# Modeling Mental Contexts and Their Interactions

**Wei Chen and Scott E. Fahlman**

Language Technologies Institute, Carnegie Mellon University
Pittsburgh, PA 15213, USA
{weichen, sef}@cs.cmu.edu

**Abstract**

The ability to understand and process multiple mental contexts is an important aspect of human cognition. By "mental contexts" we mean different beliefs, states of knowledge, points of view, or suppositions, all of which may change over time. In this paper, we propose an approach for modeling and reasoning about the interactions among multiple mental contexts using the context activation scheme in Scone knowledge-base (KB) system. Our model factors the mental context representation into two separate components: (1) a dynamic mental context network (2) a set of rules which guides the activities among mental contexts and their evolution as a result of this. Our model is capable of combining newly available information and old memories stored in the context network to produce new mental states. We demonstrate the approach with a story-understanding task, in which the users feed information to the program, then ask questions about the newly updated beliefs or assumptions.

## Introduction

One challenge in building models that mimic human cognition is to understand distinct world-views from multiple agents. By world-views we mean states of belief, supposition, intention, advice, perceived reality, as well as situations expressed by tenses in natural language such as past, present and future. In this paper, we call these world-views "mental attitudes". These are represented in our model as "mental contexts" so that we can activate one context and reason about it, without mixing facts and beliefs in the activated context with things in other mental contexts. Even young children are skilled in handling nested mental states, and simple children's stories are rich in examples: the wolf pretended to be grandmother, and for a while little red-cap believed this, but it was not really true. However, computer programs are still not good at understanding these. The purpose of our work is to model the human cognition in mental states and reason about what is true and what would happen in the mental contexts using a context activation scheme. An edited excerpt of the story "Little Red-Cap"[1] illustrates a scenario under which this kind of inference is necessary:

> Little red-cap's mother told her to take some food to her grandmother and warned her not to run off the path. A wolf met little red-cap on her way. He suggested that she runs deep into the wood to pick flowers. When little red-cap arrived at her grandmother's home, she felt strange because her grandmother's appearance changed a lot. It turned out the wolf pretended to be her grandmother …

A human reader understands several changes in the mental states of the characters in the story. For example, little red-cap first intended to stay on the path, but once the wolf suggested that she looks around, she forgot (or neglected) her mother's words. Here, a "suggesting" event, which is conducted through the wolf's mental states and actions, brings something else to little red-cap's attention. Meanwhile, it causes some previously active statements (not to run off the path) in little red-cap's mind become dormant. After that, little red-cap felt strange because a reality-expectation matching process reported conflicts. Finally the fact is revealed through the wolf's action.

There are two fundamental parts which need to be understood in the story: (1) the mental contexts of the characters (e.g. little red-cap's belief and the wolf's intention); (2) the interactions between the mental contexts (e.g. the changes of little red-cap's belief and attention under the other's influence). Our goal is to model the interactions between mental states in a context network.

### Mental States of Agents

Mental states such as beliefs and intentions have been studied for decades in the AI community. A basic goal is to interpret and make predictions about an agent's actions. Beliefs, goals (or desires), and plans (or intentions) are usually represented with mental states and actions in formal theories to draw connections between agents' minds, actions, and communications with other entities (e.g. Cohen and Levesque 1990). Much of the attention on action control is based on the "BDI" (belief, desire, intention) architecture (Rao and Georgeff 1995). Mental interactions between rational agents are also well studied in many other cognitive architectures for cooperative tasks



[1] We use "Little Red-Cap" in Grimm's Fairy Tales as a running example in this paper. Original text understanding is shown in a later section.

(e.g. Singh 2005). Our approach is different from the others in the following aspects. First, our goal is not to model goal-driven behaviors but reactive mental state changes, namely how the cognition grows with gradually available knowledge. Second, we try to cover general concepts of mental attitudes which include things like regretting and realizing. Third, we represent mental states as descriptions within mental contexts, which gives an intuitive way of modeling various mental attitudes by a convenient context activation scheme. Fourth, unlike many formal logic representations, we do not use mutual knowledge or mutual belief in multi-agent environments; rather, we model a multi-agent reasoning scheme from purely single agents' points of view at any one time, which seems closer to human cognition.

Our model represents each psychological term (e.g. "realize") in natural language as a composed context net that could be reasoned about from our general multi-context mechanism built on top of the Scone knowledge base (KB)[1]. Because of this, our implementation can be used as a cognitive programming language, where the syntax is made up of psych-terms and semi-natural statements. An example will be shown in the experiment section.

### Representing Mental States in Mental Contexts

Contextual reasoning[2] has been emphasized to resolve the generality problem in AI (McCarthy 1987). A context is a container that holds a set of descriptions. It can either be used to represent the "state of the mind" or a "situation". We use mental contexts to model a set of mind statements under the same kind of mental attitude.

There are several reasons that we represent mental states in mental contexts. First, by using contexts, it is easy to organize mental states and knowledge in a well-formed structure that provides ease of maintenance and search. Second, once we have a structure among contexts, the mental interactions can be viewed as communications among contexts. Thus, we could factor the mental state representation into two separate components (a context structure and a set of inter-contextual rules) and study them separately. As in (Ghidini and Giunchiglia 2001), contexts are not isolated; they can be connected by *bridge rules*. In addition, contexts can be nested (Jago 2006), so it is not hard to model one person's belief of someone else's belief using mental contexts.

Similar to (Jago 2006), our model represents time, where we store one copy of a mental context at each successive time point. Therefore, each context node and its active contents can be treated as a *time slice*. On the other hand, our model captures an evolving world from each single agent's point of view (e.g. person P2's intention in person P1's view). Hence we could avoid an unrealistic entity which holds global knowledge.

[1] Scone is an open-source KB system. http://www.cs.cmu.edu/~sef/scone

[2] There are many well-written papers on formal contextual reasoning: e.g. Akman and Surav (1996), Serafini and Bouquet (2004).

People can make mistakes on their judgments when interacting with other people and the general reality. One reason is that some relevant facts are neglected or forgotten. In our model, active and dormant memories are also modeled under contexts. Dormant memory exists in the mind, temporarily inactive, but can be reactivated by new entities or events. The inactive memories are hidden in a parent environment (context). This approach is different from some formal representations where active memory (focus) is controlled by an accessibility threshold (Gordon and Hobbs 2004).

### Mental States in Natural Language

Ideally, a Scone language engine will take sentences as input and match the terms to Scone elements. Then an inference engine can take the Scone elements and perform further inference. In our case, psych-terms such as "realize" will be matched to Scone psych-events, which connect a set of mental contexts. There are other formalisms of psych-terms in natural language. For example, Mueller (1990) represents psych-terms as streams of thoughts driven by emotion and goals; Singh (2005) uses a frame-based narrative representation; McCarthy (1995) proposes mental situation calculus to model these mentalistic notions. But none of these represents the psych-terms as mental contexts, which we believe provides generality and efficiency.

### Story Understanding

In story understanding tasks, a computer program takes a story as input, understands it, and answers questions about the story. In this paper, we set our experiments within a story understanding task. Unlike some other deep understanding programs (e.g. Schank and Abelson 1977; Dyer 1983; Mueller 2004), our model currently only concentrates on understanding the mental states of the characters. Unlike some shallow understanding tasks such as the opinion and sentiment summarization applications (e.g. Riloff 1999), our method provides an in-depth inference model of a growing cognition, not merely a labeling of attitudes in text.

## Context Activation in Scone KB

In this section, we will briefly introduce Scone's context representation, which is the basis of our modeling tool. Scone is designed to be a practical KB system with emphasis on its expressiveness, ease of use, scalability, and the efficiency of the most commonly used operations for search and inference. Regarding these goals, Scone provides default reasoning with exceptions[3].

At the most basic level, Scone can be viewed as a semantic network representation, with nodes representing entities and links representing relations or statements tied

[3] For a discussion on the inference mechanism of Scone: http://sef-linux.radar.cs.cmu.edu/nuggets/?p=34

to these entities. An entity is not necessarily a physical object or type of object; it might instead be a type of action (e.g. a "telling" action) or a specific individual action: "John told X to Mary." X may be a single statement or an arbitrarily complex collection of statements (bundled together into a Scone context).

At a higher level, the types in Scone may be viewed as frames (Minsky 1975) or descriptions. Each such description has some type-restricted slots or roles, plus some statements about those roles and their relationships. When we create an instance of one of these types, we fill in some or all of the roles with specific entities, and we inherit all the other structure and roles. By the rules of inheritance, each instance behaves as a *clone* or *virtual copy* (Fahlman 1979) of the generic description. The generic "telling" action has roles for the speaker and listener (both are people), and a package of information that is conveyed; we then can create a specific instance of "telling" in which John is the teller, Mary is the listener, and the content X is filled in.

### Context Representation in Scone KB

A multi-context and context activation mechanism has been designed into Scone using the marker-passing algorithm (Fahlman 2006). In general, a context represents some state of the universe. In our case, it is used to represent the state of mental attitudes. Each node in the semantic network is tied to a context-node representing the context in which that entity exists; each link in the network is tied to a context-node representing the context in which that statement or relation is true.

In Scone, the context nodes are treated like other nodes in that they are also tied into their own inheritance hierarchy using "sub-context" links. To say "Context C2 is a sub-context of context C1" is to say that C2 behaves initially as a *clone* of C1, with the same contents. Then we can make specific modifications to C2, adding information or subtracting information that would otherwise be inherited from C1. Reasoning is always done with respect to an active context. So when we reason within C2, we see the effect of these new additions; when we reason within C1, they are invisible. Although we use inheritance to acquire the proper behavior, the relation between the two contexts is neither "is-a" nor "part-of", but something more like "is a clone of, modulo explicit changes".

At any given time, we can *activate* a context C, and reason about what is true in C. Activating C has the effect of activating all the context nodes above C in the type hierarchy, and all the nodes and links that are tied to these active contexts. All the nodes and links in inactive contexts will not appear in subsequent inference.

Contexts can also be used to represent the state of a changing world at different times. Each of the contexts represents a mental attitude at a specific time; it begins as a clone of the state in the previous time-step, and then we can add or remove a few items if necessary.

Every generic action or event, in addition to its other roles, defines two temporal contexts, one representing the world before the event and one representing the world after it. For example, in the before-context of a "telling" action, the speaker knows the information X; in the after-context, the listener is also assumed to know X. Both of these contexts inherit all the information from some surrounding or background context, so we don't have to explicitly copy everything we know about the world.

## Mental Context Representation

In this section, we discuss a multi-context architecture on which our later inference mechanism has been built. Our input to the model is a list of mental context operations extracted from text. Each of the operations corresponds to one psych-term. By semantic decomposition, we break the complex semantics of a psych-term into a set of atomic operations on single mental contexts. These contexts are organized in a hierarchical structure, which gives us a simplified representation of the human memory.

### Semantic Decomposition

The semantics of psych-terms are projected onto the context network through semantic decomposition. For example, one sense of the word "pretend" can be represented as "X is not true in reality and person P1's belief, but P1 wants person P2 to believe it" (definitions are restricted to mental contexts). This example involves several contexts: the reality, the belief of P1, the intention of P1, the belief of P2 under the intention of P1, as well as the before context and the after context of "pretend". Notice that there can be other psychological terms (e.g. "want") in the definition of "pretend", which involves other sets of atomic mental context operations.

### Mental Context Network

Semantic decomposition helps us transform psych-terms from text into a set of mental context operations that updates the structure of the context network. There are two issues in this update: (1) a context inheritance chain which represents the history of a single mental context on a timeline; (2) a hierarchical structure which organizes multiple kinds of mental context by events and agents.

In our representation, we model the mental contexts as roles attached to entities. For example, little red-cap's belief is a belief context role of little red-cap. Meanwhile, she can have different belief contexts at different times. By default, each of the newly updated versions of one's belief would inherit from his/her most recent belief.

On the other hand, little red-cap may have multiple types of mental contexts. She can have belief, intention, and her own perception of the reality; she can also have the wolf's intention under her own belief, which might not agree with the wolf's true intention. Figure 1 shows a part of the context network built from the "Little Red-Cap" story. It illustrates three basic aspects of the context structure:

- By default, the mental contexts would inherit from a general context or a particular context for a story which holds a set of basic knowledge.
- Mental contexts can be organized by events. A typical mental event has an agent whose mental contexts are changed as an effect of the event. Each psych-term would be mapped to one of the mental events. When we retrieve an event, a set of related contexts of that event would also be retrieved.
- The mental contexts are environment-sensitive. For example, the wolf's intention can be different from little red-cap's belief of the wolf's intention.

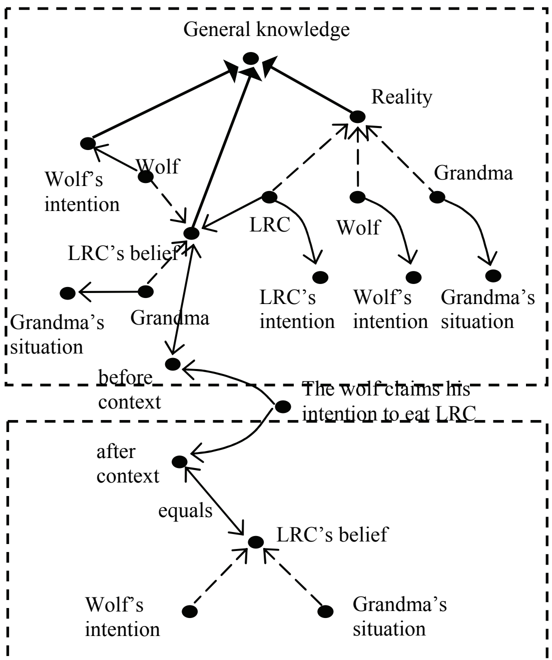


**Figure 1. A context network from the "Little Red-Cap" story. LRC is the short form for "little red-cap". Arrows with big solid heads represent "sub-context" relation. Double-headed arrows represent "equals" relation. Arrows with dotted lines represent "in-context" relation.**

## Representing Dormant Memory

There are some psych-terms involving inactive knowledge in the memory, which may later on be activated by some events. For the term "remember", if we ask questions of whether person P knew statement X in the past or not, the model should answer "yes", since the knowledge does exist in the memory, but was dormant. The effect of "remember" is to reload that inactive knowledge back into the focus. Table 1 shows the semantic decomposition of "remember" and "forget" in their usual sense.

Dormant knowledge is implemented by setting a dormant knowledge context under a target context. Dormant knowledge can then be tied to this context. The knowledge inside the dormant knowledge context would not be visible unless the context itself is activated. Hence, the inactive knowledge can be distinguished from knowledge that is never present. At some point, dormant knowledge can be brought to the active knowledge context by removing the knowledge from dormant knowledge context and adding it back to active knowledge context. These actions can be triggered by the "remember" and "forget" events. Figure 2 shows the representation of dormant context and its relation with the parent belief context.

**Table 1: The decomposed semantics of forget and remember.**

| **Psych-term** | **Decomposed Semantics** |
|---|---|
| P forget X | X exists in the "before" mental context of P; it is marked as inactive in the "after" context. |
| P remember X | P must previously have known X and then forgotten it (see above).<br>In the "after" context, X is active again. |

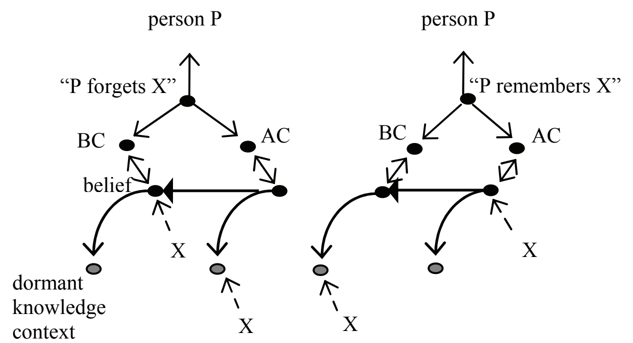


**Figure 2. Dormant knowledge. Grey nodes represent dormant knowledge contexts; "AC" stands for "after context"; "BC" stands for "before context".**

## Modeling Inter-Contextual Activities

The multi-context representation we have introduced serves as a basis for further inference of psychological activities. In this section, we explain how the inter-contextual activities could be modeled using the multi-context mechanism.

In general, there are several types of interactions between mental contexts: mental reactions caused by perception of the reality, inter-agent mental interactions, and self-retrospection. In our model, one person does not directly interact with another person's mental states. Instead, he builds a model of the other's mental states according to his perception of the reality. We generally avoid accessing multiple agents' mind through a third party that holds some common knowledge. The interactions among mental contexts are handled through inter-contextual rules.

In our representation, reality and various mental contexts are treated very similar under the multi-context mechanism.

Different instances of these contexts are well organized in a dynamic context structure. We could then constrain the behaviors of different mental contexts under different mental events using inter-contextual rules. Once a mental event happens, the related mental contexts would check and modify their own structures and contents based on the new information. Usually this self-adjustment can be achieved by a realization of a difference between the external world and the belief, assumption or expectation. According to this, newly updated mental contexts would be constructed. Figure 3 shows an example of how little red-cap changes her mind about the world according to the wolf's intention and actions. We use a simple rule saying that when a conflict is detected between the perceived reality and mental contexts, build new beliefs according to the perceived reality (this is a default rule; if the perception turns out to be illusion, the program would go back and revise the context). The depicted scenario is the same as in Figure 1, but the diagram shows another dimension of the inference from the perspective of inter-contextual interactions.

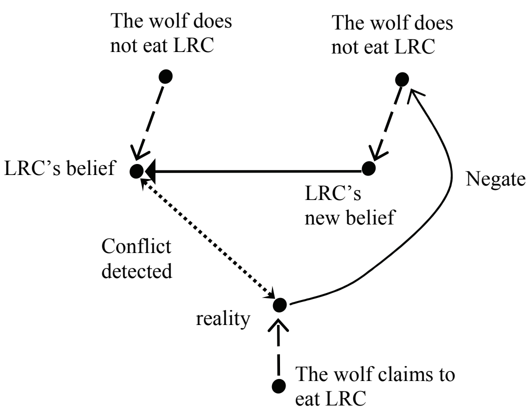


**Figure 3. Inter-Contextual Activities.**

## Story Understanding by Mental Context Reasoning

The Grimm's Fairy Tales contains many interesting psychological activities that can be studied. From these, we choose "Little Red-Cap" that involves dormant memory and mental context interactions. This experiment demonstrates the capability of our model to understand the story, remember what has happened, and reason about the environment and the mental states of the characters in the story. The input to the model is a set of mental operations translated from English. As the story goes on, we will ask questions about the mental states to the model and retrieve its answer. The mental context operations are presented in the form of pseudo code[1]; the model outputs are marked by "=>". To focus on the interesting cases, we only present the mentality-related parts of the story.

[1] The Lisp form of the code and the actual program output are available at http://www.cs.cmu.edu/~weichen/examples/020.lisp.

One day her mother <u>said</u> to her, … Take them to your grandmother, she is ill and weak…do not run off the path, ...

```
  New-event: "expect"
    Agent := "redcap"
    Statement := "grandma in bed"
  New-event: "suggest"
    Agent := "mom"
    Patient := "redcap"
    Statement := "do not run off the path"
    => suggestion accepted
  In-context := "should-do" of "redcap"
      True?: "run off the path"
      => No
```

… and just as little red-cap entered the wood, a wolf met her. Red-cap <u>did not know</u> what a wicked creature he was ...

```
  New-didn't-know: Statement: "wolf is bad"
        Agent: "redcap"
    In-context: "belief" of "redcap"
        True?: "wolf is bad"
        => No
  In-context: "reality"
    True?: "wolf is bad"
    => Yes
```

The wolf <u>thought</u> to himself, … she will be better to eat than the old woman. I must act craftily, so as to catch both. … and then he <u>said</u>, "see little red-cap, how pretty the flowers are about here. Why do you not look round…" … and so she ran from the path into the wood to look for flowers.

```
  New-event: "intend"
    Agent := "wolf"
    Statement := "wolf eat redcap and grandma"
  New-event: "suggest"
    Agent := "wolf"
    Patient := "redcap"
    Statement := "run off the path"
  => Conflict detected: "redcap" should "not run
off the path"
  New-accept: "suggest"
    Agent := "redcap"
    Statement := "run off the path"
In-context: "belief" of "redcap"
  In-context: "intention" of "wolf"
    True?: "wolf eat redcap and grandma"
    => No
In-context "intention" of "wolf"
  True?: "wolf eat redcap and grandma"
  => Yes
In-context: "should-do" of "redcap"
  True?: "run off the path"
  => Yes
  In-context: "dormant-knowledge"
    True?: "run off the path"
    => No
```

She was <u>surprised</u> to find the cottage-door standing open …

```
  New-event: "surprise"
    Agent := "redcap"
    Statement := "cottage door is open"
```

```
In-context : "previous belief" of "redcap"
    True?: "cottage door is open"
    => No
```

There lay her grandmother … "Oh, but, grandmother, what a terrible big mouth you have." "The better to eat you with."

```
  In-context: "belief" of "redcap"
    True?: "grandma in bed"
    => Yes
    True?: "grandma has big mouth"
    => No
    True?: "wolf eat redcap"
    => No
  New-event: "claim"
    Agent := "wolf"
    Patient := "redcap"
    Statement := "wolf eat redcap"
  => Conflict detected: "wolf do not eat redcap".
Accepted claim by default.
  In-context: "belief" of "redcap"
    True?: "wolf eat redcap"
    => Yes
  In-context: "previous belief" of "redcap"
  True?: "wolf eat redcap"
    => No
```

## Conclusion

The main contributions of this paper include:

- A multi-mental-context network that represents various mental states.
- An inter-contextual inference mechanism which performs reasoning based on new information and a multi-modal memory.

The major features that distinguish our approach from other cognitive models include:

(1) Our mental activity representation comes out of a general-purpose and efficient modeling methodology, which uses "context" to model mental states.

(2) The mental state representation consists of two parts: a multi-mental-context architecture and inter contextual rules that guide the behaviors (interaction and search) between different mental contexts.

(3) We are currently not dealing with goal driven behaviors, but to model reactive mental state changes. However, we believe that goal driven behaviors are likely to be incorporated into our model by modifying and adding sets of rules for mental context communication and actions.

## Acknowledgments

Development of Scone has been supported in part by the Defense Advanced Research Projects Agency (DARPA) under contract numbers NBCHD030010 and FA8750-07-D-0185. Additional support for Scone development has been provided by generous research grants from Cisco Systems Inc. and from Google Inc.

## References

Akman, V. and M. Surav. 1996. Steps toward formalizing context. *AI Magazine*, 17(3):55-72.

Cohen, P. R. and H. J. Levesque. 1990. Rational Interaction as the basis for communication. In *Intentions in Communication*. MIT Press, Cambridge, MA.

Dyer, M. G. 1983. *In-depth understanding: A computer model of integrated processing for narrative comprehension*. MIT Press. Cambridge, MA.

Fahlman, S. E. 1979. *NETL: A System for Representing and Using Real-World Knowledge*. MIT Press. Cambridge, MA.

Fahlman, S. E. 2006. Marker-passing Inference in the Scone Knowledge-Based System. *First International Conference on Knowledge Science, Engineering and Management*. Guilin, China.

Ghidini C. and F. Giunchiglia. 2001. Local models semantics, or contextual reasoning = locality + compatibility. *Artificial Intelligence*, 127(2):221-259.

Gordon A. S and J. R. Hobbs. 2004. Formalizations of Commonsense Psychology. *AI Magazine*, 25:49-62.

Jago, M. 2006. Modeling Assumption-based Reasoning using Contexts. In *2nd International Workshop on Context Representation and Reasoning.* Riva del Garda, Italy.

McCarthy, J. 1987. Generality in artificial intelligence, *Communications of the ACM*, 30(12):1030-1035.

McCarthy, J. 1995. Making robots conscious of their mental states. *Machine Intelligence*, 15:3-17.

Minsky, M. 1975. A framework for representing knowledge. In *The Psychology of Computer Vision*, pp. 211-277. Winston, P. (ed). McGraw-Hill. New York, NY.

Mueller, E. T. 1990. *Daydreaming: In Humans and Machines: A Computer Model of the Stream of Thought.* Ablex Publishing Corporation. Norwood, NJ.

Mueller, E. T. 2004. Understanding script-based stories using commonsense reasoning. In *Cognitive Systems Research*, 5(4):307-340.

Rao, A. and M. Georgeff. 1995. BDI Agents: From Theory to Practice. *Proceedings of the First International Conference on Multiagent Systems*. San Francisco, CA.

Riloff, E. 1999. Information extraction as a stepping stone toward story understanding. In *Understanding language understanding: Computational models of reading*, pp. 435-460. MIT Press, Cambridge, MA.

Schank, R. C. and R. P. Abelson. 1977. *Scripts, plans, goals, and understanding: An inquiry into human knowledge structures*. Lawrence Erlbaum, Hillsdale, NJ.

Serafini, L. and P. Bouquet. 2004. Comparing formal theories of context in AI. *Artificial Intelligence,* 155:41-67.

Singh, Push. 2005. *EM-ONE: An architecture for Reflective commonsense thinking*. PhD Thesis. MIT. Cambridge, MA.