IRST

ISTITUTO PER LA RICERCA
SCIENTIFICA E TECNOLOGICA

38050 Povo (Trento), Italy
Tel.: + 39 461 314575 · Fax: + 39 461 314591
e-mail: prdoc@itc.it · url: http://www.itc.it

Contextual Reasoning

Fausto Giunchiglia

November 1992

Technical Report # 9211-20

Contextual Reasoning

Fausto Giunchiglia
IRST - Istituto per la Ricerca Scientifica e Tecnologica
38050 Povo - Trento, Italy
University of Trento, Via Inama 5, Trento, Italy
fausto@irst.itc.it

Abstract

It is widely agreed on that most cognitive processes are *contextual* in the sense that they depend on the environment, or context, inside which they are carried on. Even concentrating on the issue of contextuality in reasoning, many different notions of context can be found in the Artificial Intelligence literature. Our intuition is that reasoning is usually performed on a subset of the global knowledge base. The notion of context is used as a means of formalizing this idea of localization. Roughly speaking, we take a context to be the set of facts used locally to prove a given goal plus the inference routines used to reason about them (which in general are different for different sets of facts). Our perspective is similar to that proposed in [McC87, McC91].

The goal of this paper is to propose an epistemologically adequate theory of reasoning with contexts. The emphasis is on motivations and intuitions, rather than on technicalities. The two basic definitions are reported in appendix A.

Ideas are described incrementally with increasing level of detail. Thus, Section 2 describes why contexts are an important notion to consider as part of our ontology. This is achieved also by comparing contexts with situations, another ontologically very important concept. Section 3 then goes more into the technical details and proposes that contexts should be formalized as particular mathematical objects, namely as logical theories. Reasoning with contexts is then formalized as a set of deductions, each deduction carried out inside a context, connected by appropriate "bridge rules". Finally, Section 4 describes how an important example of common sense reasoning, reasoning about reasoning, can be formalized as multicontextual reasoning.

1 Contexts

We take the explicitly known facts (awioms or explicitly derived consequences) plus the machinery used to reason about them to be all and only the state of a reasoning individual. Notice that we take a fairly restrictive notion of state. For instance our ontology does not take into consideration all those global *internal* variables, such as the state of the stack or the commitment to achieve a certain goal, which usually affect the behaviour of a program or a human. Neither do we consider the *external* state of the outside world, as conveyed by the sensors. With our approach, given an appropriate choice of the language and the known facts, we are able to talk and reason about sensors and internal variables; however they are not part of the basic notion of state.

The basic intuition underlying the work described in this paper is that reasoning is always local to a subset of the known facts. We never consider all we know but rather a very small subset of it. This small subset is what determines the context of reasoning. We therefore take a *context c to be that* **subset of the complete state** *of an* **individual** *that is used for reasoning about a given goal.*

A context is a *theory* of the world which encodes an individual's subjective perspective about it. It is a *partial theory* as the individual's complete description of the world is given by the set of all the contexts. It is an *approximate theory* (in the sense described in [McC79]) as we never describe the world in full detail. A detailed description of the world is in general impossible, and, even when possible, seems a very bad idea (as it would force us to consider a lot of irrelevant information). We may have different contexts which are theories of the same phenomenon and which describe it at different levels of approximation. Contexts can be partially ordered depending on the level of approximation. The "right" level of approximation depends, among other things, on the problem to be solved.

Contexts are not situations. Remember (following [MH69]) that *a situation s is* **the complete state** *of* **the universe** *at an instant of time.* Compare this definition with that of context. Notice also that some philosophers would argue that it does not make sense to talk of the "complete state of the universe" as the universe is "incomplete". This issue is not discussed here because irrelevant to the goals of the paper. It is worth noticing, however, that a situation records what can be talked about of the unierse at given instant of time, whether the universe is complete or incomplete *per sé* is not relevant.

A situation records the state of the world as it is, independently of how it is represented in the mind of the reasoner. A context, rather, is inside the reasoning individual. It is part of his state and, as such, it is responsible of his subjective view of the world. For instance a certain phenomenon can be described and reasoned about by using different predicates in different contexts, the same predicate may have different truth values in different contexts and so on. The level of approximation used in a context to describe reality is itself a subjective choice. Notice that philosophers have discussed for centuries about the distinction about object and subject, real world and mind and about their interaction, *e.g.* whether the object exists independently of the subject or it is generated by it. These issues do not play any role here. We are not interested in the first principles but rather in epistomologically adequate theories of common sense reasoning (using the terminology introduced in [MH69]).

A situation is complete in that it records *all* the state of the world. At the same time it has also a dimension of partiality along the temporal dimension as it works *only* for a precise instant of time. The complete (temporal) picture of the world is recorded by the sequence of situations it evolves through. A context is partial as it records only part of the state of an individual. It is at the same time complete as it records all the state which affects reasoning.

It is easy to think of axiomatizations where both situations and contexts are used. The evolution through situations will account for the evolution of the world through time, the evolution through contexts will account for the evolution of the reasoning process through different states. There are four possibilities.

The first is to take the state of the world at a given situation s and describe it as a set of contexts $c_1, ..., c_n$. Having multiple contexts describing one situation corresponds to the case where the reasoner uses multiple contexts, each being a different approximate theory of the same situation. We have evolution in the contexts used in the reasoning still considering the same situation. An example, suggested by John McCarthy, where this kind of axiomatization is useful is the following. Suppose you are planning a London-Glasgow- Moscow flight. You will consider the plane connection in Glasgow without testing whether you have the ticket. You do so on the basis of some law of inertia which suggests that you should have it as you had it in London just before taking off. But now, suppose that you are in the situation s_{12} where, in Glasgow, you are connecting flight and you realize that you have lost the ticket (an unexpected obstacle). In the context c_{35} which records your axiomatization of the situation s_{12} you will not be able to deal with the obstacle. Indeed it is very likely that you will not even be able to mention the formula $hasticket(John, s_{12})$ as $hasticket$ will not be part of the language of c_{35}. In order to solve this problem you will set up a new context c_{36} for the same situation s_{12} with $hasticket$ in the signature and with a state which axiomatizes a subset of the state variables of s_{12}. c_{36} will have to be less approximate than c_{35}, at least for what concerns the problem of the lost ticket. For instance one axiom of c_{36} will say that John does not have the ticket in Glasgow, *i.e.* $\neg hasticket(John, s_{12})$.

The second possibility is to associate contexts to situations and to describe each situation s_i with a context c_i. Each situation is axiomatized with a different context; that is the time evolution of the world corresponds one-to-one to the evolution of the reasoning process through partial states. [GW88, Giu91], for instance, take this approach. In particular in [Giu91] the temporal of evolution physical processes, *e.g.* the

flowing of the water from a container to another, is modeled qualitatively as a set of situations, each situation described by a context. This one-to-one mapping of situations into contexts is very useful as it allows to keep reasoning monotonic inside each context, to maintain local consistency (inside each context) and to achieve a form of nonmonotonicity by switching between contexts which possibly contain contradictory information. This kind of nonmonotonicity has been termed *inessential* in [GW88]. Thus, in the description of the water flowing from one container to another, we may get to a situation s_{15} (axiomatized by context c_{15}) where the level of the water in the container $C1$ should be at the same time strictly higher and strictly lower than that in the container $C2$ (in formulas $HIGHER(heightwater(C1), heightwater(C2), c_{15})$ and $\neg HIGHER(heightwater(C1), heightwater(C2), c_{15}))$. This of course cannot be and would cause c_{15} to be inconsistent. The solution is to switch to a new context c_{16} such that, for instance, $HIGHER(heightwater(C1), heightwater(C2), c_{15})$ and $\neg HIGHER(heightwater(C1), heightwater(C2), c_{16})$ but not $HIGHER(heightwater (C1), heightwater(C2), c_{16})$. The proof of $HIGHER(heightwater(C1), heightwater (C2), c_{16})$ is disallowed simply by adding a new appropriate axiom to c_{16}. The process is similar to that of building multiple extensions in default logic.
The third possibility is to have one context correspond to many situations. These axiomatizations correspond to the case where evolution in the world is not reflected in any change in the context considered in the reasoning. Almost all the work in the literature on nonmonotonic reasoning and the situation calculus can be described as taking this approach and considering only one context (see for instance Lifschitz' solution to the Yale shooting problem [Lif87]). Not using contexts can be described as using always the same context and by making the obvious simplification of dropping the extra context argument from all the application symbols. Differently from above, here we can make, in the same context, assertions about two distinct situations, *e.g.*, $HIGHER(heightwater(C1), heightwater(C2), s_{15}) \wedge \neg HIGHER(heightwater(C1), heightwater(C2), s_{16})$. The obvious drawback is that we are forced to consider all we know in any reasoning step.
The fourth case of many situations described by many contexts can be obtained as a combination of the cases described above.

2 Formalizing contexts

In section 1 we have taken a context to be that subset of the complete state of an individual which is used during a given reasoning process. The next step is to formalize this notion of "state" by saying more precisely what we mean by "context" and by "reasoning with contexts". The goal is to model reasoning as deduction. We show below how contextual reasoning can be formalized as deduction in a system allowing multiple first order theories. Appendix A gives the definition of the resulting formal system, called "Multilanguage System" (ML-System) and of the notion of deduction associated with it.

2.1 Reasoning inside a context

The starting point is that the set of facts which provide the context of reasoning is in general only a subset of the knowledge base. We therefore describe the knowledge base as structured into sets of facts, which we write $A_1, ..., A_n$. Taking a context to be any A_i would lead to a notion of context which is similar to the notion of a partition in partitioned data bases or to the similar notion of microtheories in CYC [Guh90]. Notice that these partitions would not need to be static and fixed once for all.

This does not seem a satisfactory enough solution. In fact, in general, each A_i is written using different sets of constants, predicate, and function symbols. For instance, the set of facts about arithmetic will have, as part of the signature, 0, +, and $*$, while the theory of how to get on a plane will use constants like $existsflight$ and $plane$. In the London-Glasgow- Moscow example the signature of c_{36} will have $hasticket$ in its signature while c_{35} will not. Basic beliefs and beliefs about beliefs have again different signatures: the second set need only contain the belief predicate and the names of formulas, but not the formulas themselves (see section 3). We therefore require that each context come with its own signature.

More interestingly, we also take the notion of wellformedness to be *localized* and distinct for each A_i. (Notice that in partitioned data/knowledge bases (*e.g.* CYC) the notion of wellformedness is taken to be the same everywhere.) This allows us, for instance, to have a context which is essentially a data base of atomic ground formulas and another whose facts express metalevel heuristics, all expressed in clausal form. Another interesting example (which has been implemented in `GETFOL`) is the following. We may have a context c_2 containing facts which are abstractions of the facts in c_1 [GW89]. c_2 can then be used do drive search in c_1. If we use one of the GPS abstractions [NSS63], then the language of c_2 allows only sets of atomic formulas.

We formalize the requirement that each A_i come with its own signature and wellformedness rules by associating a language L_i to each A_i. This amounts to saying that the language used to write a set of facts A_i, and not only the set of facts A_i, is part of the state used in a reasoning process.

The next step is to model reasoning. The standard solution is to have a unique inference engine (possibly consisting of a set of inference modules) which can be applied to any set of facts or, even, to combinations of them. Our proposal is to associate a distinct inference engine to each distinct set of facts A_i. This allows us to localize the form of reasoning and, for instance, to define special purpose inference engines which exploit the local form of wffs. For example, we can use PROLOG on clausal languages and set

inclusion in the GPS abstract context c_2.
If we call Δ_i the set of inference rules associated with a set of facts A_i, then we can define a context c_i to be the triple $c_i = \langle L_i, A_i, \Delta_i \rangle$. In other words, we take a context to be a logical theory, presented as an axiomatic formal system. This allows us to take the usual notion of deduction (see for instance [Pra65]) as the formalization of reasoning *inside* a context. The components of c_i, that is its language L_i, its set of known facts A_i, plus the explicitly derived theorems, the inference rules Δ_i are all and only the state that is used when reasoning about a given goal.

2.2 Reasoning with multiple contexts

A knowledge base contains in general a set of interacting contexts $c_1, ..., c_n$. We need to capture the idea that reasoning in one context may influence reasoning in other contexts. We introduce therefore a new set of rules which allow us to derive a fact in a context because we have derived another fact in another context. We can represent these "linking" rules as follows (in the case of single premises):

$$\frac{\langle A_i, c_i \rangle}{\langle A_j, c_j, \rangle}$$

where c_i and c_j are two different contexts, A_k is a formula written in the language of the context c_k. We call **bridge rules** the rules of the form above as they allow us to bridge deductions in c_i to deductions in c_j by allowing us to derive A_j in c_j just because we have derived A_i in c_i. We say also that A_j is a **justified assumption** of c_j as it is an assumption in c_j which is justified by a derivation in another context. The rules where premises and conclusion belong to the same language are called are called L_i rules.
One example of L_i-rule for a theory i is *modus ponens*

$$\frac{\langle A \to B,\ i \rangle \quad \langle A,\ i \rangle}{\langle B,\ i \rangle}$$

One example of bridge rule between the theories i, j and the theory k is *multicontextual modus ponens*

$$\frac{\langle A \to B,\ i \rangle \quad \langle A,\ i \rangle}{\langle B,\ k \rangle}$$

The meaning ot the two rules is very different. The first allows us to derive B inside the theory i just because we have derived $A \to B$ and A in the same theory. Thus, for instance, if we take the theory i to represent the beliefs of an agent a_i, this means providing the agent a_i with the ability of using modus ponens. Multicontextual modus ponens allows us to derive B in the theory k just because we have derived $A \to B$ in

the theory i and A in the theory j. If we take the theories i, j, k to represent the beliefs of the agents a_i, a_j, a_k, this means that a_k will be able to take B as known, not because he has derived it, but because this is the result of the interaction of the results derived by a_i and a_j. The assertion of B in k is not the result of a deduction in k but, rather, the result of the "propagation" of reasoning from i, j into k.
As another example, a notion similar to that captured by McCarthy's lifting axiom [McC91] can be formalized by the following bridge rule:

$$\frac{\langle A, c \rangle}{\langle ist(A, c), c' \rangle}$$

which intuitively says that, if we can prove A in context c, then we can prove (in context c') that we can prove A in c.

2.3 Some observations

Let us consider the interaction between the reasoning in two distinct contexts $c_1 = \langle L_1, A_1, \Delta_1 \rangle$, $c_2 = \langle L_2, A_2, \Delta_2 \rangle$ (the case with more than two contexts is a trivial generalization). The interaction between two contexts inside a multicontext system can be formalized provided the following conditions hold:

1. a subset of the wffs in L_1, let us call it L_{s1}, can be put in relation with a subset of the wffs in L_2, let us call it L_{s2};

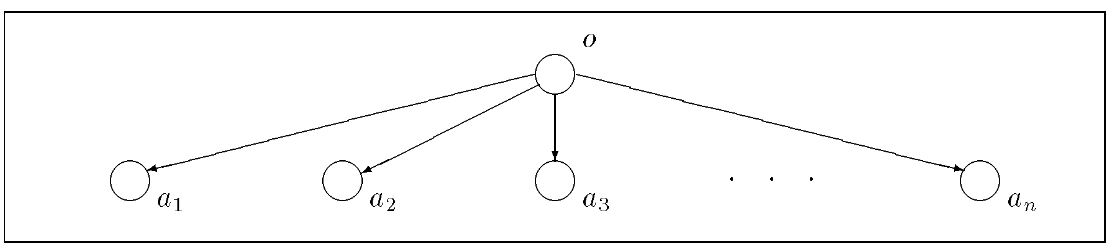
2. there exist bridge rules between c_1 and c_2.

In general we can formalize the relationship between L_{s1} and L_{s2} via a general mapping function rew which rewrites the elements of L_{s1} into the elements L_{s2}. rew is often total and surjective and sometimes injective. A not injective rew is used in abstraction to map ground formulas into abstract formulas [GW89]. For instance in GPS abstractions both $\alpha \vee \beta$ and $\alpha \wedge \beta$, where α and β are atomic wffs, are mapped into $\{\alpha, \beta\}$.
The relation between L_1 and L_2 allows us to link the semantics of c_1 to the semantics of c_2. Via the relation between the languages, we can formalize the links existing between the models of c_1 and those of c_2. Thus, for instance, in the London- Glasgow-Moscow example introduced in section 1 we can say that $John$ in c_{35} is the same $John$ as that in c_{36}. As a more complicated example we can also say that $gate15$ (used to assert $at(John, gate15, s_{12})$ in c_{36}) corresponds to $GlasgowAirport$ (used to assert $at(John, GlasgowAirport, s_{12})$ in c_{35}). The intuition is that c_{36} is a less approximate theory of s_{12} than c_{35} and that we need this more precise description in order to solve the problem (*e.g.* John will buy the ticket at the gate number 15).
The relationship between the languages constrains the kind of bridge rules which exist between the two contexts c_1 and c_2 (the premises of a bridge rule must belong to L_1

while its conclusion must belong to L_2). The existence of bridge rules (second condition) allows us to achieve proof theoretically the connection that we have semantically achieved via the connection between the languages. Thus for instance, in the London- Glasgow-Moscow examples we will have a bridge rule which will allow us to assert $at(John, gate15)$ in c_{36} from a proof of $at(John, GlasgowAirport)$ in c_{35} and another which will allow us to assert $existsflight(Glasgow, Moscow, s_{12})$ in c_{36} from a proof of $existsflight(Glasgow, Moscow, s_{12})$ in c_{35}.

3 Reasoning about reasoning

In the case of reasoning about reasoning, the mapping function rew is such that $rew(\alpha) = \pi(\text{“}\alpha\text{”})$; in other words rew maps a formula α into a predefined unary predicate π which takes the name of α, that is “α”, as argument. Sometimes π takes a second argument which is the name of the context α belongs to; this distinction is not relevant in this context. π can be instantiated to various predicates. For instance we can take π to be the provability predicate Th, in which case c_2 is a metatheory of c_1 (see for instance [GT91a]). We can also take π to be the belief predicate $\mathcal{B}$, in which case c_2 will be the context of the beliefs about c_1; we will study this example in detail in the following. π can also be taken to be the defeasible belief predicate CBB (where CBB stands for “Can Be Believed”) as it has been done in [GW88, Giu91]). Finally, we can also take π to be McCarthy's ist predicate which takes a context c and a proposition p and is such that $ist(p, c)$ means that p is true in c [McC91].
Let us consider more in detail the case of belief in a multiagent environment. As we consider the multiagent case, not only will the computer be able to have beliefs about its own beliefs but also about the beliefs of other agents. Let us suppose that we have a situation with a (possible infinite) set $\{a_i\}_{i \in I}$ of agents a_i. Each agent is associated with a context (called a_i-context) which represents his beliefs; we thus say that a_i believes a formula if that formula is provable in the a_i-context. We also have a computer observer o which is himself associated the context (called o-context) of his beliefs. Differently from the agents, o “sees” the agents' beliefs (namely all the a_i-contexts) and is able to reason about them. This is formalized by putting the o-context “on top” of the a_i-contexts, *i.e.* by defining appropriate bridge rules which allow all a_i's beliefs to propagate up into o. The following picture summarizes this idea.



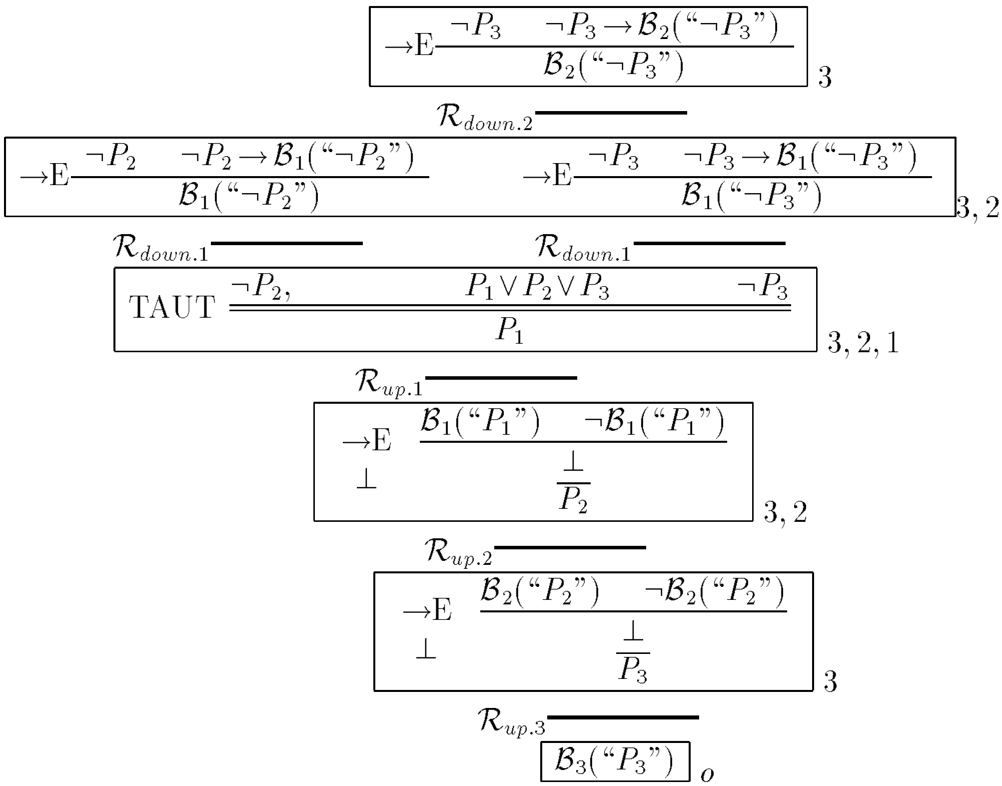
Let us concentrate on the bridge rules. The computer observer o will have a belief predicate $\mathcal{B}_i$ for each agent a_i. The idea is that o believes $\mathcal{B}_i(``\text{``}A''")$ whenever the agent a_i believes A. *The ability of o to see a_i's beliefs is formalized by defining a form of reflection up from each agent into o*: for any wff A an agent a_i will believe, o will thus be able to prove $\mathcal{B}_i(``A")$. *The fact that $o$ believes only facts which correspond to the agents' beliefs is captured by defining a form of reflection down from $o$ into each agent.* This will prevent $o$ from believing sentences of the form $\mathcal{B}_i(``A")$ where A is not provable in the a_i-context. This amounts to considering the following two bridge rules:

$$\mathcal{R}_{up.o} \quad \frac{\langle A,\ a_i\rangle}{\langle \mathcal{B}_i(A),\ o\rangle} \qquad\qquad \mathcal{R}_{down.i} \quad \frac{\langle \mathcal{B}_i(A),\ o\rangle}{\langle A,\ a_i\rangle}$$

For the purpose of this paper we need to consider the case where each agent may have beliefs about its own beliefs and about the beliefs of the other agents and so on indefinitely. To formalize this situation, it is sufficient to expand the context of each agent a_i in the previous figure with the figure itself and so on with the resulting contexts. The result is a balanced tree with infinitely long branches. [GS91] describes the technical results about this system. For instance it shows that we need the restriction that $\mathcal{R}_{down.i}$ can be applied only if $\langle A,\ a_i\rangle$ depends only on assumptions made in contexts above a_i. It also proves that the resulting system is equivalent to multimodal K (that is modal K with multiple modal operators) and it discusses how it relates to Konolige's logics of belief [Kon86, Kon85]. [GS94] studies the proof theory of this and other related systems in detail.
This system, extended with two bridge rules which formalize common belief [GS91] has been used to prove the three wise men problem. As an example, let us consider part of this proof. The variant of the statement of the three wise men puzzle we have considered is as follows:

> A certain king wishes to test his three wise men. He arranges them in a circle so that they can see and hear each other and tells them that he will put a white or black spot on each of their foreheads but that at least one spot will be white. In fact, all three spots are white. He then repeatedly asks them, "Do you know the colour of your spot?". What do they answer?

The solution is that they answer *"No"*, the first two times the question is asked, and answer *"Yes"* thereafter (see [McC90, BP91] for alternative descriptions of this puzzle). The following is part of the proof.



In the proof, P_i is the assertion that the i-th wise man has a white spot on his hat. Context o is the top context, the facts provable in this context are the computer's beliefs. There are three agents named with the three natural numbers 1, 2 and 3; thus, for instance, context 3 is the third wise man, the facts provable here are his beliefs (in o's view). At level three in the tree there are the agents' beliefs about the agents' beliefs; thus, for instance, context $3, 2$ is wise man 3's view of the beliefs of wise man 2 and so on. A similar argument applies to the contexts at level 4 (which is the deepest level needed in the proof), for instance to the context $3, 2, 1$ in figure.

The part of the proof listed above ends with the theorem that o believes that the third agent believes that he has a white spot on his hat. The particular proof steps are not relevant. Notice on the other hand the following facts. As a notational convention, the boxes surround a reasoning session inside a context. Each box is labeled by the name of the context where the session is carried out. We may have multiple reasoning sessions inside the same context (as it happens, in the proof above, with the contexts 3 and $3, 2$). The box notation has been adopted to capture the intuition that reasoning inside one context is completely isolated from the reasoning done in the other contexts.

The reasoning inside one context is performed as if this were the only context. In the example, the reasoning inside all the contexts if first order and monotonic and it amounts to multiple applications of modus ponens, one application of the reduction ad absurdum inference rule (in the second session in context 3) and, in context $3, 2, 1$ to the application of a tautology decision procedure (which may not be applicable in the other contexts).
In the example, each box is linked to one or more boxes via the application of a reflection rule. In the general case a deduction inside a multicontext system can be described formally as a set of deductions inside contexts, any two or more deductions being linked by two or more applications of bridge rules (see Appendix A).
Multicontext systems have been implemented inside `GETFOL` [GT91b], an interactive theorem prover which has been implemented on top of a re-implementation of the `FOL` system [GW91, Wey80]. In particular the three wise men problem has been machine proved within `GETFOL`. The machine proof of the proof listed above is reported and discussed in Appendix B.

Acknowledgements

The discussions and work with Paolo Bouquet, Alessandro Cimatti, Tom Costello, Enrico Giunchiglia, Bob Kowalski, Vladimir Lifschitz, John McCarthy, Luciano Serafini and Alex Simpson have been the source of many valuable insights.
This work is part of MAIA, an integrated AI project under development at IRST.

References

[BP91] A. Bottani and C. Penco. *Significato e teorie del linguaggio.* Franco Angeli, Milano, Italy, 1991. Italian translation of J. Barwise, *Scenes and other situations.*

[Giu91] E. Giunchiglia. Towards a logical treatment of qualitative reasoning. Technical Report 91-0002, DIST - University of Genova, Genoa, Italy, 1991. Short version appeared in the proceedings of the 'First European Workshop on Qualitative Reasoning about physical systems' - Genoa (Italy) - January 1991.

[GS91] F. Giunchiglia and L. Serafini. Multilanguage first order theories of propositional attitudes. In *Proceedings 3rd Scandinavian Conference on Artificial Intelligence*, pages 228–240, Roskilde University, Denmark, 1991. IOS Press. Also IRST-Technical Report 9001-02, IRST, Trento, Italy.

[GS94] F. Giunchiglia and L. Serafini. Multilanguage hierarchical logics (or: how we can do without modal logics). *Artificial Intelligence*, 65:29–70, 1994. Also IRST-Technical Report 9110-07, IRST, Trento, Italy.

[GT91a] F. Giunchiglia and P. Traverso. Reflective reasoning with and between a declarative metatheory and the implementation code. In *Proc. of the 12th International Joint Conference on Artificial Intelligence*, pages 111–117, Sydney, 1991. Also IRST-Technical Report 9012-03, IRST, Trento, Italy.

[GT91b] F. Giunchiglia and P. Traverso. `GETFOL` User Manual - `GETFOL` version 1. Manual 9109-09, IRST, Trento, Italy, 1991. Also DIST Technical Report 91-0005, DIST, University of Genova.

[Guh90] R.V. Guha. Microtheories and contexts in cyc. Technical Report ACT-CYC-129-90, MCC, Austin, Texas, 1990.

[GW88] F. Giunchiglia and R.W. Weyhrauch. A multi-context monotonic axiomatization of inessential non-monotonicity. In D. Nardi and P. Maes, editors, *Meta-level architectures and Reflection*, pages 271–285. North Holland, 1988. Also DIST Technical Report 9105-02, DIST, University of Genova, Italy.

[GW89] F. Giunchiglia and T. Walsh. Abstract Theorem Proving. In *Proc. of the 11th International Joint Conference on Artificial Intelligence*, pages 372–377, 1989. Also IRST-Technical Report 8902-03 and DAI Research Paper No 430, University of Edinburgh.

[GW91] F. Giunchiglia and R.W. Weyhrauch. `FOL` User Manual - `FOL` version 2. Manual 9109-08, IRST, Trento, Italy, 1991. Also DIST Technical Report 91-0006, University of Genova.

[Kon85] K. Konolige. A Computational Theory of Belief Introspection. In *Proc. of the 9th International Joint Conference on Artificial Intelligence*, 1985.

[Kon86] K. Konolige. *A deduction model of belief.* Pitman, London, 1986.

[Lif87] V. Lifschitz. Formal Theories of Action. In F.M. Brown, editor, *The Frame Problem in Artificial Intelligence*, pages 35–58. Morgan Kaufmann, Los Altos, CA, 1987.

[McC79] J. McCarthy. Ascribing Mental Qualities to Machines. In M. Ringle, editor, *Philosophical Perspectives in Artificial Intelligence*, pages 161–195. Humanities Press, 1979. Also in V. Lifschitz (ed.), *Formalizing common sense: papers by John McCarthy*, Ablex Publ., 1990, pp. 93–118.

[McC87] J. McCarthy. Generality in Artificial Intelligence. *Communications of ACM*, 30(12):1030–1035, 1987. Also in V. Lifschitz (ed.), *Formalizing common sense: papers by John McCarthy*, Ablex Publ., 1990, pp. 226–236.

[McC90] J. McCarthy. Formalization of Two Puzzles Involving Knowledge. In V. Lifschitz, editor, *Formalizing Common Sense - Papers by John McCarthy*, pages 158–166. Ablex Publishing Corporation, 1990.

[McC91] J. McCarthy. Notes on Formalizing Context. Unpublished, 1991.

[MH69] J. McCarthy and P. Hayes. Some Philosophical Problems from the Standpoint of Artificial Intelligence. In B. Meltzer and D. Michie, editors, *Machine Intelligence 4*, pages 463–502. Edinburgh University Press, 1969. Also in V. Lifschitz (ed.), *Formalizing common sense: papers by John McCarthy*, Ablex Publ., 1990, pp. 21–63.

[NSS63] A. Newell, J.C. Shaw, and H.A. Simon. Empirical explorations of the logic theory machine. In Fiegenbaum and Feldman, editors, *Computers & Thought*, pages 134–152. McGraw-Hill, 1963.

[Pra65] D. Prawitz. *Natural Deduction - A proof theoretical study*. Almquist and Wiksell, Stockholm, 1965.

[Wey80] R.W. Weyhrauch. Prolegomena to a Theory of Mechanized Formal Reasoning. *Artificial Intelligence*, 13(1):133–176, 1980.

Appendix A: Multilanguage Systems

An axiomatic formal system S is usually described as a triple consisting of a language L, a set of axioms $\Omega \subseteq L$ and a set of inference rules Δ, *i.e.* $S = \langle L, \Omega, \Delta \rangle$. The generalization is to take many languages and many sets of axioms while keeping one set of inference rules. We thus have the following definition of multi-language system:

Definition 1 (Multi Language System) *Let I be a set of indices, $\{L_i\}_{i \in I}$ a family of languages and $\{\Omega_i\}_{i \in I}$ a family of sets of wffs, such that $\Omega_i \subseteq L_i$. A* Multi-Language Formal System (ML System) *MS is a triple $\langle \{L_i\}_{i \in I}, \{\Omega_i\}_{i \in I}, \Delta \rangle$ where $\{L_i\}_{i \in I}$ is the* Family of Languages, *$\{\Omega_i\}_{i \in I}$ is the* Family of sets of Axioms *and Δ is the* Deductive machinery *of MS.*

We write $\langle A,\ i \rangle$ to mean A and that A is an L_i-wff. The deduction machinery Δ is therefore defined as a set of inference rules, written as:

$$\frac{\langle A_1,\ i_1 \rangle \ldots \langle A_n,\ i_n \rangle}{\langle A,\ i \rangle}\ \iota \tag{1}$$

or as:

$$\frac{\langle A_1,\ i_1 \rangle \quad \ldots \quad \langle A_m,\ i_m \rangle \quad \begin{array}{c}[\langle B_1,\ j_1 \rangle]\\ \langle A_{m+1},\ i_{m+1} \rangle\end{array} \quad \cdots \quad \begin{array}{c}[\langle B_{n-m},\ j_{n-m} \rangle]\\ \langle A_n,\ i_n \rangle\end{array}}{\langle A,\ i \rangle}\ \delta \tag{2}$$

Picture (2) represents a rule δ discharging the assumptions $\langle B_i,\ j_i \rangle, \ldots, \langle B_{m-n},\ j_{m-n} \rangle$. The rules whose premises, conclusions, and, when existing, discharged assumptions belong to the same language L_i are called *L_i-rules*, the others *bridge rules*. Each context c_i is defined as $c_i = \langle L_i, A_i, \Delta_i \rangle$ where Δ_i is the set of *L_i-rules*.

Definition 2 (Deduction ... depending on) *A formula-tree is a* deduction in MS *of* $\langle A,\ i \rangle$ depending on *a set of formulas according to the following rules:*

1. *if $\langle A,\ i \rangle$ is an axiom of MS i.e., A is an element of Ω_i, then $\langle A,\ i \rangle$ is a deduction in MS of $\langle A,\ i \rangle$ depending on the empty set;*
2. *if $\langle A,\ i \rangle$ is not an axiom of MS, then $\langle A,\ i \rangle$ is a deduction of $\langle A,\ i \rangle$ depending on $\{\langle A,\ i \rangle\}$*
3. *if Π_k is a deduction of $\langle A_k,\ i_k \rangle$ depending on Γ_k for any $(1 \leq k \leq n)$, then*

$$\frac{\Pi_1 \quad \ldots \quad \Pi_n}{\langle A,\ i \rangle}\ \iota$$

is a deduction of $\langle A,\ i \rangle$ depending on Γ, provided that ι is of the form (1) and Γ is the union of all Γ_k for $1 \leq k \leq n$; or ι is the form (2) and

$$\Gamma = \bigcup_{1 \leq k \leq m} \Gamma_k \cup \left(\bigcup_{m \leq k \leq n} (\Gamma_k - \{\langle B_{k-m},\, j_{k-m} \rangle\}) \right)$$

Definition 3 (Theorem) *A deduction* Π *is a deduction in MS of* $\langle A, i \rangle$ *from* Γ *if and only if* Π *is a deduction in MS of* $\langle A, i \rangle$ *depending on* Γ *or any subset of* Γ.
$\langle A, i \rangle$ *is a* derivable from Γ *in MS, which is denoted* $\Gamma \vdash_{MS} \langle A, i \rangle$ *if and only if there is a deduction of* $\langle A, i \rangle$ *from* Γ *in MS.*
A deduction of $\langle A, i \rangle$ *from the empty set is a proof of* $\langle A, i \rangle$. $\langle A, i \rangle$ *is* provable in *(a theorem of) MS, in symbols* $\vdash_{MS} \langle A, i \rangle$*, if and only if there exists a proof of* $\langle A, i \rangle$ *in MS.*

Deductions are trees of wffs built by starting from a finite number of assumptions and axioms, possibly belonging to distinct languages, and by applying a finite number of inference rules. Any deduction can be seen as *composed of sub-deductions in distinct languages*, obtained by repeated applications of L_i-rules, any two or more subdeductions being concatenated by one or more applications of bridge rules.

Appendix B: a GETFOL proof of the three wise men problem

The following is the GETFOL listing of the (part of the) proof reported above (the listing must be read top to bottom, left to right).

```
SWITCHCONTEXT C3;
RDW c CB look2at3;
LABEL FACT assumption3;
ASSUME not P3;
LABEL FACT looked2at3;
IMPE ^1 ^2;

SWITCHCONTEXT C32;
RDW c CB look1at3;
RDW C3 B2 looked2at3;
LABEL FACT looked1at3;
IMPE ^1 ^2;
RDW c CB look1at2;
LABEL FACT
assumption32;
ASSUME not P2;
LABEL FACT looked1at2;
IMPE ^1 ^2;

SWITCHCONTEXT C321;
RDW C32 B1 looked1at2;
RDW C32 B1 looked1at3;
RDW c CB commonbel;
TAUT P1 BY ^1 ^2 ^3;

SWITCHCONTEXT C32;
LABEL FACT
conclusion32;
RUP C321 B1("P1");
RDW C3 B2 tell12;
FALSEI ^1 conclusion32;
NOTE ^1 assumption32;

SWITCHCONTEXT C3;
LABEL FACT conclusion3;
RUP C32 B2("P2");
RDW c B3 tell23;
FALSEI ^1 conclusion3;
NOTE ^1 assumption3;

SWITCHCONTEXT c;
RUP C3 B3("P3");
```

In the above listing, each set of commands (separated from the others by an empty line) axiomatizes all the inference steps inside a box in the previous figure. The first command of each set (that is SWITCHCONTEXT) forces GETFOL to concentrate its attention to a new context. GETFOL has in fact a notion of *current context*, which is, by definition, the context where the user is currently doing the reasoning. The operation of SWITCHCONTEXT corresponds to the sequence of leaving the current context and then entering the context whose name is the argument of the command.

RUP and RDW are two of the commands implementing the reflection rules. The command "LABEL LAB" labels the result of the following proof step with LAB. This allows to refer to this proof step simply by calling it LAB. In the arguments of the commands, a number n with an arrow above it refers to the the proof line which is n proof lines above the current one.

Finally, even if this cannot be seen in the printout, GETFOL has commands which allow the user to define arbitrary multicontext systems: it can create, copy and name contexts and define arbitrary bridge rules among them. For each context the user can also define the language, the set of axioms and its internal inference rules (the default is first order ND).