Knowledge Representation and the Embodied Mind: Towards a Philosophy and Technology of Personalized Informatics

Wolf-Tilo Balke[1] and Klaus Mainzer[2]

[1] L3S Research Center,
Universität Hannover, 30539 Hannover, Germany
balke@l3s.de, http://www.l3s.de/~balke
[2] Chair for Philosophy of Science, Institute of Interdisciplinary Informatics,
Universität Augsburg, 86135 Augsburg, Germany
klaus.mainzer@phil.uni-augsburg.de
http://www.informatik.uni-augsburg.de/I3/

Abstract. Knowledge representation has a long tradition in logic and philosophy. Automated reasoning with ontologies and categories had been discussed in philosophy, before it was formalized in artificial intelligence and e.g. applied in information systems. But, most of our knowledge is implicit and unconscious, situated and personalized. It is not formally represented, but embodied knowledge, which is learnt by doing, applied by self-organization, and understood by bodily interacting with (social) environments. In a complex world, we have to be able to act and decide with incomplete and fuzzy knowledge under the conditions of bounded rationality. The bounded rationality of embodied minds is a challenge of informatics especially in the complex information world of Internet applications and Web-based services offering access to a vast variety of information sources. It overcomes traditional concepts of mind-body dualism in the philosophy of mind, traditional knowledge representation in AI, and rational agents ("homo oeconomicus") in economics. Personalized informatics opens a trans-disciplinary perspective for philosophy and working technology.

1 Knowledge and Representation

Knowledge representation which is today used in database applications, artificial intelligence, software engineering, and many other disciplines of computer science has deep roots in logic and philosophy [15]. In the beginning, there was Aristotle (384-322 B.C.) who developed logic as a precise method for reasoning about knowledge. Syllogisms were introduced as formal patterns for representing special figures of logical deductions. According to Aristotle, the subject of ontology is the study of categories of things that exist or may exist in some domain. Aristotle distinguished ten basic categories for classifying anything that may be said or predicated about anything: Substance, quality, quantity, relation, activity, passivity, having, situatedness, spatiality, and temporality. Many of these categories are today applied in information systems, e.g. spatiality in location-based services. In the middle ages, knowledge representation was illustrated by graphic diagrams and pictures. In the 'summulae logicales' (1239) of Peter of Spain, an ontological hierarchy with

K.D. Althoff et al. (Eds.): WM 2005, LNAI 3782, pp. 586 – 597, 2005.

aristotelian categories represented knowledge by genus (supertype) and species (subtype). The features that distinguished different species of the same genus were called differentiae. Raimundus Lullus (13th century) illustrated an ontological hierarchy by a tree with branches for categories. Leaves corresponded to questions or to answers which should automatically be found by a system of rotating disks for combining features of things. Actually, Raimundus Lullus applied a kind of British Museum algorithm, the first attempt to develop mechanical aids for problem solving and information retrieval. Today, we use Entity-Relationship (ER)-diagrams in suitable forms to illustrate structures of ontologies in informatics.

In modern times, Descartes considered the human brain as a store of knowledge representation. Recognition was made possible by an isomorphic correspondance between internal geometrical representations (ideae) and external situations and events. Leibniz was deeply influenced by these traditions. In his 'mathesis universalis', he required a universal formal language (lingua universalis) to represent human thinking by calculation procedures and to implement them to mechanical calculating machines. An 'ars iudicandi' should allow every problem to be decided by an algorithm after representation in numeric symbols. An 'ars iveniendi' should enable users to seek and enumerate desired data and solutions of problems. Thus, in the age of mechanics, knowledge representation was reduced to mere mechanical calculation procedures. In Kant's epistemology, recognition is not only a passive mapping of the external world, but an active construction of internal representations by a priori categories of pure reason. In modern terms, categories are considered as tools which must be assumed before ('a priori') any application of knowledge representation. Cognitive constructivism roots back to Kant's epistemology. In the tradition of Brentano's and Husserl's phenomenology, Aristotelian ontologies had been again discussed for knowledge representation. Recognition needs intentional actions, which direct our awareness and consciousness to objects of the world. Thus, according to Husserl, understanding is not possible by symbolic representations of the external world, but by the intentionality of human consciousness [7]. Intentionality also became a prominent criterion to distinguish human consciousness and computer representation of knowledge in recent AI-debates [19].

Computational cognitivism arose on the background of Turing's theory of computability. In his functionalism, the hardware of a computer is related to the wetware of the human brain. The mind is understood as the software of a computer. Turing argues: If human mind is computable, it can be represented by a Turing program (Church's thesis) which can be computed by a universal Turing machine, i.e. technically by a general purpose computer. Even if people do not believe in Turing's strong AI-thesis, they often claim classical computational cognitivism in the following sense: Computational processes operate on symbolic representations referring to situations in the outside world. These formal representations should obey Tarski's correspondence theory of truth: Imagine a real world situation X1 (e.g. some boxes on a table) which is encoded by a symbolic representation A1 = encode(X1) (e.g. a description of the boxes on the table). If the symbolic representation A1 is decoded, then we get the real world situation X1 as its meaning, i.e. decode(A1) = X1. A real-world operation T (e.g. a manipulation of the boxes on the table by hand) should produce the same real-world result A2, whether performed in the real world or on the symbolic representation: decode(encode(T)(encode(X1))) = T(X1) = X2. Thus, there is an isomorphism between the outside situation and its formal representation in Cartesian tradition.

2 Self-organization and the Embodied Mind

Knowledge representations with ontologies, categories, frames, and scripts of expert systems work along the discussion in section 1. However, they are restricted to a specialized knowledge base without the vast, if somewhat unspecific background knowledge of a human expert. Human experts do not only rely on explicit (declarative) rule-based representations, but also on intuition and implicit (procedural) knowledge [6]. Moreover, as already Wittgenstein knew, our understanding depends on situations. The situatedness of representations is a severe problem of informatics. A robot, e.g., needs a complete symbolic representation of a situation, which has to be updated, if the robot's position is changed. Imagine that it surrounds a table with a ball and a cup on it. A formal representation of their respective relative positions in a computer language may be ON(TABLE, BALL), ON(TABLE, CUP), BEHIND(CUP, BALL), etc. Depending on the robot's position relative to the arrangement, the cup is sometimes behind the ball or not. So, unlike the representation ON(TABLE, BALL) the formal representation BEHIND(CUP, BALL) always has to be updated or at least checked in changing positions. How can the robot prevent incomplete knowledge? How can it distinguish between reality and its relative perspective? Situated agents like human beings do not need symbolic representations and constant updating. They look, talk, and interact bodily, e.g., by pointing to things. Even rational acting in sudden situations does not depend on internal representations and logical inferences, but on bodily interactions with a situation (e.g. looking, feeling, and reacting).

Thus, we distinguish formal and embodied acting in games with more or less similarity to real life: Chess for instance is a formal game with complete representations, precisely defined states, board positions, and formal operations. Soccer is a non-formal game with skills depending on bodily interactions, without complete representations of situations and operations, which are never exactly identical. According to Merleau-Ponty, intentional human skills do not need any internal representation, but they are trained, learnt, and embodied in an optimal 'gestalt', which cannot be repeated [16]. An athlete like a pole-vaulter cannot repeat his/her successful jump like a machine generating the same product times and again. Neither can athletes explicitly specify how they exactly achieved the result. Husserl's representational intentionality is replaced by embodied intentionality.

The embodied mind is no mystery. Modern biology, neural, and cognitive science give many insights into its origin during the evolution of life. The key-concept is self-organization of complex dynamical systems [13]. The emergence of order and structures in nature can be explained by the dynamics and attractors of complex systems. They result from collective patterns of interacting elements in the sense of many-bodies problems that cannot be reduced to the features of single elements in a complex system. Nonlinear interactions in multi-component ("complex") systems often have synergetic effects, which can neither be traced back to single causes, nor be forecasted in the long run or controlled in all details. The whole is more than the sum of its parts. This popular slogan for emergence is precisely correct in the sense of nonlinearity.

The mathematical formalism of complex dynamical systems is taken from statistical mechanics. If the external conditions of a system are changed by varying certain control parameters (e.g., temperature), the system may undergo a change in its macroscopic global states at some critical point. For instance, water as a complex system of molecules changes spontaneously from a liquid to a frozen state at a critical temperature of zero Celsius. In physics, those transformations of collective states are called phase transitions. Obviously they describe a change of self-organized behavior between the interacting elements of a complex system. The suitable macrovariables characterizing the change of global order are denoted as "order parameters". They can be determined by a linear-stability analysis. From a methodological point of view, the introduction of order parameters for modeling self-organization and the emergence of new structures is a huge reduction of complexity. The study of, perhaps, billions of equations, characterizing the behavior of the elements on the microlevel, is replaced by some few equations of order parameters, characterizing the macrodynamics of the whole system. Complex dynamical systems and their phase transitions deliver a successful formalism to model self-organization and emergence. The formalism does not depend on special, for example, physical laws, but must be appropriately interpreted for different applications.

Obviously, self-organization leads to the emergence of new phenomena on sequential levels of evolution. Nature has demonstrated that self-organization is necessary, in order to manage the increasing complexity on these evolutionary levels. But nonlinear dynamics can also generate chaotic behavior which cannot be predicted and controlled in the long run. In complex dynamical systems of organisms monitoring and controlling are realized on hierarchical levels. Thus, we must study the nonlinear dynamics of these systems in experimental situations, in order to find appropriate order parameters and to prevent undesired emergent behavior as possible attractors. The challenge of complex dynamical systems is 'controlled emergence'.

A key-application is the nonlinear dynamics of brains. Brains are neural systems which allow quick adaptation to changing situations during the life-time of an organism. In short: they can learn, assess and anticipate. The human brain is a complex system of neurons self-organizing in macroscopic patterns by neurochemical interactions. Perceptions, emotions, thoughts, and consciousness correspond to these neural patterns. Motor knowledge for instance is learnt in an unknown environment and stored implicitly in the distribution of synaptic weights of the neural nets. In the human organism, e.g. walking is a complex bodily self-organization, largely without central control of brain and consciousness: It is driven by the dynamical pattern of a steady periodic motion, the attractor of the motor system. Motor intelligence emerges without internal symbolic representations.

What can we learn from nature? In unknown environments, a better strategy is to define a low-level ontology, introduce redundancy – and there is a lot in the sensory systems, for example – and leave room for self-organization. Low-level ontologies of robots only specify systems like the body, sensory systems, motor systems, and the interactions among their components, which may be mechanical, electrical, electromagnetic, thermal etc. According to the complex systems approach, the components are characterized by certain microstates generating the macrodynamics of the whole system.

Take a legged robot. Its legs have joints that can assume different angles, and various forces can be applied to them. Depending on the angles and the forces, the robot will be in different positions and behave in different ways. Further, the legs have connections to one another and to other elements. If a six-legged robot lifts one of the legs, this changes the forces on all the other legs instantaneously, even though no explicit connection needs to be specified [18]. The connections are implicit: They are enforced through the environment, because of the robot‘s weight, the stiffness of its body, and the surface on which it stands. Although these connections are elementary, they are not explicit and could be easily included if the designer wished. Connections may exist between elementary components that we do not even realize. Electronic components may interact via electromagnetic fields that a designer is not aware of. These connections may generate adaptive patterns of behavior with high fitness degrees (order parameters). But they can also lead to sudden instability and chaotic behavior. In our example, communication between the legs of a robot can be implicit. In general, much more is implicit in a low-level specification than in a high-level ontology. In restricted simulated agents, only what is made explicit exists (cf. the closed world assumption in database applications), whereas in the complex real world, many forces exist and properties obtain, even if the designer does not explicitly represent them. Thus, we must study the nonlinear dynamics of these systems in experimental situations, in order to find appropriate order parameters and to prevent undesired emergent behavior as possible attractors.

But not only ‘low level’ motor intelligence, but also ‘high level’ cognition (e.g., categorization) can emerge from complex bodily interaction with an environment by sensory-motor coordination without internal symbolic representation. We call it ‘embodied cognition’: An infant learns to categorize objects and to build up concepts by touching, grasping, manipulating, feeling, tasting, hearing, and looking at things, and not by explicit representations. The categories are based on fuzzy patchworks of prototypes and may be improved and changed during life. We have an innate disposition to construct and apply conceptual schemes and tools (in the sense of Kant). Moreover, cognitive states of persons depend on emotions. We recognize emotional expressions of human faces (e.g. sadness) with pattern recognition of neural networks and react by generating appropriate facial expressions (e.g. concern) for non-verbal communication. Emotional states are generated in the limbic system of the brain which is connected with all sensory and motoric systems of the organism. All intentional actions start with an unconscious impulse in the limbic system, which can be measured half a second before the actions’ actual performance. Thus, embodied intentionality is a measurable feature of the brain [8]. Humans often use feelings to help them navigate the ontological trees of their concepts and preferences, to make decisions in the face of increasing combinatorial complexity: Emotions help to reduce complexity.

The embodied mind is obviously a complex dynamical system acting and reacting in dynamically changing situations. The emergence of cognitive and emotional states is made possible by brain dynamics which can be modeled by neural networks. According to the principle of computational equivalence [13, 14], any dynamical system can be simulated by an appropriate computational system. But, contrary to Turing’s AI-thesis, that does not mean computability in any case. In complex dynamical systems, the rules of locally interacting elements (e.g., Hebb’s rules of synaptic interaction) may be simple and can be programmed in a computer model. But their nonlinear dynamics can generate complex patterns and system states, which cannot be predicted in the long run without increasing loss of computability and information. Thus, artificial minds could [5] have their own intentionality, cognitive and emotional states that cannot be forecast and computed like in the case of natural minds. Limitations of computability are characteristic features of complex systems.

In a complex dynamical world, decision-making and acting is only possible under conditions of bounded rationality. Bounded rationality results from limitations on our knowledge, cognitive capabilities, and time. Our perceptions are selective, our knowledge of the real world is incomplete, our mental models are simplified, and our powers of deduction and inference are weak and fallible. Emotional and subconscious factors affect our behavior. Deliberation takes time and we must often make decisions before we are ready. Thus, knowledge representation must not be restricted to explicit declarations. Tacit background knowledge, change of emotional states, personal attitudes, and situations with increasing complexity are challenges of modeling information and communication systems. Personalized information systems in dynamic situations should be referred to ubiquitous and invisible computing of world-wide interactive media, in order to improve human-oriented information services and to support a sustainable information world.

3 Towards Advanced Personalization in Computer Systems

Especially for areas in computer science that rely on a user's expression of individual needs like e.g. query processing in databases and information systems, media retrieval in document collections or selection problems in Web services or e-commerce workflows, getting a precise account what the individual user wants or means is mission critical. In practical system implementations such information usually is deduced from a user's profile or some explicitly stated preferences. User modelling is concerned with trying to describe completely what part of the users' interests should influence a computer application. But since research in psychology shows that even in purposeful tasks users are usually not fully conscious of their exact wishes and needs, see e.g. [1], eliciting preferences directly from users is a difficult matter. It often needs a tedious process like the manual selection of services or areas of interest for personalization in publish/subscribe systems. Moreover, given that some knowledge is embodied the elicited information will be naturally incomplete and simply logging and storing and using user-stated keywords/behaviour will sometimes lead to counter-intuitive results. In order to raise a personalized system's performance in terms of relevance, a system thus has not only to focus on explicit user specification, but should also take information into account, that is specified by the user's implicit notions, situation or assumed common knowledge. This information can be gathered mainly from four sources:

- **long-term preferences:** The notion of relevance from previous interactions or generally applicable knowledge about a user is used
- **intention:** The specific user's purpose of the interaction is included in personalizing the system
- **situation:** The present state and environment of a user is used to decide whether specific preferences or rules are applicable
- **domain:** Knowledge on the specific domain (often referred to as expert knowledge) is used within an interaction

Let us consider typical instances of these kinds of personalization information. Among long-term preferences typical re-occurring individual preferences are collected e.g. individual tastes like colours, general areas of interest or preferred layout settings. Generally this kind of preference can always be used to personalize a system for individual users and is the usual kind often stored in user profiles. Systems, however, cannot always rely on these preferences, since they might be either further specified for certain categories or simply not applicable in a certain context. Consider for instance a set of colour preferences in an e-commerce setting. Though a user can be assumed to have a certain favourite colour that will apply to shopping decisions, the preferences might be different for e.g. clothing and cars, since driving an e.g. red car differs from actually wearing red clothing. Moreover, for e.g. book shopping the colour preference becomes entirely inapplicable, since the request to buy a red book is usually not sensible. A first framework for tagging and storing this kind of information can be found in [10].

Of a less general, but more interesting kind for personalization tasks are the preferences for the last three categories. Consider for example intentionality in a real world application like book shopping. Personalized book stores (e.g. Internet portals like www.amazon.com) will usually keep a list of recommendations based on the topics a user was interested in during previous interactions (i.e. a long-term profile of topical categories). Now assume that the same user accesses the book store with the intention of buying a present for some acquaintance. In most cases this present will focus on the preferences of the acquaintance; hence neither is the typical user profile applicable, nor should the interaction and the topics accessed be used to update the user's personal profile. In this example the intention of the user to buy a book for him-/herself or for a different person makes all the difference. Thus, the (assumed) intention of a user will help to decide, which choices a user should be offered in personalization and what characteristics a user generally cares about at a certain point in time during his/her interaction with the system. Typical examples for using these intentions in Web-based systems are also adaptive hypertext applications, where depending on a user's previous interactions and current navigation patterns the environment can be effectively personalized to support users, see e.g. [11].

Also the current situation has an impact on how to personalize a system. Context-aware systems have to use clues from a user's direct environment (like time or location), personal characteristics like emotional states, technical characteristics like client device capabilities, or certain high level situation information like "user in a business meeting" or "user at home" (both are examples for social situations) for personalization tasks. Examples for systems integrating this kind of information are location-based services, situation-based communication routing, or context-aware synthesis of multimedia content like discussed in e.g. [20].

The most renowned realizations of the last kind of preferences for personalization based on domain knowledge are the so-called expert systems that encode domain knowledge elicited from domain expert in a system usually be means of introducing rules for deduction. However, [6] shows that there cannot be a complete set of expert knowledge rules, since most expert knowledge is not represented by rules, but embodied in the experts themselves. Thus, we cannot simply consider domain preferences in the sense of expert systems, but have to rely on domain specific heuristics like which general preferences (and in what combination) might be applicable, or what users generally care about in a certain domain. The notion behind this is often referred to as 'common knowledge' or 'world knowledge', i.e. the knowledge about the environment that is assumed to be common to or implicitly shared by all humans interacting in a certain domain. Ontologies present a good way of representing some of this knowledge for use by non-experts of a domain.

In today's systems the latter three kinds of preferences – if at all – are mostly built directly into the application logic and represent the embodied mind as opposed to collected individual long-term preferences that form a user's individual profile. In the next section we will consider two sample scenarios and focus on how to use these kinds of preferences not in a hard-coded fashion, but flexibly mixed with information from user profiles.

4 Case Studies for Preference-Based Personalization

Let us consider two short application studies where we can see parts of the embodied mind represented by adequate preferences and used for improved system personalization. For effective personalization knowledge from all the four sources discussed above has to be blended with the specific user-provided details/keywords for an individual interaction. Though generally not all embodied knowledge can be captured that way, this method nevertheless provides a useful way of personalizing systems under the notion of bounded rationality.

First consider personalized information search and retrieval tasks in databases and information systems. The classical relational model that is still predominant in today's practical database applications uses relational algebra to specify rigid selection predicates that allow selecting objects with certain characteristics from usually large data sets. Though this model is applicable in a variety of simple cases, e.g. if customer information for a bank account with a certain account number has to be retrieved, modern information-driven environments demand for somewhat fuzzier capabilities in specifying what kind of information a user needs to accomplish his/her task. In most practical applications like e.g. Web search engines, information searches will lead to empty or too many results. If a user does ask a very specific query in necessarily fuzzy tasks like information searches, the query may be overspecified, e.g. by choosing too specific keywords, and lead to an empty result that will not be very helpful to users. On the other hand, asking rather unspecific (i.e. underspecified) queries is bound to lead to the flooding effect, i.e. lots of items that only more or less match a user's needs. However, not knowing the underlying database content or information collection users simply cannot be expected to know the degree of specificness their query needs to show to retrieve a helpful, yet manageable set of results.

Preferences that show the structure of interests for an individual user or ontologies modelling the common understanding of topics in a certain domain can help to tackle this problem. As stated in [2], expanding queries along user-specific preferences goes back to the area of cooperative answering, see e.g. [17] for an overview. The basic notion of cooperative retrieval systems is that they will relax the user-specified terms until a match in a collection of data can be found. Thus even an overspecified query will lead to some 'best efforts' results and avoid empty result sets together with the often necessary tedious manual refinement of queries. This way of dealing with query predicates as 'soft constraints', is also necessary for personalization tasks using individual preferences that have not been explicitly stated for a specific interaction. Since they have been implicitly assumed by the system as representations of common or embodied knowledge either from long-term profiles, intention, situation, or domain, they have to be considered on a lower level than the explicitly provided terms (i.e. as soft constraints that may refine too large result sets, but can be relaxed if empty result sets are retrieved). Recently [12] introduced a system of integrating preferences in the form of strict partial orders with a simple "I like A better than B" semantics into database queries. For example searching for a rental car a user could state that he/she likes a car with automatic transmission better than gear transmissions. If two offers are retrieved meeting all basic requirements, the result set can be ordered by or even limited to those objects fulfilling also the preference on transmissions. Such basic preferences on single aspects like a car's type of transmission can be modelled and combined into more complex queries using operators for deriving Pareto sets (i.e. all preferences are considered as equally important following the Pareto principle from economics), prioritized sets (i.e. a certain order is imposed on the preferences like e.g. lexicographic orders) and ranked result sets (i.e. preferences on numerical domains that are aggregated using suitable utility functions).

As a second example let us consider discovering useful Web services or selecting suitable services to construct complex workflows in a personalized manner like motivated in [3, 4]. Service-oriented application infrastructures are getting more and more common. In times of the ubiquitous Internet the Web service paradigm is expected to substantially alter the world of modern business processes. Essential components of this emerging service paradigm are Internet-based, modular applications that provide standard interfaces and communication protocols for efficient and effective service provisioning between different business units or businesses and customers. Especially the reusability of basic building blocks (or implementations) that are common in certain different workflows and easy customization within complex workflows are particularly appealing. But like in information searches, also here a retrieval model based on exact matches only is not likely to succeed. Users generally are rather interested in accomplishing high-level tasks and do care less about the exact intermediate steps. Thus, making them exactly specify all characteristics of the services needed instead of using a more fuzzy understanding of the workflow in question will be counterproductive. With the number and diversity of Web services expected to grow, enhanced techniques for service discovery and selection will be needed.

When designing a service like e.g. restaurant reservation or flight booking service providers already have quite specific ideas what capabilities the service should provide and what kinds of interaction to expect. Thus providers are domain experts that can provide a set of useful domain preferences and even ontologies for categorizations to foster successful execution/composition of services even for non-expert users. Moreover, providers also may anticipate different possibilities for usage of the service (possibly also in different situation scenarios). Generally in well-defined services only a certain number of typical requests/business processes will exist. These typical interactions for different users/groups also are preference patterns or usage patterns open for our personalization approach. A usage pattern may e.g. depend on the basic intentions of a significant group of users. Different intentions will need different patterns that reflect on both a user's profile stating his or her notion of a service's usefulness or desired characteristics (like execution costs, quality guarantees, etc.) and the service profiles that are employed to carry out the actual business task. Also here the basic method of relaxing demands in the case of empty result sets is necessary to support users in a cooperative fashion.

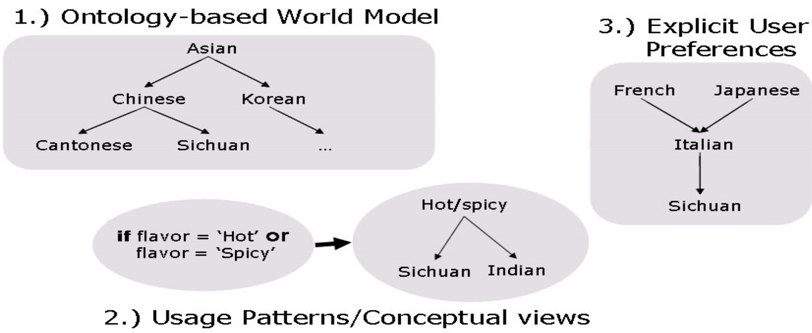

Fig. 1. Preferences derived from different sources

To show some examples for the different types of preferences and how to derive them consider a restaurant booking service. One of the parameters that is important in choosing a restaurant, is the type of cuisine. Every restaurant can be characterized by an adequate parameter, but querying for them is a tedious process, if it is not cooperatively supported. For example, if a user is interested to have lunch in a certain area asking for a restaurant offering Sichuan cuisine might be too specific and often deliver an empty result set. On the other hand just asking for a restaurant offering Asian cuisine might result in a large set of rather unspecific choices offering Chinese, Japanese, Indian, or Thai cuisine. One thing a cooperative system could use is an explicit preference (either explicitly specified with the current service request or derived from previous interactions of the same user). Such a preference could e.g. specify that a user prefers Sichuan cuisine over Italian and over French and Japanese (cf. Fig. 1).

But also in the case that such a preference is not explicitly given the service provider can assume some information. For example the general notion of similarity between cuisines could be made on the notion of geographical closeness reasoning that cuisines in similar geographic regions work with the same kind of ingredients, herbs and spices. A possible assumption would thus be that, since Sichuan is a specific Chinese cuisine, other Chinese cuisines like Cantonese could be acceptable offers for a user asking for Sichuan cuisine. To express this knowledge a domain-specific ontology like shown in Fig. 1 could be provided and in the case of empty results and no explicitly stated preferences be used to relax a user's request. As an example for user intentionality consider the usage pattern (or conceptual view) shown in Fig. 1. If we assume that a user is rather interested in hot and spicy food (i.e. a user has a rather taste-based conception of the similarity in restaurants that the geographical one presented above), some cuisines like Indian or Mexican are rather more similar to Sichuan cuisine (known for its spiciness) than the geographically close ones. Thus, if a user is known to subscribe to a specific conception as induced by his/her conception the more specific pattern has to be used for cooperative request relaxation.

5 Summary and Outlook

In this paper we focused on the representation of knowledge for personalization tasks in Informatics. Starting from the notion that most relevant information for personalization tasks cannot entirely be elicited as expert knowledge, but is embodied in the individual user (which is also consistent with current brain research), we propose to use flexible preference-based frameworks to personalize computer systems under the paradigm of bounded rationality. This means that though an electronic system cannot anticipate all possible influential factors, it can at least enrich user-related processes with some intentional, situational and domain-specific common knowledge.

As discussed for typical user interaction in the areas of personalized retrieval in databases/information systems and proactive Web service discovery/selection personalizing the interaction with preferences from each individual user's long-term profile, intention, situation, and domain will result in an improved effectiveness of the systems. This is because the user-provided information can be expanded with information representing the 'embodied' information necessary for a certain task. Since this information is not conscious, this expansion really adds value to the personalized task. However, since all preferences used for expanding the user information are only used on a lower level of importance than the explicitly provided information (and will be relaxed if necessary), the system's expansion will respect an individual users needs and never violate explicit constraints.

Obviously, personalized computer systems do not aim at complete computational models of the human embodied mind, which was an impractical illusion of traditional AI and the expert systems. In the current trend of modern informatics, we want to construct effective and appropriate tools and service systems under the conditions of bounded knowledge which need interdisciplinary cooperation especially with cognitive science and philosophy of mind.

Acknowledgements

Part of this work was supported by a grant of the German Research Foundation (DFG) within the Emmy Noether Program.

References

1. Ackhoff, R., Emery, F.: On Purposeful Systems. Aldine Atherton, Chicago (1972)
2. Balke, W.-T.: A Roadmap to Personalized Information Systems by Cognitive Expansion of Queries. Int. Workshop on Content-Knowledge Management and Mass Personalization of E-Services (EnCKompass 2002), Paris, France (2002)
3. Balke, W.-T., Wagner, M.: Cooperative Discovery for User-centered Web Service Provisioning. Int. Conf. on Web Services (ICWS'03), Las Vegas, USA (2003)
4. Balke, W.-T., Wagner, M.: Towards Personalized Selection of Web Services. Int. World Wide Web Conference (WWW'03), Budapest, Hungary (2003)
5. Dennett, C.D.: Brainchildren: Essays on Designing Minds, MIT Press1998
6. Dreyfus, H.L.: What Computer's can't do – The Limits of Artificial Intelligence. Harper & Row, New York, USA (1979)

7. Dreyfus, H.L.: Husserl, Intentionality, and Cognitive Science. MIT Press, Cambridge, USA (1982)
8. Freeman, W.J.: How and why brains create meaning from sensory information. Int. J. Bifurcation and Chaos 14 (2004) 515-530
9. Floridi, L. (ed.): Philosophy of Computing and Information. Blackwell, Oxford, UK (2004)
10. Holland, S., Kießling, W.: Situated Preferences and Preference Repositories for Personalized Database Applications. Int. Conf. on Conceptual Modeling (ER'04), Shanghai, China (2004)
11. Kaplan, C., Fenwick, J., Chen, J.: Adaptive Hypertext Navigation Based On User Goals and Context. Int. J. on User Modeling and User-Adapted Interaction. Vol. 3(3) (1993)
12. Kießling, W.: Foundations of Preferences in Database Systems. Int. Conf. on Very Large Data Bases (VLDB'02). Hong Kong, China (2002)
13. Mainzer, K.: Thinking in Complexity. The Computational Dynamics of Matter, Mind, and Mankind. 4th enlarged ed. Springer, New York, USA (2004)
14. Mainzer, K.: KI - Künstliche Intelligenz. Grundlagen intelligenter Systeme. Wissenschaftliche Buchgesellschaft, Darmstadt, Germany (2003)
15. Mainzer, K.: Computerphilosophie – Zur Einführung. Junius Verlag, Hamburg, Germany (2003)
16. Merleau-Ponty, M.: Phenomenology of Perception. Routledge & Kegan Paul (1962)
17. Minker, J.: An Overview of Cooperative Answering in Databases. Int. Conf. on Flexible Query Answering Systems (FQAS'98), Springer LNCS 1495, Roskilde, Denmark (1998)
18. Pfeifer, R., Scheier, C.: Understanding Intelligence. MIT Press, Cambridge, USA (1999)
19. Searle, J.R.: Intentionality. An Essay in the Philosophy of Mind. Cambridge University Press (1983)
20. Wagner, M., Balke, W.-T., Kießling, W.: An XML-based Multimedia Middleware for Mobile Online Auctions. In *Enterprise Information Systems III*, Kluwer, The Netherlands (2002)