
Context-based Retrieval for Explainable Reasoning

Stefan Schulz* — Thomas R. Roth-Berghofer**

* Carleton University, Dept. of Systems and Computer Engineering,
1125 Colonel By Drive, Ottawa, Ontario, K1S 5B6, Canada
schulz@sce.carleton.ca

** TU Kaiserslautern / German Research Center for Artificial Intelligence
DFKI GmbH, Knowledge Management Department, Erwin-Schrödinger-Str. 57,
67663 Kaiserslautern, Germany
trb@dfki.uni-kl.de

ABSTRACT: On the background of mobile, ubiquitous, and pervasive applications, context determination and assignment is a necessary factor to provide IT solutions suited to a user and the user's current situation. In this paper, context is seen as n-ary relationship. Context gets embedded into ontologies, which are used to structure application specific knowledge.

We present an integrative, case-based modelling approach for context and context management. We discuss the incorporation of context-based reasoning and explanation. And finally, we show how to apply our approach for trust management.

RÉSUMÉ: Dans le cadre des applications mobiles, diffuses et omniprésentes, la détermination et l'explicitation du contexte est un facteur nécessaire pour fournir des solutions IT personnalisées à un utilisateur donné dans une situation donnée. Dans ce papier, le contexte est considéré comme une relation n-aire. Le contexte est supposé contenu dans des ontologies qui sont utilisées pour structurer les connaissances spécifiques des applications. Nous présentons une approche basée sur une modélisation de cas pour exprimer le contexte et sa gestion. Nous discutons l'incorporation d'un raisonnement et la génération d'explications basés sur le contexte. Nous montrons finalement comment appliquer notre approche dans la gestion de la confiance.

KEY WORDS: context, context-based reasoning, case-based reasoning, ontology, trust management, explanation

MOTS-CLÉS: Contexte, Raisonnement à base de cas, Raisonnement basé sur le contexte, Ontologie, Gestion de la confiance, Explication

1. Motivation

A core issue for managing and sharing knowledge is to decide on relevance, applicability, and reliability of pieces of information. This is especially important, as several decisions base on these aspects, e.g., the choice of policies and strategies on dissemination and reception, the evaluation of exchanged information, the categorization of a piece of information, and activities to start upon an information's content. These decisions are inseparably interwoven with the aspects of context, as the decision factors are given by the current contextual information: the topic of the piece of information, the exchange partners and their relationship, the physical and the logical location, the current activity and emotional state of each partner, and so forth.

As the determination of context generally is subject to uncertainty, models and methods are needed, which support a soft matching of a present situation to possible decisions or decision factors. We propose a modelling approach to integrate contexts and contextual knowledge into application knowledge. Hence, reasoning on context results in the task to retrieve a matching persistent context by a given transient context, where transience refers to context information available at a single point of time, while persistence stands for a (memorized) recurring pattern of context (Lei et al., 2002). Based on the transient context, it will be possible to make decisions for a subject in that context based on experience of similar past situations.

Accepting a decision usually is a matter of understanding the reasons that led to it. In addition to matching a context, the models and methods must be capable of providing explanatory information. This information will explicate how and why a system has come to a specific decision.

Context and related decision factors build a basis for the aspect of trust. In other words, trust can be understood as a valuation of given contextual information. Determining the context therefore is essential to be able to decide upon the trust to put in a subject. Applying trust allows a soft valuation of knowledge regarding its relevance and reliability depending on the context the knowledge is retrieved or will be used in. Furthermore, as trust always is a personal valuation of someone or something, it is a necessary prerequisite for a person to understand the decision on trust made by a retrieval system. Thus, it is necessary to be able to explain the decision on trust.

The following sections reference current approaches in context modelling and context-based reasoning. In Section 2, the integrative modelling approach for context is described and how matching persistent contexts will be retrieved. Section 3 discusses the incorporation of context-based retrieval and explanations. Finally, Section 4 shows how to apply our approach for determining and managing trust and what the explanations accomplish.

1.1. Context Modelling

Generally, context is defined as “the interrelated conditions in which something exists or occurs” (Merriam-Webster, Online Edition). Transferred to exchange and management of knowledge in the information technology sense, context can be seen as a multi-dimensional space, which by nature cannot be determined in its whole. In (Lenat, 1998), (Pascoe, 2001), (Schmidt, 2002), and (Mitchell, 2002) various approaches are given that try to reduce context to a “most useful selection” of aspects of the environment, e.g., time, place, topic, granularity, trustability, cultural information, provability, social information, and availability.

Early models for context representation come from Artificial Intelligence (Brézillon, 1999). Here, contexts are mostly represented in a logical notation. Functions and predicates are given for each of the aspects (dimensions) of the environment. This results in a rule system having stand-alone context objects with concrete characteristics in their aspects. Hence, abstract and detailed requests for matching contextual information are possible. But it will be difficult to compare contexts or retrieve similar or related contexts. In particular, the model in (Bagci et al., 2003) describes context as named, aggregated structure, where simple contexts consist of a descriptor-state-pair and complex contexts consist of aggregations of simple and complex contexts.

In contrast to above, our approach follows an integrative and open way of modelling context, which does not limit or specify its dimensions a-priori. The structure and semantic will be given in the design phase of an application in conjunction with modelling application knowledge. Using these context structures as types for context representation allows for efficient matching of contexts.

1.2. Context-based Reasoning

Coming from the domain of intelligent agents and autonomous systems, (Gonzalez and Ahlers, 1998) introduces Context-Based Reasoning (CxBR) as a paradigm simulating human behaviour to perform tactical tasks. Its main premise is that humans make decisions based on a subset of available knowledge and context information. CxBR incorporates transitions between contexts to determine actions to be performed. These transitions correspond to matching transient and persistent contexts. In (Norlander, 1999), the mechanism used for transitions is realized by rules. A revised specification of CxBR is described in (Stensrud et al., 2004), which incorporates new aspects on learning by replacing or extending the rule-based transition logic with neural networks.

Also coming from the intelligent agents' domain is the approach on Context-Mediated Behavior (CMB) by (Turner, 1997). Knowledge about context is represented as contextual schemas (*c-schemas*), which contain persistent contexts related

behavioural information for the agent. Decisions about actions are taken by matching a situation to a c-schema.

The retrieval mechanism used in our approach can be seen as structured, learning transition logic for CxBR. In contrast to neural networks, knowledge on retrieving and matching contexts gets explicitly modelled in a human understandable way. By matching context structures, the mechanism allows a fuzzy and soft matching in opposite to complex rules. Our approach is closely related to the work on CMB even regarding the way of retrieving matching contexts. However, no proprietary model is needed to represent contexts and our approach makes use of relations in the application knowledge model to enable soft retrieval.

2. Integrative Context Modelling

Separating context and application knowledge in IT applications has only grown reasons. Human knowledge processes do not explicitly differ between kinds of knowledge. The same mechanisms and processes are used to store and handle knowledge of either kind. Depending on the needs, knowledge may switch roles. It may be once used as explicit knowledge for problem solving and another time as contextual knowledge.

Integrative context modeling refers to the likewise use of mechanisms and representations as being used for general application knowledge modeling. Further, it integrates application and context knowledge. Hence, using ontologies for representing structures in knowledge can also be used for representing context. The aspects of a context will be taken from the pool of application knowledge or extends this pool, respectively.

2.1. Context as Relationship

Relationships have the actual leading role in an ontology. Only with the use of relationships a simple collection of terms transforms into a structured, semantic net. Context intrinsically is a semantic element: A specific selection of information objects describes a particular context. The aspects of a context can be mapped to determined roles of the associated information objects. A context relation describes a structured scope in which the information object is valid (or applicable).

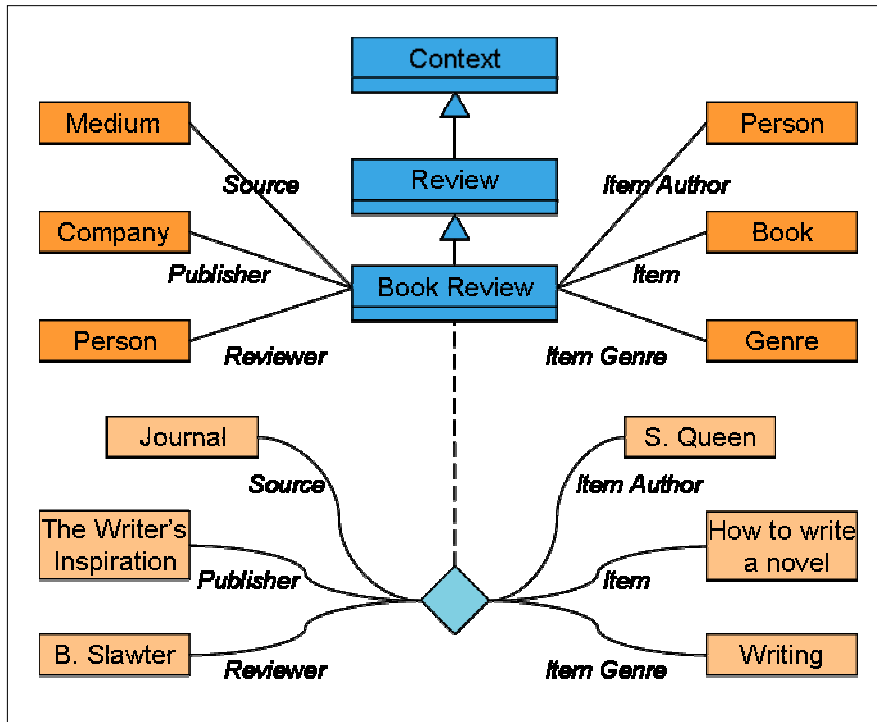


Figure 1. Simplified book review context

Figure 1 depicts a relationship for defining and instantiating a book review context. Here, six aspects describe a context type *Book Review* with the related aspect roles *Reviewer*, *Publisher*, *Source*, *Item*, *Item Author*, and *Item Genre*. The relationship type is a specialization of the context type *Review*, which again is a specialization of a generic *Context* type. To instantiate a *Book Review*, the n-ary relation between information objects gets typed and bound to information objects of the aspects type.

2.2. Advantages of Integrative Modelling

The advantage in comparison with existing, mostly application specific models is a flexible but structured possibility to provide context for context-aware systems. For the Topic Maps standard (TopicMaps, 1999), the model gives a more precise semantics for context and an extension of the integrated scope approach (Pepper and Grønmo, 2001). Regarding the Semantic Web (Berners-Lee et al., 2001), the W3C

standards RDF, RDF Schema¹, and OWL² can be used. Here, context relations must be defined first. There is no built-in scoping. Necessary contextual information then can be inferred using the same mechanisms as for the application knowledge.

Embedding context in application knowledge makes modeling and management of context and contextual knowledge easier to understand and to maintain. Further, by scoping based on contexts, views can be defined in the knowledge model (and hence the context model) to define the relevant visible sub-ontology including context itself. Of course, tool support is required to handle modelling and maintenance of the different contexts/views.

With the characteristics of a relationship it is possible to define the properties of specific contexts on the level of knowledge modeling. These are, among others, relevant domains for a context and their cardinality, dependencies between concrete aspects and the context, and the information objects having the scope of this context. Especially because of dependency and scope modeling, fragmentation of the knowledge space, e.g., for mobile applications (Schwotzer and Geihs, 2003), will enclose a context at its whole or not at all.

Regarding modeling, contexts can also act as aspects for more complex contexts. This way, historic or temporal dependencies regarding a sequence of contexts can be modeled. This may be essential for decision making, as different chains of actions may lead to different decisions, also if the final context is the same.

Another important aspect of integrating context into the application knowledge is the conjoint use of management tools and environments. This opens a variety of possibilities not only for defining a model but also to maintain and retrieve context.

2.3. Context Retrieval

By conjointly modelling context knowledge and application knowledge, a basis is given for retrieving context analogously to other knowledge. Basically, then, it is possible to recognize identical contexts: two contexts are equal, if they have the same type and if the aspect specific roles each are bound to the same information objects.

Moreover, the aspect of fuzzy reasoning is of much more interest when applying context. (Bergmann and Schaaf, 2003) describes the close relationship between Ontology-based Knowledge Management and Structural Case-Based Reasoning. Case-Based Reasoning (CBR) shortly can be described as follows: In order to solve a particular problem, look for a similar problem experienced in the past, take the solution from the past problem and try to reuse it to solve the actual problem.

¹ Resource Description Framework: <http://www.w3.org/RDF/>

² Web Ontology Language: <http://www.w3.org/2004/OWL/>

Applying Structural CBR to reason on context allows to find more detailed contexts to an incomplete defined context request, and vice versa to find more generalized contexts for over-specified requests. As a transient context in most cases will not have an exact equivalent in their aspects, the application of fuzzy reasoning will allow finding the best matching persistent context.

In (Mikalsen and Kofod-Petersen, 2005), the authors describe the use of CBR for retrieving the matching persistent context for a transient one. Their approach still has a separate model for context and contextual knowledge based on a context-taxonomy. In a way, this taxonomy can be seen as the definition of an application specific type structure for context, providing a context hierarchy for, e.g., task, physiology, mental, and spatio-temporal contexts.

2.4. Context Management by Case-base Maintenance

In the scenario described above, the body of contextual knowledge (or the context base) resembles a case base. Thus, managing contexts, i.e., keeping the context base up-to-date and consistent, can be understood as a maintenance task, where maintenance, according to (Groenendijk, 1996), is the combination of all technical and associated administrative actions intended to retain an item, or restore it to, a state in which it can perform its required function. In this case, the item is the context base.

Aamodt and Plaza (1994) developed the most influential, widely acknowledged process model that operationalizes the informal description given above. Their dynamic model identifies four sub-processes (also called steps): *Retrieve*, *Reuse*, *Revise*, and *Retain*. In the first step, the case-based reasoner retrieves the most similar case (or cases) that fits the current situation. Then, the system reuses the information and knowledge in that case and proposes a solution (suggested solution). If the solution is rejected the CBR system revises the proposed solution, and suggests a further revised case until it is accepted (confirmed solution). During *Retain*, the confirmed solution is stored in the case base. The four steps describe the general tasks in a case-based reasoner. They provide a global external view to what is happening in the system.

In order to include maintenance processes, (Reinartz et al., 2001) enhanced the original process model with two new steps: *Review* and *Restore*. The *Review* step considers the current state of a CBR system, checks its quality, and invokes the *Restore* step if necessary. The *Restore* step actually changes the contents of the CBR system to get it back to a desired level of quality.

In (Roth-Berghofer, 2003), a methodology for maintaining the knowledge of CBR systems is described. The SIAM methodology enhances the INRECA methodology for developing industrial CBR systems (Bergmann et al., 2003). The SIAM methodology employs Event-Condition-Action (ECA) rules, which are rooted in

multiple disciplines of database management systems research (Reinert and Ritter, 1998), to manage the maintenance tasks. In order to maintain the knowledge of a context-based system, the respective system must be analyzed to find triggers that change the system, leading from the application phase, where the application knowledge remains unchanged, to the maintenance phase. The transition is triggered, on one hand, by learning a new context (depicted by the arrow from Revise to Retain, in Figure 2) and, on the other hand, by manually adding new knowledge. Then, an appropriate maintenance manual must be developed to handle the changes appropriately.

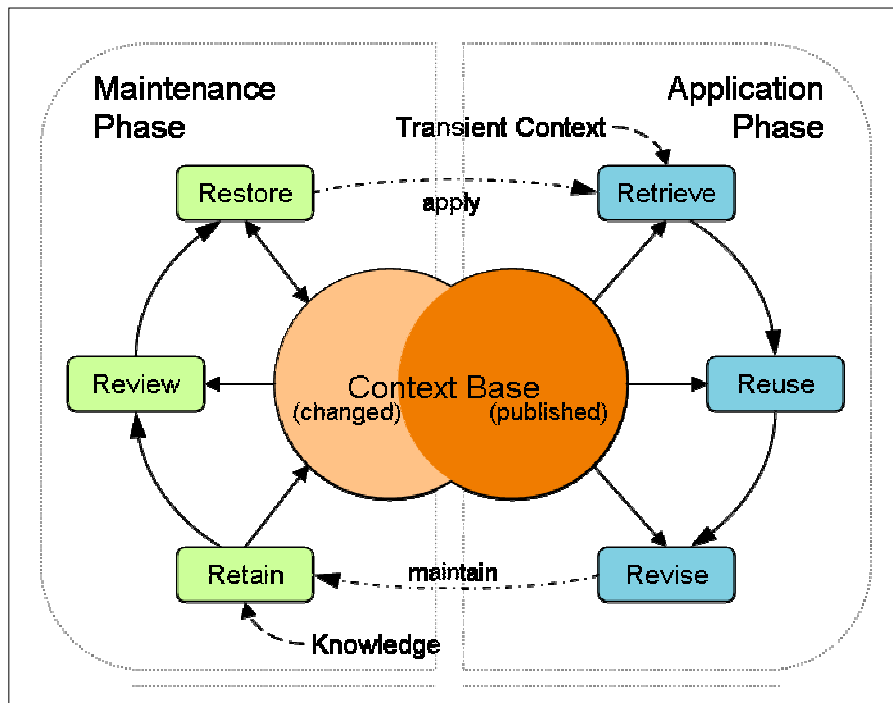


Figure 2. Six steps process model of Case-Based Reasoning (adapted from (Reinartz et al., 2001))

3. Context and Explanation

As soon as contextual information is available and a decision support system uses the context information for its reasoning process, the need arises for explaining system answers to the user with respect to the contextual information. The user is interested in how much trust he or she can have in a system. An obvious approach to increasing the confidence in a system's result is to output explanations as part of the result (Moore and Swartout, 1988). Belief in a system can be increased not only by the quality of its output but, more importantly, by evidence of how it was derived (Swartout, 1983). The user wants to have a sense of control over the system (Swartout and Moore, 1993).

Explanations are studied in depth in Philosophy of Science. There, scientific explanations are answers to why-questions (Schurz, 1993): "Can some fact E (explanandum) be derived from other facts A with help of general laws (explanans, $L \cup A$)?".

3.1. Requirements on Explanations

Expert Systems research early on operationalized explanations, at a time, when context was not considered explicitly. Now, we can hope to develop more relevant explanations. Explanation generation always involves communication with the user who is part of the context of the system. The respective system must make assumptions on what the user already knows and where he or she lacks knowledge.

Nevertheless, Expert Systems research derived valuable guidelines on what makes an explanation good. Five aspects of good explanation in a knowledge-based system are deemed important and fall into three classes (Swartout and Moore, 1993). The first requirement is concerned with how the explanations are generated. The second and third are requirements on the explanations themselves. The fourth and fifth both concern the effect of an explanation facility on the construction and execution of a knowledge-based system.

Fidelity. The explanation must be an accurate representation of what the knowledge-based system does. Therefore, the explanations must be based on the same knowledge that the system uses for reasoning.

Understandability. The generated explanations must be understandable, i.e., conceptually as well as regarding their content. This involves factors such as terminology, user sensitivity, abstraction, and summarization. Swartout and Moore further identified the factors perspectives, feedback, and linguistic competence. The system should be able to explain its knowledge from different perspectives and should allow for follow-up questions if the user indicates that he or she does not understand (part of) an explanation. The explanations should sound 'natural' and adhere to linguistic principles and constraints.

Sufficiency. The system has to 'know' what it is talking about. Enough knowledge must be represented in the system to answer the questions users have. Explanation knowledge cannot be derived from problem solving knowledge. Knowledge acquisition, therefore, must be extended to also address explanation needs. Asking questions on what the system should explain later on when it is deployed, of course, helps to acquire problem solving knowledge more completely.

Low construction overhead. Explanation must either impose a light load on the construction of a knowledge-based system, or any load that is imposed should be rewarded, for example, by easing some other phase of the knowledge-based system's life cycle. The question of what should be explained by a knowledge-based system is surely domain and application specific. If the users can be provided with explanations beforehand by training or easy to grasp documentation one should probably not add complex explanation capabilities.

Efficiency. The explanation facility should not degrade the run time efficiency of the knowledge-based system.

3.2. Kinds of Explanations

But what kinds of explanations are of interest? According to (Spieker, 1991), there are five useful kinds of explanations in the context of expert systems:

Conceptual Explanations. Conceptual Explanations are of the form 'What is . . . ?' or 'What is the meaning of . . . ?'. The goal of this kind of explanation is to build links between unknown and known concepts. Conceptual explanations can take different forms:

– Definition: "What is a bicycle" "A bicycle is a land vehicle with two wheels in line. Pedal cycles are powered by a seated human rider. A bicycle is a form of human powered vehicle."

– Theoretical proposition: "What is force?" "Force is Mass times Acceleration."

– Prototypical examples: "What is a bicycle?" "The thing, the man there crashed with."

– Functional description: "What is a bicycle?" "A bicycle serves as a means of transport."

Why-explanations. Why-explanations provide causes or justifications for facts or the occurrence of events. Again, one has to clearly distinguish between causes and justifications. Whereas the first concept is causal in nature and not symmetric, the latter only provides evidence for what has been asked for. For example:

– Justification: "Why does the universe expand?" "Because we can observe a red shift of the light emitted by other galaxies."

– Cause: "Why does the universe expand?" "Because the whole matter was concentrated at one point of the universe and because the whole matter moves away from each other."

How-explanations. How-explanations are a special case of why-explanations, describing processes that lead to an event by providing a causal chain. How-questions ask for an explanation of the function of a device, for example: "How does a combustion engine work?" "A combustion engine is an engine that operates by burning its fuel."

Purpose-explanations. The goal of Purpose-explanations is to describe the purpose of a fact or object. Typical questions are of the form "What is ... for?" or "What is the purpose of ?", for example: "What is the valve for?" "The valve is used to seal the intake and exhaust ports."

Cognitive Explanations. Cognitive Explanations explain or predict the behaviour of "intelligent systems" on the basis of known goals, beliefs, constraints, and rationality assumptions. They are also a special case of why-explanations, distinguishing action and negative explanations:

– Action explanation: "Why was this seat post selected?" "For the given price, only one other seat post for this bicycle was available. But that seat post was too short."

– Negative explanation: "Why was no carrier chosen?" "A carrier is only available for touring bikes. The user did not choose a touring bike."

The first four categories of explanations describe variations of scientific explanations, which answer questions based on laws of nature, thus explaining the physical world. Expert systems answer such questions by using the knowledge contained in their (static) knowledge base. Cognitive explanations, on the other hand, reflect a system-related view. They deal with the processing of the system. In a way, cognitive explanations explain the social world and individual patterns of behaviour.

4. Trust Management with Context-Based Retrieval and Explanation

With the ability to retrieve matching contexts even for over- or under-specified transient contexts, a base of persistent contexts acts as a source for reasoning. The proposition is that in similar situations one would arrive at similar decisions. Wrong decisions will result in an adaptation of the decision with respect to the context.

For that reason, context can act as a carrier for representing the circumstances for a decision, which allows reasoning based on contextual knowledge. More than that, reasoning on context builds a foundation for providing an enhanced adaptability of solutions.

4.1. Definition of Trust

Trust has been studied as the valuation of a relationship between individuals. In most cases an individual as such is regarded as being trustworthy. The early psychological definition of trust by Morten Deutsch (Deutsch, 1962) has been brought to the point in (Rousseau et al., 1998) stating that “Trust is a psychological state comprising the intention to accept vulnerability based on positive expectations of the intentions or behaviour of another.”

An approach to formalize trust is given by Marsh (Marsh, 1994) who identified three aspects of trust: *basic trust* defining the general willingness of an individual to trust into someone; *general trust* defining the trust of an individual into a specific person; and *situational trust* defining the trust of an individual regarding the current situation.

4.2. Context as Carrier for Trust Values

In the context-based approach described in the previous chapters, these aspects of trust are incorporated by the specification of a persistent context, i.e., a situation of basic trust would be reflected by a context with no aspects and for general trust the context would have a person as the only aspect. Usually, at least some contextual information is available by the time of decision making. Hence, the basis for a decision on trust becomes broader by having persistent contexts that vary in their aspects as well as in the degree of details.

Following the definition of psychologists above, the more trustful a person is, the more he or she will risk trusting into new situations and persons. Generally, the decision on trust is made upon experiences, i.e., comparable past situations. In the integrative approach, persistent contexts will provide a carrier for information on trust. Retrieving matching or *similar* persistent contexts to a transient context provides a basis for decisions on trust for a current situation.

4.3. Computing Trust

The value of trust usually is represented as a range of numeric or symbolic values ranging from distrust to trust. For example, in (Golbeck et al., 2003) trust is defined using nine levels from absolutely distrust via neutrally trust to absolutely trust in certain people and, more fine grained, in certain people but only with respect to particular topics. In (Schulz et al., 2003) trust is defined as a continuous interval from -1 to 1, reflecting the same intension as the level approach.

Computing a value of trust becomes necessary, if for a subject no definition is given. For example, when meeting someone never met before, a kind of initial trust is used (the basic trust as defined in (Marsh, 1994)). If one has heard of that person

from known individuals, the trust put into the person is derived from the knowledge gathered on him or her. One such approach is described in (Golbeck *et al.*, 2003) as Friend-of-a-Friend³ scenario. Trust is computed according to the path length in the social network built by the relationships of individuals among one another. The resulting value of trust computes either as maximum, minimum, or average path length in the network. (Richardson *et al.*, 2003) modelled trust by using a path algebra that allows the computation of trust values people have in particular statements. They evaluated their approach successfully in the domain of product recommendation.

A generalization of this approach towards Context-based Reasoning results in computing “trust distances” from a transient context to known persistent contexts. A trust distance Θ can be understood as the similarity of a transient context C_t and persistent context C_p combined with the trust in C_p , with the following definition:

$$\Theta(C_t, C_p) = f(\sigma(C_t, C_p), \tau(C_p))$$

Where σ computes the similarity of two contexts and τ determines the trust associated with the context.

For example, let the value range for σ be $[0,1]$ and let f be the multiplication operation. Then, the resulting trust distance is the trust for a known context C_p factorized by the similarity value. Note, that σ and Θ generally are not symmetrical.

Now, let Ξ be the set of all persistent contexts of the context base. Consequently, the value of trust in C_t computes to:

$$\tau(C_t) = g(\{\Theta(C_t, C_p) \mid C_p \in \Xi\})$$

Examples for g are minimum, maximum, and average, which corresponds to the approach for path length computation.

4.4. Context-based Trust Management

The notion of trust management is mostly seen in the domain of security and access policy in distributed systems (Blaze *et al.*, 1996). Here, trust is seen as “a unified approach to specifying and interpreting security policies, credentials, relationships which allow direct authorisation of security-critical actions”.

Recently, trust also gains interest in the more abstract domain of decision making. An approach on sharing knowledge based on trust is given in (Schulz *et al.*, 2003). And in (Grandison and Sloman, 2003), trust management in the area of electronic commerce is defined as “the activity of collecting, codifying, analyzing and evaluating evidence relating to competence, honesty, security or dependability with

³ <http://www.foaf-project.org/>

the purpose of making assessments and decisions regarding trust relationships for Internet applications”.

In short, trust management is the maintenance of subject related evidence and associated decisions on trust. In this paper’s approach, evidence is stored in form of persistent contexts, which describe the circumstances that led to the linked value of trust. Hence, the same tools and mechanisms can be used for maintaining evidence (i.e., context) as is used for the knowledge.

4.5. Explaining trust

The computation of trust as a distance of persistent contexts provides not only a mechanism for retrieving appropriate trust values, but also a source for explaining it. A common way to explain a decision is to present the decision factors and comparable decisions from the past. Both is at hand with using context as carrier for trust. In the following, we describe, how the requirements for good explanations cohere with our approach.

Fidelity. In our case, the explanation must show how a certain trust value was derived and where knowledge came from. The issue of knowledge provenance (e.g., recently addressed in (McGuinness and Pinheiro da Silva, 2003) and (Fox and Huang, 2003)) plays an important role in building up trust.

Understandability. Looking at the path algebra interpretation of trust, described in (Richardson et al., 2003), which uses several mathematical functions applied to matrices, and the generalized approach, described in Section 5.3., which raises the question of complexity by using similarity and trust for computing distances, shows the difficulty of providing understandable explanations of trust calculation. And, because trust is part of a context, trust must be explained with respect to the context.

Sufficiency. The system has to explain, how a trust value was retrieved. Therefore it needs to know about the computing algorithm and how each aspect contributes to the decision.

Low Construction Overhead. There is no methodology on building such systems as described in this paper in sight yet, but such a methodology must address the topic of low construction overhead.

Efficiency. Regarding Case-Based Reasoning, results from several year of research and application are at hand. For example, with Case Retrieval Nets (Lenz and Burkhard, 1996) an efficient technology is with minimum storage requirements is given, but those methods are not built to work with dynamic views or scopes. We, therefore, see also the need for further research activities.

So, what kind of explanations are of interest to decisions on trust and what are the questions to be asked? This question cannot be answered in a general way. This very much depends on the problem at hand. Cognitive explanations will surely be

the most important kind, because those explanations inform the user about what the system actually did, thus, providing the user with some sense of control over it. Nevertheless, the kinds of explanations, described in Section 3.2. can help to develop the explanation capabilities needed for a particular system. We will address it in our further research.

5. Conclusions

It has been proposed, that a proper context model lays a basis for situation depending decisions as for example trust. Context can act as a carrier for such decisions, providing a mechanism to conclude from a current context to experienced ones. The use of such complex mechanisms leads to the necessity for explanations. We discussed the requirements for good explanations and what kinds are of most interest. Further, the incorporation of trust into a context-based reasoning approach has been shown and how to compute trust for new contexts to allow trust-based decisions and facilitate trust management. And the need for explanations on the resulting trust values has been discussed.

6. References

- Aamodt A. and Plaza E., Case-based reasoning: Foundational issues, methodological variations, and system approaches. *AI Communications*, 7(1):39–59, 1994.
- Bagci F., Petzold J., Trumler W., Ungerer T., Einsatz von XML zur Kontextspeicherung in einem agentenbasierten ubiquitären System, *XMIDX-Workshop*, Berlin, 2003.
- Bergmann R., Althoff K.-D., Breen S., Göker M., Manago M., Traphöner R., Wess S., *Developing Industrial Case-Based Reasoning Applications: The INRECA Methodology*. LNAI 1612. Springer-Verlag, Berlin, second edition, 2003.
- Bergmann R. and Schaaf M., On the Relations between Structural Case-Based Reasoning and Ontology-based Knowledge Management, *German Workshop on Experience Management (GWEM 2003) at WM 2003*, Luzern, 2003.
- Berners-Lee T, Hendler J., Lassila O., The semantic web: A new form of web content that is meaningful to computers will unleash a revolution of new possibilities. *ScientificAmerican.com*, May 2001.
<http://www.scientificamerican.com/article.cfm?articleID=00048144-10D2-1%C70-84A9809EC588EF21> [Last access: 2004-04-29].
- Brézillon P., Context in Artificial Intelligence. *Journal Computers and Artificial Intelligence*, Vol. 18/4, P. 321-340, and Vol.18/5, P. 425-446; 1999.
- Blaze M., Feigenbaum J., Lacy J., Decentralized trust management. *IEEE Conference on Security and Privacy*, Oakland, CA, 1996.

- Deutsch M., Cooperation and trust: Some theoretical notes. *Nebraska Symposium on Motivation*, Nebraska University Press, 1962.
- Fox M. S. and Huang J., Knowledge Provenance: An Approach to Modeling and Maintaining the Evolution and Validity of Knowledge. Technical report, Enterprise Integration Laboratory, University of Toronto, Toronto, Canada, 2003.
<http://www.eil.utoronto.ca/km/papers/fox-kp1.pdf> [Last access: 2004-11-11].
- Golbeck J., Parsia B., Hendler J., Trust Networks on the Semantic Web. *Conference on Cooperative Intelligent Agents*, 2003.
- Gonzalez A. J. and Ahlers R., Context-Based Representation of Intelligent Behavior In Training Simulations, *Transactions of the Society for Computer Simulation International*, Volume 15, No.4, 1998.
- Grandison T. and Sloman M., Trust Management Tools for Internet Applications. *1st Int. Conference on Trust Management*, Crete, Springer LNCS 2692, pp 91-107, 2003.
- Groenendijk W., Maintenance management system: Structure, interfaces and implementation. In Süleyman Özekici, editor, *Reliability and Maintenance of Complex Systems*, volume 154, Berlin, 1996. Springer-Verlag.
- Lei H., Sow D. M., Davis J. S., Banavar G., Ebling M. R., The design and applications of a context service. *Mobile Computing and Communications Review 6, ACM SIGMOBILE*, pp 45–55. ACM Press, NY, USA. 2002.
- Lenat D. B., The Dimensions of Context-Space. Cycorp, Austin, Texas, 1998.
- Lenz M. and Burkhard H.-D., Case Retrieval Nets: Foundations, Properties, Implementation, and Results, Technical Report, Humboldt University of Berlin, 1996.
- Marsh S. P., *Formalising Trust as a Computational Concept*. University of Sterling, 1994.
- McGuinness D. L. and Pinheiro da Silva P., Infrastructure for Web Explanations. *The Semantic Web — ISWC 2003*, pages 113–129, 2003.
- Mikalsen M. and Kofod-Petersen A., Context: Representation and Reasoning, in this issue, 2005.
- Mitchell K., Supporting the Development of Mobile Context-Aware Computing, Ph.D. Thesis in Computer Science, Lancaster University, 2002.
- Moore J. D. and Swartout W. R., Explanation in expert systems: A survey. Research Report RR-88-228, University of Southern California, Marina Del Rey, CA, 1988.
- Norlander L., A Framework for Efficient Implementation of Context-Based Reasoning in Intelligent Simulation, Master's Thesis, ECE Dept. University of Central Florida, 1999.
- Pascoe J., Context-Aware Software, Ph.D. Thesis in Computer Science, University of Kent at Canterbury, U.K., 2001.
- Pepper S. and Grønmo G. O.. Towards a general theory of scope, *Extreme Markup Languages 2001*, Montréal, Canada, 2001.

- Reinartz Th., Iglezakis I., Roth-Berghofer Th., Review and restore for case base maintenance. *Computational Intelligence: Special Issue on Maintaining Case-Based Reasoning Systems*, 17(2):214–234, 2001.
- Reinert J. and Ritter N., Applying ECA-rules in DB-based design environments. In *Tagungsband CAD'98 "Tele-CAD – Produktentwicklung in Netzen"*, pages 188–201. Informatik Xpress 9, 1998.
- Richardson M., Agrawal R., Domingos P., Trust management for the semantic web, *The Semantic Web — ISWC 2003*, pages 351–368, Springer-Verlag, 2003.
- Roth-Berghofer Th. R., Knowledge Maintenance of Case-Based Reasoning Systems – The SIAM Methodology, volume 262 of *Dissertationen zur Künstlichen Intelligenz*. Akademische Verlagsgesellschaft Aka GmbH / IOS Press, Berlin, Germany, 2003.
- Rousseau D. M., Sitkin S. B., Burt R. S., Camerer C., Not So Different After All - A Cross-Disciplined View of Trust. *Academy of Management Review*, Vol.23/3, P. 393-404, 1998.
- Schmidt A., Ubiquitous Computing – Computing in Context, Ph.D. Thesis in Computer Science, Lancaster University, U.K., 2002.
- Schulz S., Herrmann K., Kalklösch R., Schwotzer T., TAKEUP - Trust-based Agent-mediated Knowledge Exchange for Ubiquitous Peer Networks, In *Agent-Mediated Knowledge Management: Selected Papers*, LNAI 2926, Springer, Germany, 2003.
- Schurz G., Scientific explanation: A critical survey, IPS-Preprint 1, Department of Philosophy, University of Salzburg, 1993.
- Schwotzer T. and Geihs K., Mobiles verteiltes Wissen. *Datenbank Spektrum*, 3(1), 2003.
- Spieker P., Natürlichsprachliche Erklärungen in technischen Expertensystemen. Dissertation, University of Kaiserslautern, 1991.
- Stensrud B. S., Barrett G. C., Gonzalez A. J., Context-Based Reasoning: A Revised Specification, *Seventeenth International Florida Artificial Intelligence Research Symposium Conference*, Florida, USA, AAAI Press, 2004.
- Swartout W. R., XPLAIN: A system for creating and explaining expert consulting programs. *Artificial Intelligence*, 21(3), 1983.
- Swartout W. R. and Moore J. D., Explanation in Second Generation Expert Systems. *Second Generation Expert Systems*, pages 543–585, Berlin, 1993, Springer Verlag.
- Turner R. M., Context-Mediated Behavior for Intelligent Agents, *International Journal of Human-Computer Studies*, 48(3):307–330, 1997.
- TopicMaps*. ISO/IEC 13250: Topic Maps: Information Technology - Document Description and Markup Languages, Michel Biezunski, Martin Bryan, Steven R. Newcomb, ed., 1999. <http://www.y12.doe.gov/sgml/sc34/document/0129.pdf>

Thomas R. Roth-Berghofer is affiliated with the Knowledge Management Research Department of the German Research Center for Artificial Intelligence DFKI GmbH since 2002. He is also managing the Knowledge-based Systems group of Prof. Andreas Dengel at the University of Kaiserslautern. Dr. Roth-Berghofer gives lectures about the Semantic Web and Case-Based Reasoning. His main research interests are in trustworthy and explainable knowledge-based personalized support of KM activities. He co-initiated a workshop series on Philosophy and Informatics and an accompanying Special Interest Group as well as a workshop series on Modelling and Retrieval of Context. He is co-organizer of the Third Conference Professional Knowledge Management WM 2005 in Kaiserslautern and co-chair of the European Conference on Case-Based Reasoning ECCBR 2006 in Ölüdeniz/Fethiye, Turkey.

Stefan Schulz is affiliated with the Department of Systems and Computer Engineering of the Carleton University in Ottawa. He currently researches self-organizing aspects for combinatorial information retrieval and creation. His main research interests are peer-based, context-aware systems focusing on sharing and managing knowledge in communities of practice and organizations. He co-initiated a workshop series on Modelling and Retrieval of Context.