A novel approach towards an integration of multiple knowledge sources

Jacques Calmet *

Barbara Messing †

Joachim Schü *

Abstract

This work is to be considered as an extension of the integration mediator [20] approach to heterogenous databases. Former approaches towards an integration of different knowledge sources through a common interchange format, KIF [14] seem to be dead [10] because of the unability to incorporate future developments in knowledge representation. We would like therefore to perform the integration of *already existing* knowledge sources, such as databases (relational and object-oriented), knowledge-based systems, spreadsheets and statistical programs by the use of a deductive database based upon annotated logic [16, 18, 19] and constraint programming. The use of annotated logic allows a proper integration of inconsistent, temporal and uncertain knowledge, the use of multiple constraint domains enables the amalgamation of semantically different knowledge representations. Annotated logic furthermore enables the use of methods for performance improvement from classical logic such as magic sets since annotated logic resolution calculus is closely related to ordinary resolution calculus.

Keywords: integration of multiple knowledge sources, annotated logics, distributed AI

*Institut für Algorithmen und Kognitive Systeme, Universität Karlsruhe, Postfach 6980, 76128 Karlsruhe, Tel.: +49/721 608-4208, {calmet,schue}@ira.uka.de

†Institut für Angewandte Informatik und Formale Beschreibungsverfahren, Universität Karlsruhe, Tel.: +49/721 608-3924 bme@aifb.uni-karlsruhe.de

1 Introduction

To solve a problem in a distributed manner is either required or recommended in many applications for a variety of purposes. For instance performance can be improved, solutions can be accepted with more confidence and can be found although an agent may fail [5]. The question how to recognize and manage contradictory information provided by different databases, expert systems, humans or sensors which could belong to different independent agents, is one of the basic questions of Distributed AI. Coordination of conflicting information requires establishing priorities or strategies from a global point of view. The main problem of coordinating these multiple sources of knowledge which are independently developed and mostly *pre-existing* is to find a method for a proper integration of these different knowledge sources that is based upon a semantically well defined framework.

From a logical point of view, programs that contain a sentence and its negation are meaningless because anything can then be inferred. Usual deductive databases are restricted to definite clauses and therefore do not allow representation of $A \wedge \neg A$, but in the area of distributed and cooperating agents such contradictions are customary and conflicts concerning one issue should on one hand be represented and, on the other hand, must not destroy agreements in other issues.

In this paper we describe how both tolerating conflicts and clear semantics can be handled within logic programming. Our approach makes it possible to handle not only contradictory information but also simple temporal, uncertain and incomplete knowledge. We will therefore use extensions of classical deductive databases. One extension is based upon a special kind of multivalued logics, annotated logics, first introduced by V.S. Subrahmanian [16, 18, 19]. The advantage of annotated logics is that it makes possible to specify explicitly how to deal with contradictory information instead of computing maximal consistent subsets. Dealing with different structures of the set of truth values, we have a very flexible instrument to integrate divergent informations that may also be temporal, uncertain or vague.
Through the use of multiple domain constraints we are furthermore able to amalgamate schematic different knowledge representations and to define a suitable negation operator more efficient than the stable semantics of ordinary deductive databases. An already implemented multivalued hybrid knowledge representation system MANTRA [4] provides already some of the necessary capabilities. We are currently extending it by a component that uses annotated logics. Up to now, hybrid knowledge representation has only considered hybrid inference within frames, semantic nets, rules, and first-order logic, but our framework enables efficient reasoning within different knowledge sources.

Reasoning within such heterogeneous knowledge sources has been tackled by researchers in the areas of deductive databases and distributed artificial intelligence. There doest not exist a multi-agent testbed based upon multi-valued logics although it is recognized that such a framework is needed. Our work is a step in this direction. At this stage, the implementation is underway. This paper sketches some of the key ideas which make it possible.

It is structured as follows. We first give a very short introduction to database mediators and the theory of annotated logics. In section 3 we give an example of integrating different knowledge bases. Section 4 contains a large example from an insurance company and in section 5 we will discuss some strategies to combine divergent information from different knowledge sources.

2 Database mediators

Wiederhold [20] proposes a mediator approach to an integration of data from different heterogeneous databases: "A mediator is a program that helps integrating data or in some other way helps representing a higher level view to its applications". He identifies four different mediators : integration mediators, domain model mediators, monitor mediators and local mediators. Figure 1 illustrates the extended mediator architecture along the ideas of [9, 20]. There are different kinds of heterogeneity, semantic heterogeneity such as different names for the same entity and the use of different database systems on different hardware. Fahl [9] is currently working on an integration mediator supporting declarative queries based upon OSQL (object-oriented SQL), which is object-oriented and functional. Other researchers such as Krishnamurty [17] state that Horn clause logic is better suited to schema integration of relational databases than relational languages. Our approach is based on a logical framework as introduced by Subrahmanian [19], who suggested the use annotated logic and multi-domain constraints as a mean to the integration of schematic different knowledge sources. We could express temporal, inconsistent and uncertain knowledge in unified way through the use of annotated logic. Multi-domain constraints offer reasoning with different domains of knowledge stored in heterogenous databases in such an environment. On one hand we have semantical richer language than former languages to express different schemas from different knowledge sources in a common data model. On the other hand we could reason about already existing knowledge from different representation systems as relational and objectoriented databases which are not amenable by a transformation into a common data model.

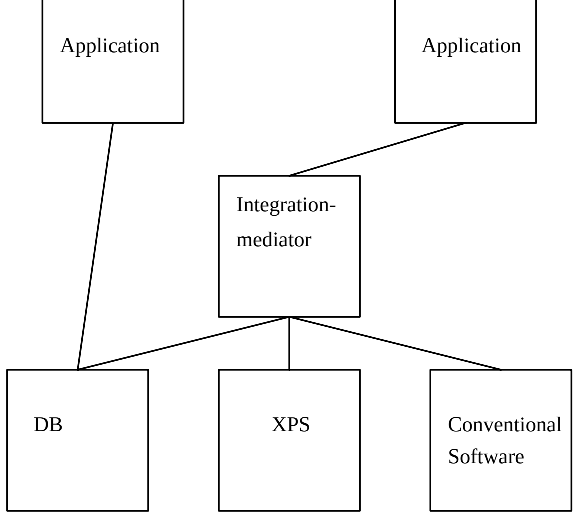

Figure 1: A mediator architecture

3 Introduction to annotated logics

We give here a very brief and informal introduction to annotated logics which is the theoretical background of our approach. For details the reader is referred to [16].

Definition 3.1 *A constrained annotated clause is :*

$$\begin{aligned} A:\mu \quad \leftarrow \quad & \textit{Multiple Domain Constraints} \,\| \\ & B_1:\mu_1 \wedge \ldots \wedge B_k:\mu_k \wedge \ \mathbf{not}(B_{k+1}:\mu_{k+1}) \ldots \wedge \ \mathbf{not}(B_{k+n}:\mu_{k+1}) \end{aligned}$$

The μ_i's are called annotations and are evaluable functions over an lattice[1] or simply elements of the lattice. For example the lattice $[0,1]$ could be used for reasoning with uncertain information in the same way as in possibilistic-logic [7] and fuzzy-logic. Other lattices, particularly useful for reasoning with inconsistent information or time, can be found in the literature [15].

The expressiveness and efficiency of such rules are enormous, since annotations could consist of several arguments from different lattices. Another feature is that bilattice-based logic programming is subsumed by annotated logic programming. This result is reported in [15]. This emphasizes the generality of annotated logic programming. The use of multi-domain constraints allows besides the features mentioned in 2 the definition of a negation operator. Negation will tracted by solving a constraint over the domain of possible ground substitutions.

Definition 3.2 *An annotated logic interpretation $\mathcal{I}$ is a map $\mathcal{I}$: ground atoms $\rightarrow \mathcal{T}$ from the herbrand base onto a lattice. An annotated atom $A:\mu$ is satisfied by $\mathcal{I}$ iff $\mathcal{I}(\mathcal{A}) \geq \mu$ with A is a* strictly ground instance. *A strictly ground instance is a ground instance as in classical first-order logic and furthermore all annotation variables and functions are evaluated to constants of the lattice.*

There is a fixpoint characterization of annotated logic programs according to the above definition of annotated logic interpretations. If there emerge more than one of the same ground atoms with different annotations during the resolution based inference of annotated logic programs, they are combined into one ground atom by taking the least upper bound of the annotations. For example $A: true, A: false$ are combined to $A:\top$ where $\top$ denotes a logical inconsistency, $A:0.7, A:0,8$ are combined to $A:0.8$. The body of a constrained annotated clause is satisfied if all literals are $true$ and furthermore all multi-domain constraints are satisfied. Some simple samples of annotated clauses are as follows :

[1]It is sufficient to think of a partial ordered set as a lattice to understand the following lines. Indeed every lattice is a partial ordered set but not vice versa.

- Simple temporal information: "On all days where US Dollar is low buy HongKong Dollars on the following day" could be expressed by following rule :

$$\text{Buy\_HK\_Dollar} : succ(T) \leftarrow \text{US\_Dollar\_low} : T$$

$succ()$ denotes a function which increases every member of the set T of moments (e.g. days).

- Suppose there are two different relational databases we would like to reason about. We would furthermore like to reason with uncertainty and temporal information restricted by numerical constraints. We are now using two-dimensional annotations and numerical constraints to show the full expressiveness of such rules. The two database relations are $Ordered\ Articles < Customer, Article, Amount >$, $Articles\_on\_stock < Article, Amount >$

  The conflict solution rule is: "If there are more articles ordered by a very important customer than currently available and present production is not very high then there is a chance of 75% to fulfil the order".

$fulfil\_Order(X) : [0.75, T] \leftarrow$
$Ordered\_Articles(X, Y, Z) \wedge\ Articles\_on\_Stock(Y, Z2) \wedge\ (Z2 < Z)\ \|$
$\mathbf{not}(production\_high : [0.5, previous(T)]) \wedge\ Important(Customer) : [0.9, previous(T)]$

Particularly the last example should have shown that our framework allows a proper integration of different knowledge sources based upon a clear model-theoretic semantic.

4 An example

Imagine an insurance company. To select the tariff for a person's disability insurance, there are many criteria which must be considered carefully.

To estimate the risk that disables someone to work the company needs a strategy to combine different facts such as age, profession, or diseases. We consider different knowledge bases that contain facts and some rules of risk analysis. These knowledge bases might result from investigations of different experts or staff members.

The annotations of the single knowledge bases differ in their domain, i.e. the lattice they stem from. This depends on the kind of knowledge we deal with. The supervisory knowledge base integrates these informations and evaluates them to assess the tariff for a person's disability insurance. We allow also that the supervisor's answer is "unknown", this should mean that some more information must be required to make a decision. We don't claim this example to be realistic but it will show how amalgamating knowledge bases works using annotated logics.

We denote the single knowledge bases by KB_1,...,KB_3, the supervisory knowledge base by KB_s. KB_1 contains information about age, sex and the marital status of a person. Because the age of a person depends on the current date, we use the lattice $\mathcal{T}_1 := I^+ \times [0, 1]$. I^+ stands for quarters of years. KB_1 might contain rules as:

$age\_factor\_risk(P) : (n, e^{-\frac{(x-30)^2}{100}}) \leftarrow age(P) : (n, X).$
$personal\_risk(P) : (n, min\{x_1 - 1 + \frac{X_2}{4} + \frac{X_3}{4}, 1\}) \leftarrow$
$\quad age\_factor\_risk(P) : (n, X_1) \wedge$
$\quad female(P) : (n, X_2) \wedge married(P) : (n, X_3).$

The first of these rules expresses that the age-factor risk of a person increases until he/she has reached the age of thirty and decreases in the following years (remark this is only an example!). The second rule matches the personal risk from the age-factor risk and the facts of being female (some people think that this increases the risk) and being married (diminishes the risk).

KB_2 includes the risk depending on a person's profession. Here we could find the following:

$professional\_risk(P) : high \leftarrow musician(P) : 1.$

$professional\_risk(P) : high \leftarrow pilot(P) : 1.$

$professional\_risk(P) : low \leftarrow computer\_scientist(P) : 1.$

$personal\_risk(P) : X^+ \leftarrow professional\_risk(P) : X \wedge smoker(P) : 1.$

$personal\_risk(P) : X^+ \leftarrow professional\_risk(P) : X \wedge climber(P) : 1.$

$personal\_risk(P) : X^- \leftarrow professional\_risk(P) : X \wedge vegetarian(P) : 1.$

The risk for musicians is high because of their precious fingers; and pilots live dangerous. Computer scientists have a low risk. Smoking (unhealthy) and climbing (dangerous) raise the risk while being vegetarian diminishes it (vegetarians take more care of their health).

The lattice $\mathcal{T}_2$ consists of the elements 0, $very\ low$, low,$unknown$, $high$, $very\ high$, and 1. The order is given in this sequence. X^+ denotes the successor in this sequence, where $1^+ = 1$. The analogous definition holds for X^-.

KB_3 analyzes the diseases a person has suffered from up to now. Each disease is signed with a risk factor, that is denoted by S. The longer the person is healthy again, the better for his/her tariff. The lattice here is $\mathcal{T}_3 := I^+ \times [0,1]$ and the knowledge base might include:

$personal\_risk\_(P) : (n, f(\mu, n, S)) \leftarrow disease(P) : (\mu, S)$, where

$$f(\mu, n, S) = \begin{cases} S \quad : & \mu = n \vee \mu - n = 1 \\ \frac{S}{ln(n-\mu)} & otherwise \end{cases}$$

The function f here expresses that the risk is high if a disease has occured in the current ($\mu = n$) or the period before ($\mu = n - 1$), and the risk factor of a disease diminishes in course of time ($otherwise$).

The supervisory knowledge base KB_s integrates the statements of the previous knowledge bases. The rules have the form $personal\_risk(P) : (\{s\}, ...) \leftarrow$ ... where in the body of the clause the statements of the single knowledge bases appear. Furthermore they are previously identified by indices denoting the knowledge base they come from, e.g.

$age\_factor\_risk(P) : (n, e^{-\frac{(x-30)^2}{100}}) \leftarrow age(P) : (n, X)$ is replaced by
$age\_factor\_risk(P) : (\{1\}, n, e^{-\frac{(x-30)^2}{100}}) \leftarrow age(P) : (\{1\}, n, X)$,

where 1 is the identifier. We choose $\mathcal{T}_s := I^+ \times$ [0,1]. The following two rules may for instance be in this knowledge base:

$personal\_risk(P) : (\{s\}, (n, max\{X_1, X_{32}\})) \leftarrow X_{32} > 0.4 \wedge$
$personal\_risk(P) : (\{1\}, (n, X_1)) \wedge$
$personal\_risk(P) : (\{2\}, high) \wedge personal\_risk(P) : (\{3\}, (X_{31}, X_{32}))$

$personal\_risk(P) : (\{s\}, (n, min\{X_1, X_{32}\})) \leftarrow personal\_risk(P) : (\{1\}, (n, X_1)) \wedge$
$personal\_risk(P) : (\{2\}, low) \wedge personal\_risk(P) : (\{3\}, (X_{31}, X_{32}))$

These rules "match" the risk-factors that arise from the different viewpoints (i.e. our different knowledge bases). So, if the risk factor of the disease is greater than 0.4 and the personal risk resulting from one's profession is high, the personal risk factor of the third knowledge base is taken, except if age factor is even higher (that is the content of the first rule). If the "professional" ris is low, the supervisory knowledge base evaluates the minimum of the age-factor and the "health"-risk (second rule).

Let "a" be a female, married and 26 years old person. Suppose she is a pilot, smokes, is vegetarian and suffered from a disease of factor 0.6 four years ago. The current time is denoted by n_0. Our knowledge bases KB_1,...,KB_3 yield

$personal\_risk(a) : (\{1\}, (n_0, 0.85))$
$personal\_risk(a) : (\{2\}, high)$, and
$personal\_risk(a) : (\{3\}, (n_0, 0.43))$.

In KB_s, the first rule fires, so we get $personal\_risk(P) : (\{s\}, (n, 0.85))$. The second rule also fires (notice that $high \geq low$ in $\mathcal{T}_2$), so we also get $personal\_risk(a) : (\{s\}, (n, 0.43))$. The fixpoint operator of annotated logics now takes the greater value, so the personal risk factor of a is 0.85.

5 Strategies on the supervisory level

On the supervisory level, strategies of managing divergent information may influence. The strategies depend on the domain of application. In the area of nonmonotonic reasoning, there are many approaches to handle incomplete and inconsistent knowledge [3], especially using defaults and preferring some theories to other ones. These problems also appear in inheritance networks. But they can't be solved solely in a logical framework (just because divergent opinions are not a logical problem).
There are no pat solutions to handle contradictions. Promising approaches have their origin in decision theory [8, 12]: They consider the behavior of groups and try to find strategies to make plausible decisions in the case opinions diverge. For example, if different agents have divergent goals, the utility of goals to the single agents are evaluated. In [8], metrics between different plans of agents are defined. In [2], the Dempster-Shafer Belief calculus is applied to the problem of divergent preferences in a group of agents: Different orders on a set are combined into one (this problem, in general, is NP-complete). Preference modeling by means of many-valued logics is already treated in [6]. Our approach using annotated logics is universal enough to incorporate different supervisory strategies. It is a focal point of our future research to provide mechanisms to use existing strategies.

6 Conclusions and directions for further work

This sketch of our work aimed to illustrate that the use of annotated logic offers a promising approach to problems involving both distributed and cooperative knowledge. This is its theoretical motivation. From a practical point of view, the first step consists of selecting the relevant modules of MANTRA to transform it into a shell for distributed systems which runs on workstations. There is a long list of theoretical problems which arise from such a study, distributed possiblistic truth reason maintenance [7, 13], knowledge acquisition for distributed systems, extending the KADS framework, just to name some of them. However in relation to the short term availability of our system and its relevance for individual application the current research directions are query optimization within such heterogeneous knowledge sources and efficient caching of previously computed queries using OLDT[2] technique [1]. Our work will then be applied in a marketing company for an integration of different large pre-existing relational databases with rule-based systems.

References

[1] S. Adali, V.S. Subrahmanian. Amalgamating Knowledge Bases II: Algorithms, Data Structures, Query Processing. *Preprint University of Maryland, College Park, 1993*

[2] J.A. Barnett. Combining opinions about the order of rule execution. *Proc. AAAI(1991),pp.477-481*

[3] G.Brewka. Nonmonotonic Reasoning: Logical Foundations of Commonsense. *Cambridge Tracts of Computer Science 12.*

[4] J. Calmet, I.A. Tjandra, G. Bittencourt, MANTRA: A Shell for Hybrid Knowledge Representation *Proceedings of the Third IEEE International Conference on Tools for Artificial Intelligence, San Jose, 1991, pp.164-171*

[5] E. Durfee, V.Lesser, D.Corkill. Cooperation through communication in a distributed problem solving method. *In: L.Gasse, M.Huhns: Distributed Artificial Intelligence. Pitman, London, 1987, pp.29-58.*

[6] P.Doherty, D.Driankov, A.Tsoukias. Partiality, Para-consistency and Preference Modeling: Preliminary Version. *Research Report, University of Linköping, August 1992.*

[2] The acronym means Ordered selection strategy with Linear resolution for Definite clauses with Tabling

[7] D.Dubois, J.Lang, H.Prade. Possibilistic Logic and Partially Inconsistent Knowledge Bases. *Research Report, IRIT-90-2, February 1990, Toulouse*

[8] E.Ephrati, J. Rosenschein. Constraint intelligence action: Planning under the influence of a master agent. *Proc. AAAI (1992), pp.263-268.*

[9] G.Fahl. Integration of Heterogeneous Databases in a Mediator Architecture. *Research Report, CAELAB, Department of Computer and Information Science, Linköping University, September 1992*

[10] M.L. Ginsberg. Knowledge Interchange Format: The KIF of Death. *AI Magazine, Vol.12, No.3, 1991, pp.57-63*

[11] M.L. Ginsberg. Multi-valued logics *Proceedings AAAI-86, pp. 243-247*

[12] P.Gmytrasiewicz, E.Durfee, D.Wehe. A Decision-theoretic Approach to Coordination Multiagent Interaction. *Proceedings IJCAI (1991), Vol.1, pp. 62-68*

[13] M. N. Huhns and David Bridgeland. Multiagent Truth Maintenance. *IEEE Transactions on Systems, Man, and Cybernetics, 1991, Vol.21, No.6, pp.1437-1445*

[14] M.R. Genesereth. A KIF proposal . *Technical report, Stanford University, 1990*

[15] M. Kifer, V.S. Subrahmanian. Theory of Generalized Annotated Logic Programming and its Applications *Journal of Logic Programming 1992, No.12, pp. 335-367*

[16] M. Kifer, V.S. Subrahmanian. Theory of Generalized Annotated Logic Programming and its Applications *Journal of Logic Programming 1992, No.12, pp. 335-367*

[17] R. Krishnamurty, W. Litwin, W. Kent. *Language Features for Interoperability of Databases with Schematic Discrepancies, Proc. ACM SIGMOD 1991*

[18] V.S. Subrahmanian. Amalgamating Knowledge Bases. *Preprint 1993, To appear in: ACM Transactions on Database Systems*

[19] J. Lu, A. Nerode, J. Remmel, V.S. Subrahmanian. Towards a Theory of Hybrid Knowledge Bases *Preprint 1993*

[20] G. Wiederhold. Building Adaptive Applications using Active Mediators. *Proc. Database and Expert System Applications, DEXA 91, Springer-Verlag, 1991, pp. 508-513*