A framework for context-based adaptation (for collaboration)

Bjoern Joop, Juergen Ziegler
Department for Computer Science and Applied Cognition Science,
University of Duisburg-Essen
bjoern.joop@uni-due.de, juergen.ziegler@uni-due.de

Abstract. The topic of our talk focuses on definitions of context and approaches in computer science to model context in adaptive or context-aware systems. We present briefly an own context-understanding, a multi-layered framework for context-based adaptation. We present some current examples for context-based adaptation and conclude with some further thoughts about adding unstructured information such as tags to our context understanding to be able to mediate knowledge between different contexts and users.

Keywords: context, adaptation, context-awareness, collaboration, context modeling; co-location; co-recommendation

1 Introduction

Modern organizations work to a large extent collaborative. This is particularly true for knowledge work which is often performed by distributed teams cooperating over global distances. This poses a number of challenges that need to be addressed effectively for supporting the cooperative work.

Collaboration support requires a wide range of communication facilities, tools and information resources. These must be used in dynamically changing collaboration situations in an effective and efficient manner. This often leads to a high complexity of the collaboration environment, to cognitive overload on the users' side, inefficient interaction and thus suboptimal use or even outright rejection of sophisticated collaboration support systems [1]. To solve these problems, the functionality and interaction offered by the collaboration environment should be adjusted to the current collaboration situation. To date, users and teams wishing to adjust their working environment have to negotiate and perform such changes manually. This leads to a high cognitive overhead, ignoring the potential for improvement and thus lowers the team's performance.

The solution for these problems is to make the collaboration environment adaptive. Self-adaptation of systems to changing user needs and situations has been investigated in various application domains (e.g. intelligent tutoring systems, product recommendations, location-based services, etc.). In these approaches the focus has mainly been on the individual and not on collaboration situations.

The notion of context is paramount for any kind of adaptive system. Context-aware and context-adaptive systems have been a major research topic in fields such as mobile applications and ubiquitous computing. Even more, a generalized view of context and methods for its' systematic use in adaptive interactive systems are still missing.

We present an attempt to formulate a notion of context applicable for both single-user and collaborative work and introduce a method for managing dynamic context in an adaptive system.

2 Definitions of context

The word "context" shows its meaning inherently: *con* (meaning: with) *text*. This definition has its origin in linguistics expressing a communicative goal [2] or describes the surrounding situation for an easier interpretation [3, 4]. In philosophy context is used as a correlation between sentences or defines aspects of a situation with the goal to help understanding the semantic meaning of an expression [5]. Psychologists research context regarding how changes of the situation affect cognitive processes [6, 7].

In computer science the notion of context has played an increasingly important role especially in the area of ubiquitous computing with the aim of developing context-aware systems [8]. Context-aware systems usually use time, location, users and available resources as contextual information representing aspects of the physical world [9].

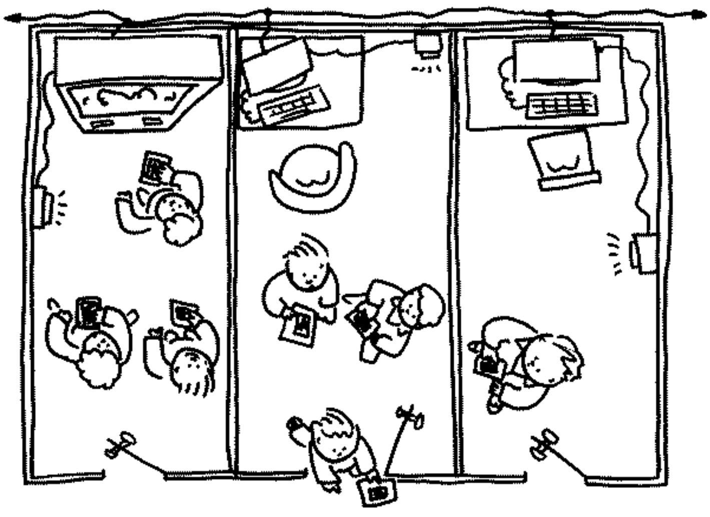

Fig. 1. Context-aware computing application scenarios, Schilit '94 [9]

Later on, these context dimensions were extended using sensors and object properties [10]. An general definition was provided by Dey et al [11, 12] defining context: *"context is any information characterizing the situation of an entity"*. Another popular definition was given by Winograd [13] defining context as an operational term for characterizing its role in communication. This means something is context because it is used in communication for interpretation and not due to its inherent properties. Winograd further distinguishes between context and setting. In his opinion, the setting includes all fixes information about users nearby, the place an action takes place, etc. On the other hand, context represents all information helping understanding a communication.

Approaches modeling context range from simple key-value models over graphical or hierarchical models up to sophisticated ontology-based context models which support validation and reasoning (cf. [14]).

3 Our understanding of context

We agree with Winograd, that we need a distinction between context and setting. But in our understanding, we need contextual (meaning: surrounding) information in regard to a center of attention (focus). The context depends on the current person's perspective, known facts and the state of the "real" world and applications. This includes information about the setting of the current situation.

But this also means, that context is ever-changing and dynamic. These changes can easily be captured and interpreted by humans; context-aware applications on the other hand have to use sensors to recognize these changes and have to interpret it due to predefined rules.

We see the contextualization as a selection process: In a complex situation, contextualization mechanisms are used to extract the most relevant elements. Therefore we need the following components for a conceptual context mode:

- Information about the current situation (state) provided by sensing components mapping internal or external information sources into state objects
- Background information about the application domain
- Contextualization rules to constitute a contextualized state
- Adaptation rules defining a set of meaningful adaptations in regard to the contextualized state.

4 A framework for context-based adaptation

In [15] we present a framework for context-based adaptation which has to meet a set of formal requirements:

- it must be a formal representation of context
- must be able to capture a wide range of adaptation purposes
- must be able to capture aspects of collaborative work
- must be able to support both single user and multi user scenarios

We proposed a multi-layered framework for such a framework. We distinguish between four layers: a knowledge layer, a state layer, a contextualization and an adaptation layer (see figure 2).

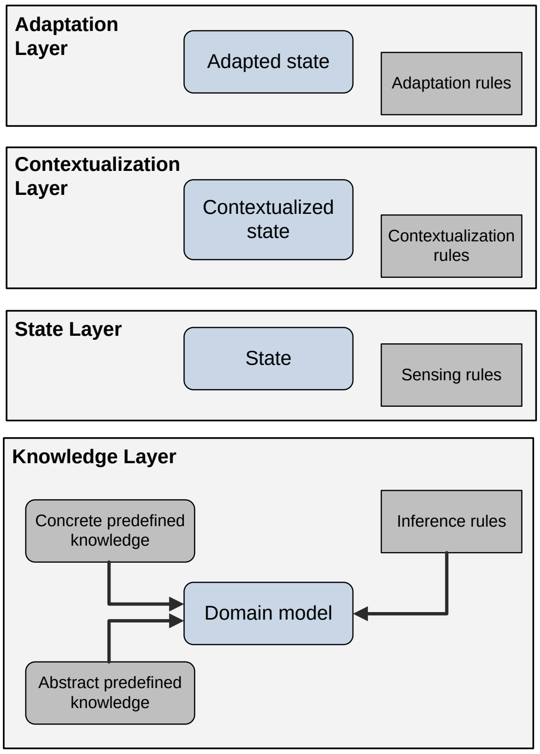

Fig. 2 A four-layered framework for context-based adaptation [15].

For more details see [15].

4.1 Knowledge Layer

The knowledge layer uses predefined information to represent aspects of the physical world, users, the computing environment and resources. The information can be represented using different techniques (e.g. Web Ontology Language using relations, classes and individuals). It is possible to distinguish between abstract and concrete knowledge.

Examples:

"Berlin is a city." represents stable, predefined knowledge because this fact will not be easily changed.

"Every city has a name." represents using classes and relations predefined abstract knowledge.

4.2 State Layer

The state layer represents the current situation including domain knowledge from the knowledge layer. Using sensors and rules we are able to filter and map external or internal (system) information into the state model.

For representing the current state, a directed graph Gs = {V,E,σ} with vertices and V, edges E and an relation strength function σ: E x T → [0,1] is used (see [15]). This can be precisely represented by an RDF graph.

4.3 Contextualization Layer

The contextualization layer provides techniques that define which subset of the state is relevant for a given focus. We call those techniques contextualization techniques (e.g. rule-based).

A focus is a non-empty set of objects from the state representing the current center of attention. The focus can be set by the user or by an application and is used as the starting point for applying contextualization techniques.

Incorporating relation strength values inside the state is optional and depends on the contextualization technique. For example Spreading Activation [16, 17] has particularly been used in information retrieval systems [18, 19, 20] and would use relation strength.

4.4 Adaptation Layer

The contextualized state contains information about individuals relevant for the current focus. The adaptation layer contains adaptation rules, which describe which adaptation actions should be performed in which case. Adaptation rules include operations for changing properties, artifacts or in general any state variable ("show", "start", etc.). Effects of these adaptations have to be propagated to the user interface.

We propose to represent adaptation rules as IF-THEN rules such as:

IF ?document is_important THEN open ?document.

5 Context-based adaptations for collaboration scenarios

In [15] we discussed the applicability of this approach for four typical collaboration scenarios or episodes:

- Co-location denotes the situation where several people meet at a physical or virtual location
- Co-access denotes the situation where several people access the same artifact

- Co-recommendation denotes the situation where explicit or implicit actions of the users are used to suggest information resources potentially useful
- Co-dependency denotes the situation where several tasks, objects or users are dependent

We introduce co-location and co-recommendation as exemplary scenarios.

5.1 Co-location

Co-location occurs when two or more people, artifacts or devices are physically near to each other. Co-location affords a range of adaptations, for example, facilitating the use of nearby devices or automatically setting up the collaboration system for joint tasks or information access.

Scenario: Alice and Bob, jointly working on a report, gather in a meeting room equipped with a large shareable display. Sensors recognize and identify the two persons. The system infers that they will likely work on the current report. It activates the display and shows the report in the workspace. Alice and Bob can now immediately start discussing the latest version of the report.

For this scenario, we use the domain model shown in figure 3. An actor works with a device, such as a computer or a display, so the device must be able to display artifacts. Additionally, each actor and each device can be at some location. Actors who are at the same location are linked by an additional relation *isLocatedWith*.

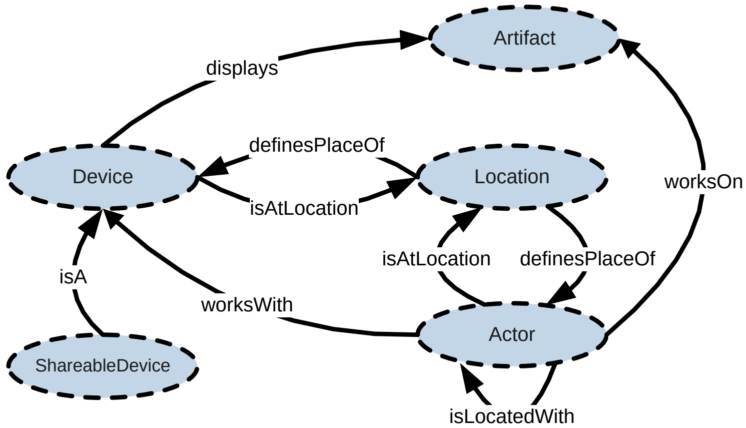

Fig. 3 An exemplary domain model including *"Actors"*, *"Location"* and *"Device"*

The state is contextualized by rules expressing which concepts are important for a given focus in this model. In the example, a rule could, for instance, state: *"IF at least two persons are in the same room AND work on the same artifact AND the room has a shareable device (e.g. a large display) THEN this particular artifact and device are important."* In this case, persons, shared artifacts and locations are the focus points, for which potentially important shared artifacts and matching shareable displays are computed as context. We define the context by examining the state using three conditions: Is the artifact simultaneously being worked on? Are the persons working on it in the same room? Does the room have a shareable display?

The contextualized state is shown in figure 4. In our scenario, it includes the location *Location:l_meetingRoom* where both *Actor:a_alice* and *Actor:a_bob* are and the devices at that location (here: *Device:d_shareableDisplay*). Furthermore, all artifacts (Artifact:a_report) on which both actors currently work on collaboratively are included.

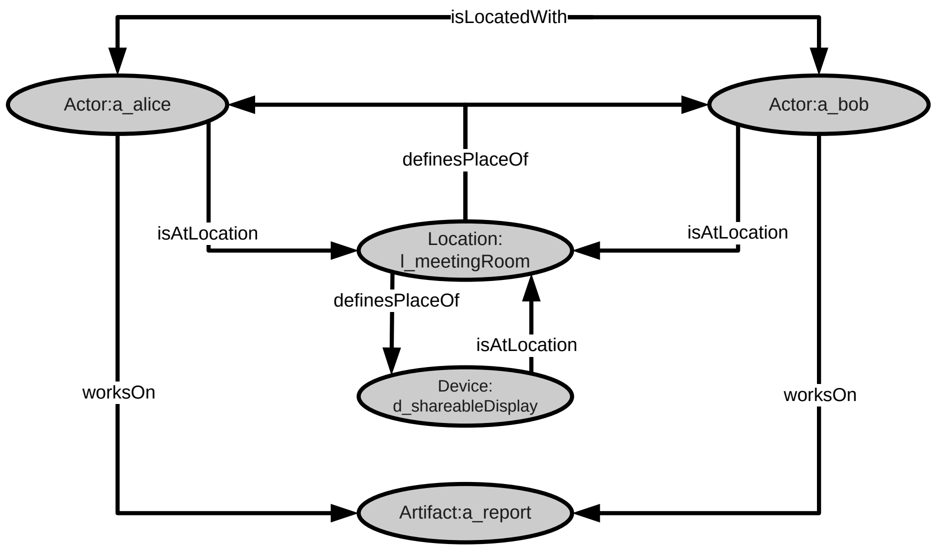

Fig. 4 Contextualized state after creating a co-located situation

An adaptation rule, which adapts all devices in the room based on the contextualized state, could be formulated as follows:

```
// Obtain artifacts and device from the contextualized
// state
artifacts := getArtifactsFromContextualizedState;
device := getDeviceFromContexutalizedState;
// If there are shared artifacts and a shareable device,
// then display the artifacts on the display
IF (notEmpty(artifacts) AND notEmpty(device)) THEN {
  display(artifacts,device);
}
```

The adaptation rule is triggered when all parts of the condition are fulfilled: The adaptation rule is triggered, because *Actor:a_bob* enters *Location:l_meetingRoom* which *Actor:a_alice* has entered before. Both work on several artifacts simultanously and *Location:l_meetingRoom* provides a shareable display. Consequently, the adaptation rule obtains an artifact and a shareable display from the contextualized state (neither of the sets is empty) and *Device:d_sharableDisplay* will be activated and it will display *Artifact:a_report*.

5.2 Co-Recommendation

Co-recommendation is a technique in which the system suggests potentially relevant information resources based on prior action of other group members and one' own interest profile.

In this exemplary scenario, an actor Bob works for a building company specialized in refurbishments. He is an expert in the area of energy efficiency and frequently tags documents or web pages with keywords from this domain. On this basis, the system classifies documents into a predefined semantic model. Alice is a new colleague and enters the system to search for information regarding thermal insulation. The system recommends a number of documents concerning this topic.

Later on, Bob adds a new document to the system which he rates highly relevant to the topic of roof insulations. Based on Bob's rating, the system recommends this document to Alice through an appropriate awareness function or when Alice logs in the next time.

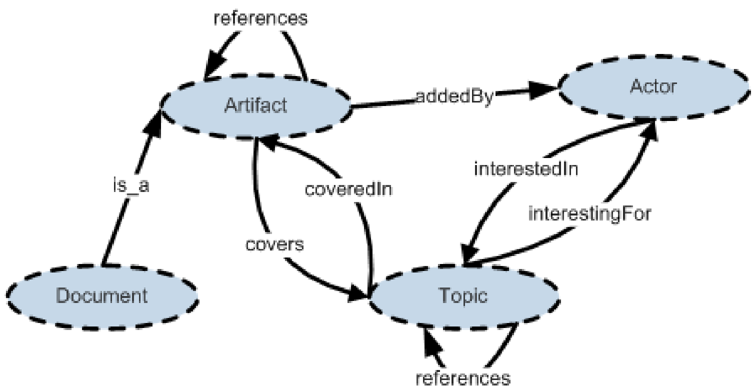

Fig. 5 Domain model for co-recommendation.

In order to allow the system to recommend Alice documents, we need a domain model (cf. figure 5) including classes for *"Topic"* and *"Document"* and create additional relations connecting them. In this example, we use a spreading activation algorithm for contextualization.

Each artifact has one or more topics assigned to it and may reference other artifacts. The relation strength represents the degree of ‘relatedness’ of two instances. In the example shown in figure 6, *Actor:a_alice* is very interested in the topic *Topic:t_energyInsulations* which is represented by a value of 0.9. On the other Hand, the topic *Topic:t_energyInsulations* is only peripherally covered in the document *Artifact:a_manualProofing* – maybe a manual for do-it-yourself house proofing – which is represented in the relation strength of 0.4.

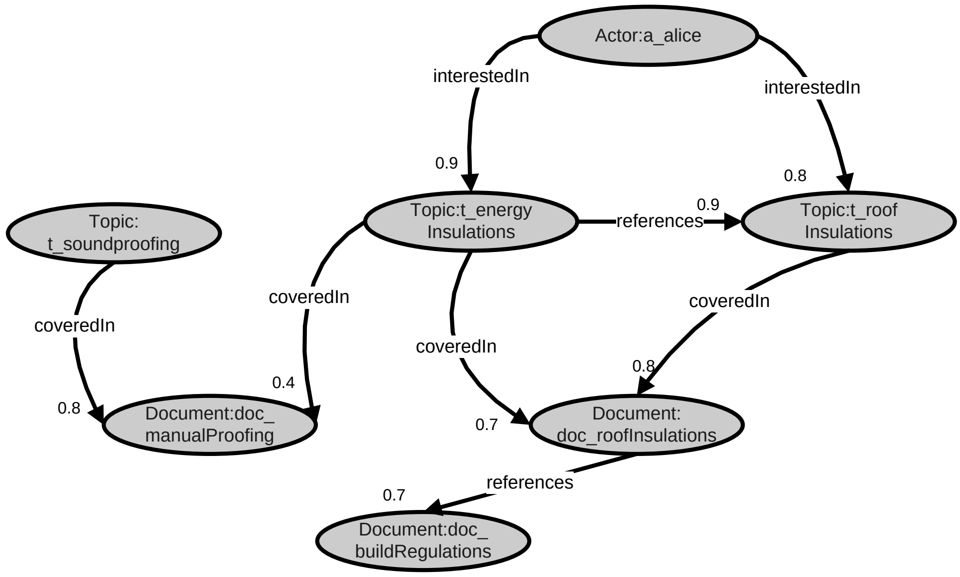

Fig. 6 A possible state for a co-recommendation scenario. Relations strengths are assigned to each relation. Note that not all relations from the domain model in figure 5 are included.

In this example, we use spreading activation (cf 4.3) as a contextualization technique. This could work as follows:

1. Select the node *Actor:a_alice* as initial nodes (the focus)
2. Activate all vertices connecting *Actor:a_alice* to adjacent ones
3. Propagate the activations until either:
   - the distance from the initial node to the current one is greater than α
   - the relation strength is greater than β
   - the propagating vertices’ activation is smaller than γ

   using α , β and γ as predefined (terminating) parameters.

The result of the spreading activation algorithm applied on the graph is being shown in figure 7.

The interpreted state shown in figure 8 is the result of applying the spreading activation techniques and choosing only those objects with an activation greater than 0.5.

Based on the contextualized state, we can define an adaptation rule for recommending artifacts of interest to *Actor:a_alice*:

```
//select all documents from the contextualized state
//with an activation > 0
Documents := getDocumentsWithActivationLagergerThan(0.5);
//displaying all documents
IF (notEmtpy(documents))  display(Documents);
```

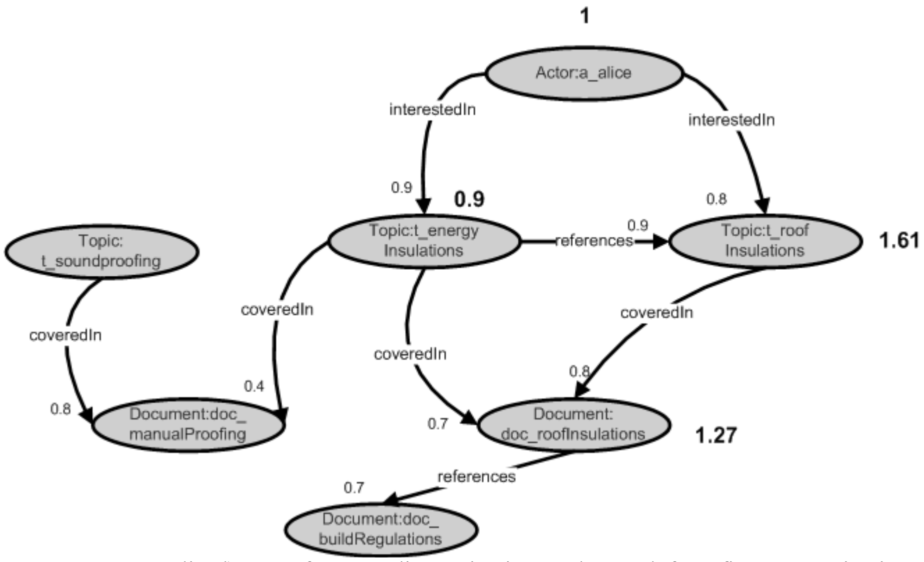

Fig. 7 (Contextualized) state after spreading activation on the graph from figure 6. Activations are shown bold highlighted.

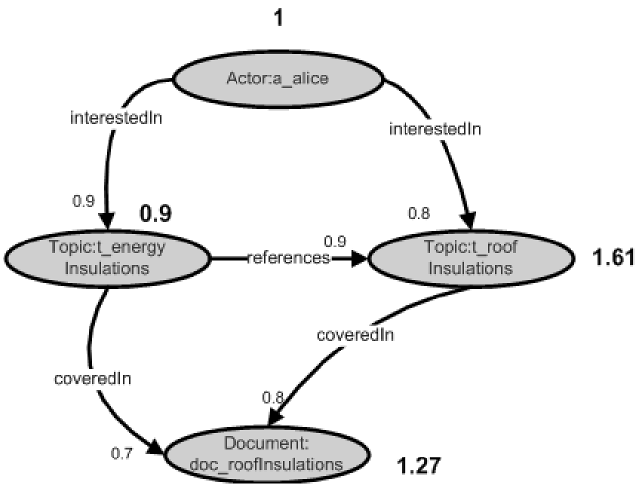

Fig. 8 Interpreted state after spreading activation and projection.

The function *getDocumentsWithActivationValueLargerThan* returns from the contextualized state a set of all documents with an activation greater than 0.5. Accordingly, *Actor:a_alice* will get a recommendation to read the document *Document:doc_roofInsulations* which covers some of the topics she is interested in.

After Bob adds a new document to the database (*Document:doc_newRoofInsulations*)– which leads to a new state (cf. figure 9) – the system has two alternatives for setting the focus: Firstly, it can use the newly added document as the new focus and thus as initial node for the contextualization by spreading activation for selecting all actors who are highly interested in the topics covered in the document. Secondly, the actor can be used as a new focus after login to recommend the most relevant documents.

Using the first alternative, awareness functions are able to directly inform actors interested in the topic *Topic:t_roofInsulations* about the new document *Document:doc_newRoofInsulations* added by *Actor:a_bob*.

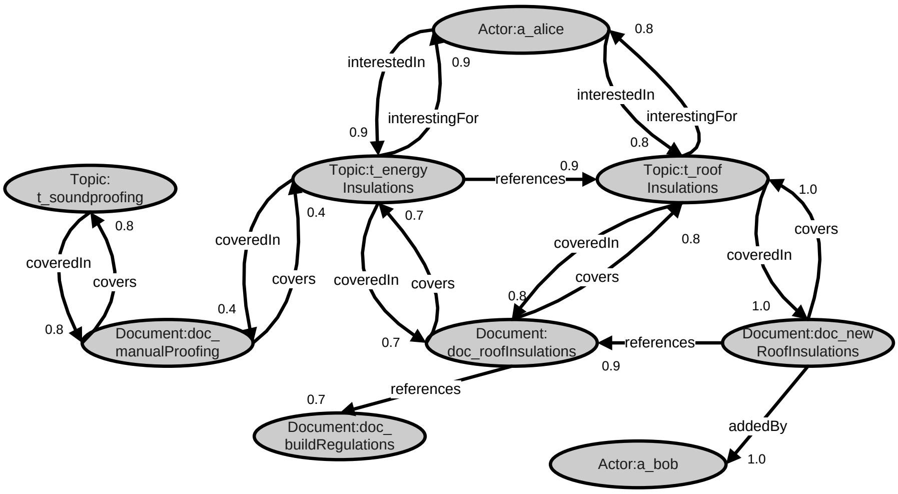

Fig. 9 Excerpt of the state after *Actor:a_bob* adds a new document *Document:doc_newRoofInsulations*. Note, that in this figure all relations from the domain model (cf. figure 5) are included.

The contextualized state after applying the spreading activation with the focus on *Document:doc_newRoofInsulations* is illustrated in figure 10.

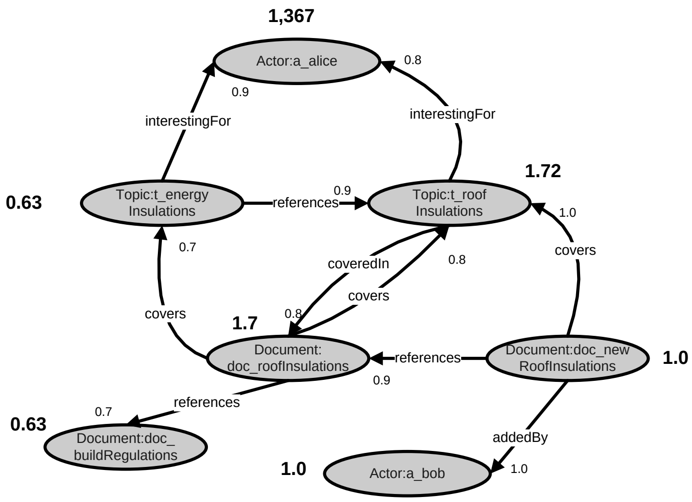

Fig. 10 Contextualized state after spreading activation with focus on new document. Note that not all nodes from figure 9 are included.

Further user-based adaptations can be made using a feedback mechanism. If a user is not satisfied with the recommendations for a specific topic, he can provide feedback which directly decreases the relation strength connecting the artifact with the given topics or to decrease the relation strength between references from artifact to artifact or topic to topic.

7 Future work

Currently we are easily able to use the context of an individual or a global, predefined shared context for all users working collaboratively. We are still missing a formal definition of a group context and a solution about how to create such a group context from individual contexts.

Additionally, we want to include unstructured information in form of individual and collaborative tagged information in our understanding of context. This approach focuses on further individualizing of user contexts using their own language and taxonomy for annotation of important information. To be able to share information in a more global context, we need to mediate between individual contexts using a shared group context from this information. This may lead to be able to switch between user contexts to "look at a situation from a totally different view".

Further work will be implementing and testing a collaborative context server to enable information transfer from one person to another. Therefore new collaboration tools with context-aware components have to be developed. Also, we will investigate new contextualization techniques and adaptation rules.

Acknowledgments. This work is supported by the German Research Foundation (DFG) within the cluster project "Context Adaptive Interaction in Cooperative Knowledge Processes" (CONTici).

References

1. V. Wulf and B. Golombek, Exploration Environments – Concepts and Empirical Evaluation, in Proceedings of the 2001 International ACM SIGGROUP Conference on Supporting Group Work (GROUP'01), 2001, pp. 107-116
2. H. Bunt, Context and Dialogue Control, *THINK Quarterly*, Vol. 3, 1994.
3. B. Malinowski, The Problem of Meaning in primitive Languages, *The Meaning of Meaning* (1923), pp. 146-152.
4. J. L. Austin, *How to Do Things with Words* (Oxford: O.U.P., 1962).
5. R. Carnap, *Meaning and Necessity: A Study in Semantics and Modal Logic*, 2nd Ed., 1988.
6. G. M. Davies and D. M. Thomson, Memory in Context; Context in Memory (John Wiley & Sons Inc., 1988), pp. 1-10.
7. T. Ziemke, Embodiment of Context, in *Proceedings of ECCS*, 1997.
8. H. Bunt, Context and Dialogue Control, *THINK Quarterly*, Vol. 3, 1994.
9. B. Schilit, N. Adams and R. Want, Context-Aware Computing Applications, (IEEE) Workshop on Mobile Computing Systems and Applications, 1994.
10. J. Roth, *Mobile Computing* (dpunkt.verlag, 2nd. Edition, 2005).
11. A. K. Dey, G. D. Abowd, P. J. Brown, N. D. Davies, M. Smith and Pete Steggles, Towards a Better Understanding of Context and Context-Awareness, in Proceedings of the 1st international symposium on Handheld and Ubiquitous Computing, 1999, pp. 304-307.
12. A. K. Dey, Understanding and using Context, Personal Ubiquitous Computing Nr. 1 Vol. 5 (2001), pp. 4-7.
13. T. Winograd, Architectures for Context, Human-Computer Interaction, Vol. 16 (1991), pp. 401-419.
14. T. Strang and C. Linnhoff-Popien, A Context Modeling Survey, Workshop on Advanced Context Modelling, Reasoning and Management, The Sixth International Conference on Ubiquitous Computing, 2004.
15. J. Haake, T. Hussein, B. Joop, S. Lukosch, D. Veiel, J. Ziegler (2009) Technische Berichte der Abteilung für Informatik und Angewandte Kognitionswissenschaft, Nr. 2009-02
16. A. M. Collins and E. F. Loftus, A Spreading Activation Theory of Semantic Processing. Psychological Review, 82(6) (1975), pp. 407–428.
17. J. R. Anderson, A spreading activation theory of memory. Journal of Verbal Learning and Verbal Behavior, 22 (1983), pp. 261–295.
18. G. Chen and D. Kotz, A survey of context-aware mobile computing research, Dartmouth College Technical Report, 2000.
19. F. Crestani, Application of spreading activation techniques in information retrieval. Artificial Intelligence Review, 11(6) (1997), pp. 453–482.
20. G. Salton and C. Buckley, On the use of spreading activation methods in automatic information. In Yves Chiaramella (Ed.), Proceedings of the 11th annual international ACM SIGIR conference on Research and development in information retrieval (pp. 147–160) 1988. Grenoble, France: ACM.