of well established modern mathematical fields like **algebra, geometry, topology, category theory** for "generic" systems modeling in engineering sciences, artificial intelligence, and computing science. A major point of his view is the basic geometric (topological) nature of all kind of systems.

At the end of the homepage `http://www.cosy.sbg.ac.at/ jpfalz/MATHmod.html`), a facsimile is displayed taken from the book by R.HERMANN with the title "Geometric Computing Science: First Steps". It is volume 25 of his extensive series Interdisciplinary Mathematics. We selected pages 386 - 389 and the literature list of the last section of that book with title "REMARKS ON AI AND SHEAVES".

Subsequently, we quote some passages from the PREFACE of the book, where the author writes:
"This book outlines a mathematical way of thinking about the interaction of 'pure' and 'applied' computer science. I will be more subjective than is customary in science and mathematics exposition and sketch my personal vision, in terms of my own experience in pure and applied mathematics and my extensive readings in the scholarly computer science literature. ....

The first obvious questions are: **What is Computer Science ? What is Geometric Mathematics ?** What will it mean **to combine the two ?** I can not give complete answers here, ...

First, any serious attempt to combine the two disciplines will find that the **algebra of Category Theory** will be an essential intellectual tool, ...

My task would be easier if a contemporary Felix Klein of Computer Science had laid out his unifying vision. However, it seems that we are still at an early historical stage ....

Certainly, I see a central question:
What is the **Mathematical Nature of a Computer Programm ?** Perhaps this is analogous to the question - What is an Elementary Particle ? ....

A key mathematical question will be:
What is the right **categorical notion of 'data' ?**............"

References

[AHS90] J. Adámek, H. Herrlich, and G.E. Strecker. *Abstract and Concrete Categories.* John Wiley & Sons, 1990.

[EEPT06] H. Ehrig, K. Ehrig, U. Prange, and G. Taentzer. *Fundamentals of Algebraic Graph Transformation.* Springer Verlag, Monographs in Theoretical Computer Science, 2006.

[Lan98] S. Mac Lane. *Categories for the Working Mathematician.* Springer Verlag, Graduate Texts in Mathematics **5**, 2nd ed., 1998.

[LS00] F.W. Lawvere and S.H. Schanuel. *Conceptual Mathematics: A First Introduction to Categories.* Cambridge University Press, 2000.

[Pfa87] J. Pfalzgraf. A note on simplices as geometric configurations. *Archiv der Mathematik*, 49:134–140, 1987.

[Pfa91] J. Pfalzgraf. *Logical Fiberings and Polycontextural Systems.* 1991. In: Proc. Fundamentals of Artificial Intelligence Research, Jorrand Ph., Kelemen J. (Eds.). Lecture Notes in Computer Science 535 (subseries LNAI).

[Pfa94] J. Pfalzgraf. On a general notion of a hull. In *Automated Practical Reasoning, J.Pfalzgraf and D.Wang (eds.). Texts and Monographs in Symbolic Computation, Springer Verlag Wien, New York*, 1994.

[Pfa97] J. Pfalzgraf. On geometric and topological reasoning in robotics. *Annals of Mathematics and Artificial Intelligence, 19*, pages 279–318, 1997.

[Pfa04a] J. Pfalzgraf. Applied and computational category theory. In *Tutorial for 27th German Conference on Artificial Intelligence, KI-2004, September 2004, Univ. Ulm*, 2004.

[Pfa04b] J. Pfalzgraf. Modeling connectionist networks: Categorical, geometric aspects (towards "homomorphic learning"). In *Proceedings CASYS'2003, "Computing Anticipatory Systems: CASYS 2003". August 11-16, 2003, Liège, Belgium. American Institute of Physics, AIP Conference Proceedings, Vol.718, D.M.Dubois (Ed.). - Received a Best Paper Award*, 2004.

[Pfa04c] J. Pfalzgraf. On logical fiberings and automated deduction in many-valued logics using Gröbner bases. *RACSAM, Rev. Real. Acad. Ciencias, Ser. A. Mat., Vol. 98(1)*, pages 213–227, 2004.

[Pfa05] J. Pfalzgraf. On categorical and logical modeling in multiagent systems. In *Anticipative and Predictive Models in Systems Science, Vol.1, pp. 93-98. George E. Lasker and Daniel Dubois (Eds.), ISBN 1894613-49-X, published by the IIAS, Windsor, Ontario, Canada (2005)*, 2005.

[Pfa06] J. Pfalzgraf. On an Idea for Constructing Multiagent Systems (MAS) Scenarios. In *Advances in Multiagent Systems, Robotics and Cybernetics: Theory and Practice. (Volume I). Edited by George E. Lasker and Jochen Pfalzgraf. Published by: The INTERNATIONAL INSTITUTE for ADVANCED STUDIES in SYSTEMS RESEARCH and CYBERNETICS (Tecumseh, Canada). ISBN 1-897233-36-1*, 2006.

[PS07] J. Pfalzgraf and T. Soboll. On a general notion of transformation for multiagent systems. In *Proceedings of the Conference: Integrated Design and Process Technology, IDPT-2007, June 3-8, 2007, Antalya, Turkey. Society for Design and Process Science, ISSN 1090 - 9389*, 2007.

[Woo02] M.J. Wooldridge. *An Introduction to Multiagent Systems.* John Wiley and Sons LTD, 2002.