Legal Event Reasoning for Software Agents[1]

Alexander Yip & Jim Cunningham
{ay1,rjc}@doc.ic.ac.uk
COMMA Group, Department of Computing
Imperial College London, SW7 2AZ

Abstract

In a previous paper (Yip & Cunningham 03) the authors extended a top-level reference ontology to support the representation and reasoning of legal ownership concepts in multi agent communication. In this paper we further enhance this ontology. However, for a software agent to reason about such concepts, it is also necessary to have a computational model of the ontological system.

One way to model such system is to treat the communication as process and actions. This paper presents a partial implementation of an agent that supports such temporal events (process and actions) for reasoning about legal concepts.

1 Introduction

There is a tradition of artificial intelligence researchers working in the area of legal discourse. Sergot et al.'s illustration of computational rules for reasoning about the British Nationalisation Act (Sergot et al. 1986) and McCarty's Language for Legal Discourse (McCarty 1989) have contributed to the logical representation of legal problems and suggested ways to reason about them. Our work is in this tradition but with more emphasis on the world of software agents, where there is need for an agent to interact with a dynamic environment which includes other agents. Hence we seek a simplistic, but computationally effective basis for legally supportable behaviour by software agents, using niave models of human rationality rather than models based on argumentation (Lodder & Herczog 95; Prakken&Sartor 96,98). The latter seem more appropriate for modelling sophisticated dialogue in negotiation and legal argument itself.

Thus we look at reasoning from a naïve perspective, emphasising the need for common language and definitions in agent interaction. The present paper elaborates and extends the metaphysical framework proposed at the LEA'03 workshop to provide for ontologies in legal domain. An interval-based reification of occurrences now allows the definition of durative transactions between agents. A modest generalisation of a multi-threaded computational logic provides a basis for implementing such legal processes, but raising problematic issues in state approximation, communication and participant roles.

2 A Reference Ontology

2.1 Definition of ownership

In our work, we take the common definition of ownership from English and American legal systems. In such systems, "*ownership is a relationship between a legal person (individual, group, corporation, or government) and an object. The object of concern may be corporeal, such as furniture, or completely the creature of law, such as a patent, copyright, or annuity. The most basic meaning of ownership is that one's government or society will help to exclude others from the use or enjoyment of one's possession without one's consent, which may be withheld except at a price.*" (EB)

[1] In "The Law of Electronics Agents", Proceedings of the LEA 04 Workshop, CIRSFID, University of Bologna, June 2004, pp.35-53

Although ownership can be treated as a single legal object, sometimes it is necessary to see it as a bundle of rights. (McCarty 2002) For example, copyright (ownership of a artistic creation) is a collective term for a bundle of rights which usually includes the "right of publication", the "right of exploitation" and the "right to remuneration" in respect of a piece of work. So a person who owns the copyright (has the right to possession of the copyright) of a piece of work, can license the right of publication (a component of the ownership) to a third person (a publisher).

As ownership is a creature of law, and in turn the creation of law is heavily influenced by the local social system, it is possible that different cultures will have different interpretations of ownership. Because of this divergence, the ownership ontology presented here seeks to address the issue at the philosophical level rather than at the fine detail level of each individual legal system. In our previous work we have treated rights as an integral part of ownership. In the revised ontology a right has been given a more expressive presence, but the main focus of the paper remains the dynamic expression of interaction about ownership.

2.2 *Ownership ontology*

Figure 1 and table 1 are revisions of a top–level reference ontology and a supplementary concrete legal ontology proposed in our LEA03 paper. The reference ontology was adapted from work of Schneider on a naïve metaphysics for everyday reasoning (Schneider 2001). It is an informed, but in term of modern physics, naïve conception of the human world as substances (entities and objects etc.), occurrences (or temporal eventualities) and the roles by which the instances of substances and occurrences relate. Our previous adaptation was simply to make a place for legal entities and legal occurrences (states and events), each having more concrete instantiations in Table 1. Introducing legal types and roles already gives special status to collective mental concepts which could be considered as embraced by sociological categories. Since we do need a place for communicative events, figure 2 is a possible revision of the types in figure 1 which embraces the social. This, however, is already politically contentious, since it would probably be unacceptable in a theocracy.

As before, our main concern for legal reasoning by agents, are the occurrences of three effectively independent domains – Environmental, Mental and Legal. The most observable are perceived as physical, part of what we have called environmental domain. The mental domain represents the internal states and events of an agent, be it a human or a software agent. To give an example in terms of a BDI software agent (Rao 91,95), the environmental states and events represent the environment in which the agent operates while the mental states are its beliefs, desires and intention. A mental event can be an update to the agent's belief or a deliberative act to choose a new intention through which to achieve an adopted desire.

The third domain, the legal domain, envelops legal states of affairs and legal events. The domains are effectively independent because each domain consists of a set of atomic terms for states and events, mutually disjoint from those of the other domains. However, the domains have a common ontology of "occurrences" (states and events), substances, and roles, as illustrated in the taxonomical structure of figure 1. Thus an inanimate object is a substance and may have both a physical state e.g. hot or cold and a legal state e.g. being owned. We consider a software agent to be an animate object. A decision it makes is a mental event. In practice an event is usually a process rather than just a change of state, so we introduce an interval based temporal model. An agent needs to consider the states and events of other domains. When an agent, software or human, makes a decision, it needs to take into account the environmental factors (environmental states) and the mental states of other participants. To obtain inanimate resources, it should take into account the legal states of the resources – for example, a book may be owned by someone else.

- *Kinds*
  - *Types*
    - $\lambda x\ Substances(x)$
      - $\lambda x\ Animate(x)$
      - $\lambda x\ Inanimate(x)$
        - $\lambda x\ Physical(x)$
        - $\lambda x\ Legal(x)$
    - $\lambda o\ Occurrence(o)$
      - $\lambda o\ State(o)$
      - $\lambda o\ Event(o)$
        - $\lambda o\ Mental(o)$
        - $\lambda o\ Environmental(o)$
        - $\lambda o\ Legal(o)$
  - *Roles*
    - *Basic Roles*
      - *Complement Roles*
        - $\lambda x\ Origin(o,x)$
        - $\lambda x\ Theme(o,x)$
        - $\lambda x\ Addressee(o,x)$
        - $\lambda x\ Benefactive(o,x)$
      - *Supplement Roles*
        - $\lambda x\ Instrumental(o,x)$
        - $\lambda x\ Manner(o,x)$
        - $\lambda x\ Locative(o,x)$
        - $\lambda x\ Motion(o,x)$
        - $\lambda x\ Temporal(o,x)$
        - $\lambda x\ Purpose(o,x)$
        - $\lambda x\ Reason(o,x)$
        - $\lambda o\ Cause(p,o)$
        - $\lambda o\ Consequence(p,o)$
        - $\lambda x\ Concession(o,x)$
      - *Legal Roles*
    - *Derived Roles*
      - $\lambda x\ Agentive(o,x)$
        - $\lambda x\ Performer(o,x)$
        - $\lambda x\ Experiencer(o,x)$
      - $\lambda x\ Cause(o,x)$
      - $\lambda x\ Legal(o,x)$

Figure 1 - A top-level reference ontology (revised)

- *Kinds*
  - *Types*
    - $\lambda x\ Substances(x)$
      - $\lambda x\ Animate(x)$
      - $\lambda x\ Inanimate(x)$
        - $\lambda x\ Physical(x)$
        - $\lambda x\ Social(x)$
          - $\lambda x\ Legal(x)$
    - $\lambda o\ Occurrence(o)$
      - $\lambda o\ State(o)$
      - $\lambda o\ Event(o)$
        - $\lambda o\ Mental(o)$
        - $\lambda o\ Environmental(o)$
        - $\lambda o Social$
          - $\lambda o\ Communicative(o)$
            - $\lambda o\ Declarative(o)$
            - $\lambda o\ Imperative(o)$
            - $\lambda o\ Interrogative(o)$
            - $\lambda o\ Performative(o)$
            - $\lambda o\ Permissive(o)$
            - $\lambda o\ Commissive(o)$
          - $\lambda o\ Legal(o)$
  - ...

Figure 2 - A extension to include social and communicative occurrences

2.3 A concrete ownership ontology

To aid reasoning in ownership related situations, it is helpful to pre-define certain concrete ontological terms. These are presented in table 1. It is possible to "hang" this ownership ontology as leaves on the top level ontology. Examples are ownership, a legal object, which is an instantiation of legal inanimate; while a buys event is an instantiation of the legal event.

Lexical Element	Semantic Category
$\lambda x.person(x)$	Physical animate
$\lambda x.legalperson(x)$	Legal animate
$\lambda x.jurisdiction(x)$	Legal inanimate
$\lambda x.object(x)$	(Legal or physical) inanimate
$\lambda x.agent(x)$	Physical animate
$\lambda x.money(x)$	Physical inanimate
$\lambda y.ownership(x,y)$	Legal inanimate
$\lambda y.right(o,y)$	Legal inanimate
$\lambda y.obligationt(o,y)$	Legal inanimate
$\lambda o.possesses(o,x,y)$	Physical state
$\lambda o.owns(o,x,y)$	Legal state
$\lambda o.transfers(o,x,y,z)$	Legal event
$\lambda o.buys(o,x,y,z)$	Legal event
$\lambda o.sells(o,x,y,z)$	Legal event
$\lambda o.borrows(o,x,y,z)$	Legal event
$\lambda o.hires(o,x,y,z)$	Legal event
$\lambda o.pays(o,x,y,z)$	Legal event
$\lambda x.authority(o,x)$	Legal role
$\lambda x.represents(o,x,y)$	Legal role
$\lambda x.consideration(o,x)$	Legal role
$\lambda x.delegates(o,x,y)$	Legal role
$\lambda x.possessor(o,x,y)$	Derived legal role
$\lambda x.owner(o,x,y)$	Derived legal role
$\lambda x.seller(\mathrm{o,x,y})$	Derived legal role
$\lambda x.delegate(\mathrm{o},x,y)$	Derived legal role
$\lambda x.mandator(o,x,y)$	Derived legal role

Table 1 – Concrete ownership ontology

3 Roles, Rights and Reasoning

Our development is chiefly been concerned with legal events and legal states themselves, and mechanisms whereby agents distinguish the legal from the physical. But the more intricate and valuable aspects of reasoning model based on the metaphysical categorization of the world lie in the participation of objects, and in particular animate objects in roles associated with states and events. Thus the *owner* of a car, in possession of a licence to drive, can enjoy ownership in this way.

Legal reasoning is sophisticated, using abstract but generic terms like rights and obligations for ephemeral legal entities that can be bought, sold and possessed as substances, not necessarily individually, but for the purpose of giving more detail to otherwise variable meanings to more embracing bundles of legal entities like ownership. We will consider legal rights and obligations as exercised through roles in events and states.

McCarty (McCarty 1989), has proposed a Language for Legal Discourse (LLD) and later used the language to present a case on reasoning about ownership in (McCarty 2002). The 2002 paper presents a detailed and insightful technical analysis of the legal concept ownership under Anglo-American legal systems. In the paper, the language LLD is presented as a tool for formalising legal statements, for example contracts. Although LLD has the concept of objects and events, (as well as tame and obligations) it is a language for *describing* events and situations, executable more in the sense of simulation, than by design as a programming language.

We tackle similar problems from a different angle. As software agents mainly communicate using message passing, we employ natural language analysis to reason about communication in term of individuals (objects/events) and thematic roles. We provide a reference ontology which defines a foothold for legal concepts to interact with other general concepts like location (origin, destination), roles (performer, experiencer), etc. To complement the ontology, we implement an interval logic reasoner as part of a concurrent procedural logic to reason about communication and hence legal event in term of intervals.

3.1 Legal Events

Although occurrences like buying and selling have legal significance, their temporal effect is similar to other everyday events which transfer properties between substances. In buying and selling it is the ownership of a legal object which is transferred, but for some consideration. The temporal effect can be logically axiomatised, and explicitly defined with the help of interval operators, for example, using *beforehand*, *afterwards* and *during* to intuitively signify underlying temporal relationships which here we do not define further[2]. There are then many ways of providing formal definitions for mechanised interpretation, just as there are many ways to draft legislation, all can be open to criticism. For simplicity, suppose:

$$buys(x,y,z) \leftrightarrow beforehand\ \underline{owns}(z,y) \wedge afterwards\ \underline{owns}(x,y) \wedge during\ pays(x,a,z)^{3}$$

This is illustrated in Figure 3 (see footnotes), but we may treat it as a definitional ideal. We underline owns to indicate it as a state (see section 4.2). Obviously such an axiom depends on the definition of *pays*, while a *sells* event has a complementary definition and a more general *transfers* event can be defined. Because the interval of occurrence is implicit in this style of logic (but can be made accessible), auxiliary properties of the occurrences can be indicated by adding simple conjunctions, for instance a property like within *jurisdiction*(*j*). This feature of "event reasoning" helps with scalability

[2] Binary relations between intervals on a real line were proposed by Allen (1984) for temporal reasoning. These were embedded in a modal logic for qualitative reasoning by Halpern and Shoham (1986), a fragment of which was used in Leith and Cunningham (2001) for reasoning about tense and aspect.

[3] To avoid clutter, quantification is left implicit. Here *x,y,z* are free variables which can be universally quantified. The operators *beforehand*, *afterwards* and *during* can be regarded as possibility modalities, as illustrated in figure 2. The stative nature of the predicate *owns* can be expressed using the *during* operator, in essence by requiring that the predicate is also true on all sub-intervals during an interval on which it holds. Here such homogeneity is signified by underlining.

of the formal system. (The use of an explicit definition also has certain advantages when dealing with what is usually called the frame problem of qualitative reasoning). We are not seeking here to demonstrate an adequate formalisation, merely remind ourselves that at least the simpler characteristics of certain classes of everyday legal durative and commercial activity can be brought within the framework of logical definitions.

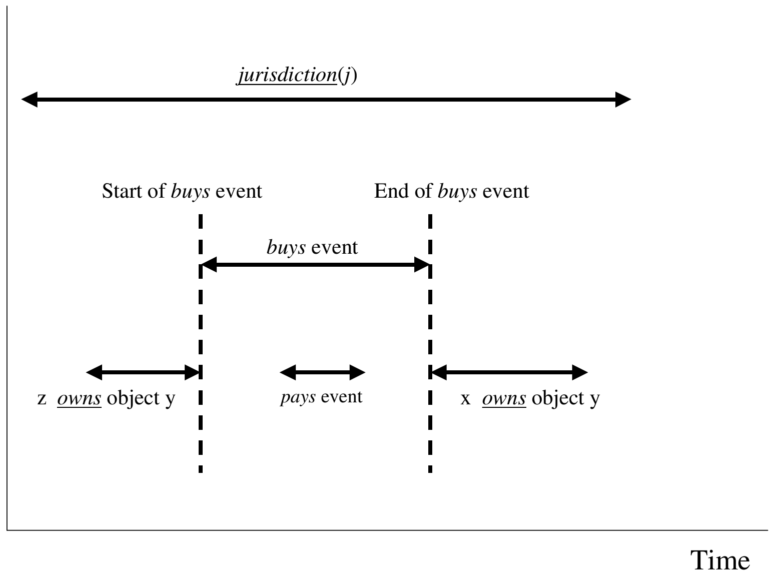

Figure 3 - Temporal interval interpretation of an ideal ***buys*** *eve*nt

The above definition does not specify ownership during the buying event. It could be considered inappropriate for some types of sales, such as the purchase of a new house to be constructed on land owned by the vendor. Figure 4 illustrates some alternatives; only the third case (Alternative 3) being incompatible with the above definition. Each case is readily expressed with the form of interval logic we suggest.

The subtle but important differences between buying, borrowing, hiring *etc*. can readily be expressed, albeit naively, by exposing the further details of the ownership using the same notation:

X borrows y from Z, we assume at start Z has the whole ownership of y. During the borrowing period, Z transfer the ownership of the "right of possession" of y to X, but after the period the whole ownership is reverted back to Z.

borrows(x,y,z) ↔ *beforehand* *<u>owns</u>(z,y)* ∧ *during* (*<u>owns</u>(x,r)* ∧ *<u>possesses</u>(r,y)*) ∧ *afterwards* *<u>owns</u>(z,y)*

Similarly, hiring is a borrowing arrangement with a consideration:

hires(x,y,z) ↔ *beforehand* *<u>owns</u>(z,y)* ∧ *during* (*owns(x,r)* ∧ *<u>possesses</u>(r,y)* ∧ *pays(x,a,z)*) ∧ *afterwards* *<u>owns</u>(z,y)*

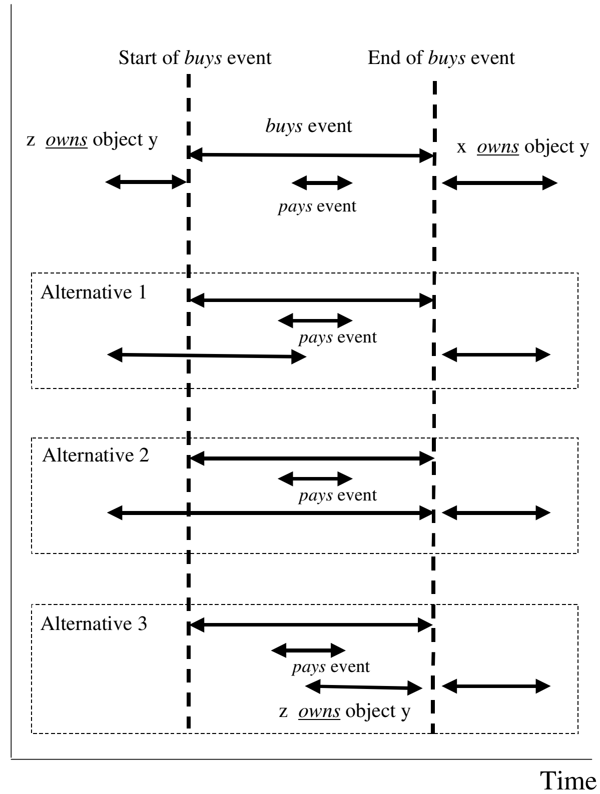

Figure 4 - Temporal alternatives to "beforehand owns"

4 Computational Realisation

4.1 Background

Several special logic based formalisms and meta-level programming systems have been developed as research tools to facilitate the realisation of planned but revisable behaviour by so called intelligent systems, including cognitive robots, in a temporal world. These include the situation and event calculi (McCarty & Hayes 69), belief-desire-intent logics (Rao 91, 95), and some executable temporal and modal logic systems (Fisher 1994). The issue is not just how to express the rationale in a logical way, or how to program such systems in maintainable ways, so that they are in keeping with the spirit of executing high level, declarative specifications; but that at the same time the program must interact with real world events. It seems that without an interval interpretation we cannot express the difference between a homogeneous condition, like the stative predicate owns, and a process or event predicate which might test it. This is also part of the problem: we cannot *execute* a stative condition; it would incur an unbounded number of tests. Instead, a human or a computational agent can only execute a finite number of tests, perhaps just one. So our ideal world for judicial transfer of ownership does not exist but the approximation we need has become more explicit.

We take it as evident that a contemporary programming system must support concurrent and distributed processing in some form, and be able to communicate with other processing agents. But logic programming systems have already provided this in various ways. As a case for comparison, the QuProlog system (QuProlog) provides multi-threading, inter-thread co-ordination using controlled access to a shared store, and asynchronous message-passing, in which case communication can be with any other processing agent in the world that has the support an Internet Communication Module in the style of McCabe and Dale (1998). This ICM allows FIPA speech-act level agent communication primitives and uses a Unix-like environment, but these days Linux can run on a chip. So the burden of responding to external and internal real time events, in a timely manner is alleviated.

In order to provide an efficiently executable logical specification of real-world event processing by each multi-threaded agent in a world of agents we propose to prefer a semantic interpretation where each executed literal is true on some closed temporal interval. The usual local left-to-right execution of conjunctive sub-goals, that crucial compromise for efficient execution of a logical specification is not only accepted, as in the real time logic programming variant Golog (Hector et al. 1997), but is part of the temporal interpretation of the logic. This way each local (logic) program thread is constrained to behave more like a conventional sequential program, but communication with concurrent processing threads and processing agents provides expressive compensation. Furthermore, where necessary, the interval relations which were expressed in the preceding definitions using model style operators, can instead, via pre-processing, or with obvious direct coding, be embedded as constraints on an auxiliary pairs of rational time parameters to represent the start and end of the interval of interpretation for each predicate instance. Each realisation is temporal approximation based on processing events, including messages which realise the unspecified communication events of the earlier definitions.

4.2 Reasoning in Intervals

In our reference ontology there are two main types of temporal entities – events and states. We find that using Allen's interval model we can readily express the difference between events and states.

A state is a condition that is homogeneously true throughout a temporal interval. An event is temporal incident that happens on an interval (or degenerately, a point). Allen defines thirteen different types of binary relations between intervals, but only six are independent.

In Prolog there is no built in support for interval based reasoning. It is necessary to implement such function to facilitate interval modelling. The implemented reasoner implements all thirteen relationships from the Allen's logic, e.g. Before, After, Meets, Met_By, Overlaps, Overlapped_By, During, Includes, Starts, Started_By, Finishes, Finished_By and Equal, in predicate form. A query likes "Given a time line T, is the interval (a,b) before interval (c,d)?" can be readily answered by the following Prolog query:

isBefore((a,b),(c,d),T)

This provides the foundation for us to reason about the interaction between the different events and states, for example, the interval (a,b) may indicate the period where the buyer is making a payment to the seller, and the interval (c,d) may indicate the process where the seller hand over the priceless antique!

4.3 Buying a ticket

A realisation of one (legal) agent, B, buying a ticket from an agent S, can be expressed in a programming language like QuProlog as the execution of threads which are complementary *approximations* to definitions for *buys* and *sells* threads. We then find that the first and second alternatives of Figure 4 allow simpler approximations.

We assume that agent B has already established an intention to buy a ticket, perhaps in an internal reasoning thread, and has already located the seller via other reasoning and discovery mechanisms. We mention related ontological aspects of these roles in a final remark. It is not essential to include interval parameters in this case but their inclusion shows the relative timing and indicates a way to compile the earlier, more abstract definitions, prior to seeking approximations. Successful QuProlog buy and sell threads are essentially :

```
buys(ticket, agentS, money, T0,T10) :-
        test(owns(agentS,ticket),T1,T2),
        fipa_request(ticket,money,T3,T4) ->> agentS,
        fipa_agree(T5,T6) <<= agentS,
        fipa_inform(money, T7,T8) ->> agentS,
        send_ticket(X,T9,T10) <<= agentS,
        T0<=T1, T1<=T2, T2<=T3, T5<=T6, T6<=T7, T9<=T10.
```

```
sells(ticket, X, money, T0, T10) :-
        fipa_request(ticket,money,T3,T4) <<= X,
        fipa_agree(T5,T6) ->> X,
        fipa_inform(money,T7,T8) <<= X,
        send_ticket(X, T9,T10) ->> X,
        T3<=T4, T4<=T5, T7<=T8, T8<=T9.
```

The code fragments are written in QuProlog but some operational details have been omitted to preserve clarity. For example, the *test* function encapsulates an event which checks the state of a stative predicate e.g. *owns*. It is possible that such predicate requires further elaboration to involve interactions with external entities. In QuProlog, each thread has a separate message queue. To send a message, the sender thread executes a clause with the pattern "*Message ->> Receiver*" and the content of *Message* will be stored in the receiver's message queue. When the receiver is ready to process the message, it can use the pattern "*Message <<= Sender*" to match a message stored in the message queue. If the queue is empty, the thread will suspend until a message arrives.

We use *fipa_* as a prefix to indicate a FIPA communicative act, or a communicative event. Each predicate represents an occurrence of an event on the interval indicated by its pair of temporal parameters T_{start}, T_{end}. A state is represented by the existence of the stative predicate in the data base and it may be tested by a query event. By using the intervals as constraints, it is possible to visualise the interaction as shown in Figure 5.

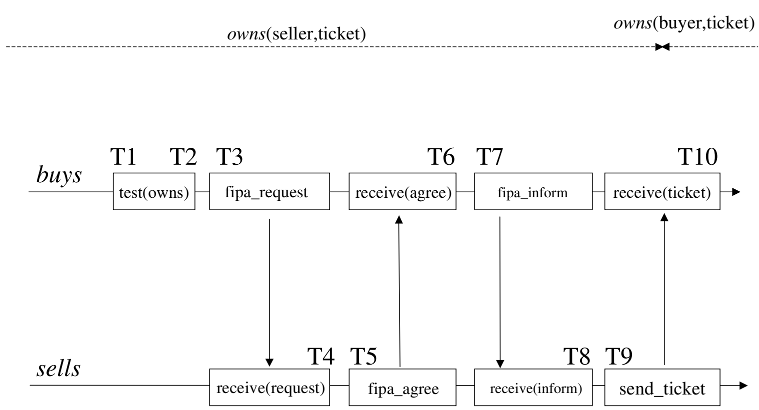

Figure 5 - Execution Intervals

Communication is not instantaneous in either the software or the physical world. It takes time for a message to be delivered to the receiver after the sender has sent it. But a waiting thread, upon receiving a message, is notified that a message event has occurred and can access the content. The interval parameters record when it occurred. Even though each communication is asynchronous, each is a single communicative event; so it appears as true on the same interval in each thread.

Where a software agent is not a legal person, it does not have the legal capacity to enter into a contract or to represent its owner as a legal entity. All it can be is a mechanism for the owner, so it *buys_for* and *sells_for*, and *pays_for* the owner. We have discussed some of these ownership issues in the past (Yip & Cunningham 2002). In this case, all transactions of an agent musty appear as extensions of the owner's legal capacity, where the agent has an instrumental role rather than a legal role. These seem to require that a legal transaction likes *buys*, be re-expressed in terms of an instrumental buys-for, where the transaction is recorded and either on trust, or conditional until certification by the legal owner.

Trust and certification are not concepts we have included in the present ontology, so even though a mechanistic interpretation of an agent's transactions may not be very different from current electronic commerce, we leave the formal definition for the future.

As a final remark, in the context of a top level ontology for reasoning about knowledge using thematic semantics, such as that suggested in Schneider, it is helpful to regard the thematic roles of arguments as sortal constraints so that the distinct perspectives of the same event in the interactions of multiple agents can be expressed. Global coherence requires that roles like buyer and seller (payer & payee, lender & borrower and auctioneer & bidder etc.) be recognised as derived conditions of the same complex transaction and be consistent with roles in the detailed events which comprise a transaction.

5 Conclusions

In order that computational agents be equipped to reason about legal events, we have revisited the ontological foundations of agent reasoning about the world and presented a top level ontology which embraces the legal domain.

This has enabled us to distinguish legal states from legal events and to illustrate some temporal features that we believe to be important in definitions of durative legal transactions. There appears to be a fundamental problem of approximation in using a test to assert a legal state, or for that matter any external state. Moreover, although our interval-based representation of a durative process appears to have a natural reification in a contemporary multi-threaded logic programming language, this itself raises new issues regarding the treatment of durative communicative events and the visible safeguarding of roles in a transaction.

Thus the authors feel that this paper sheds some light on fundamental and still open issues in the application of software agents in areas like the law and hope it can help the development of solutions.

6 Acknowledgments

The authors wish to thank Luc Schneider for his prior art and for the sharing of his insight during our discussions. We also want to thank Adrian Williams and Lloyd Kamara for their helpful suggestions on an earlier draft.

7 Bibliography

[Agentcities 2002] EU Agentcities.RTD project IST-2000-28385 http://www.agentcities.net

[Allen 1984] J.Allen. *Towards a general theory of action and time*. Artificial Intelligence,23(2):123-154. 1984.

[Barbuceanu] Barbuceanu, M., *An architecture for agents with obligations*, http://citeseer.nj.nec.com/57306.html

[EB] "Ownership", *The New Encyclopaedia Britannica (15th Ed.)* Encyclopaedia Britannica, Inc., Chicago

[FIPA 2002] FIPA Communicative Act Library Specification http://www.fipa.org/specs/fipa00037/

[Fisher 1994] M.Fisher. *A Survey of Concurrent METATEM – The Language and its Applications*. Temporal Logic – Proceedings of the First International Conference (LNAI vol 827), page 480-505. 1994

[Gelati et al. 2002] J.Gelati, G.Sartor, A.J.I.Jones, and M.Sergot. *Normative relations between agents and humans: Representation, mandate and ownership*. EU IST-1999-10298 ALFEBIITE project Deliverable. 2002.

[Halpern & Shoham 86] J.Halpern, and Y.Shoham. *A propositional modal logic of time intervals*. In Proceedings of Symposium on Logic in computer Science, pages 279-292. Cambridge, Massachusetts. June 16-18 1986.

[Hector et al. 1997] J.Hector, L.Reiter, R.Reiter, Y.Lesperance, F.Z.Lin, and R.B.Scherl. *Golog: A Logic Programming Language for Dynamic Domains*. Journal of Logic Programming 31(1-3), pages 59-83. 1997.

[Jones & Sergot 1996] A.J.I.Jones, and M.J.Sergot. *A Formal Characterisation of Institutionalised Power*. Journal of the IGPL 4(3):429—445. June 1996.

[Leith & Cunningham 01] M.F.Leith, and J.Cunningham. *Aspect and Interval Tense Logic*. Linguistics and Philosophy 24(3), pages 331-381. 2001

[McCabe & Dale 1998] F.G.McCabe, and J.Dale. *Asynchronous Messaging*. Fujitsu Laboratories of America Technical Memorandum FLA-NARTM98-04, Fujitsu Laboratories of America, April 1998

[McCarty & Hayes 69] J.McCarty, and P.Hayes. *Some Philosophical Problems from the Standpoint of Artificial Intelligence*. In B.Meltzer and D.Michie (ed.) Machine Intelligence 4, pages 463-502. Edinburgh University Press. 1969.

[McCarty 1989] L.T.McCarty. *A language for legal discourse I. basic features*. In Proceedings of the Second International Conference on Articial Intelligence and Law, pages 180-189, New York, ACM Press. 1989.

[McCarty & Costello 98] J.McCarty, and T.Costello. *Combining Narratives*, In Proceedings of the sixth International Conference on Principles of Knowledge Representation and Reasoning 98, pages 48-59. 1998.

[McCarty 2002] L.T.McCarty. *Ownership: A case study in the representation of legal concepts*. Artificial Intelligence and Law 10(1-3), pages 135-161. 2002.

[Parsons 1990] T.Parsons. *Events in the Semantics of English: A Study in Sub-atomic Semantics*. MIT Press, Cambridge/MA. 1990.

[QuProlog] www.itee.uq.edu.au/~pjr/HomePages/QuPrologHome.html

[Rao 1991] A.S.Rao, and M.P.George. *Modeling rational agents within a BDI-architecture,* Proc. KR'91, pages 473-484. Morgan Kaufmann, 1991

[Rao 1995] A.S.Rao, and M.Georgeff. *BDI Agents from Theory to Practice*, Technical Note 56, AAII, April 1995

[Schneider 2001] L.N.Schneider. *Naive Metaphysics: Merging Strawson's Theory of Individuals with Parson's Theory of Thematic Roles as a Basis for Multiagent Semantics*. MSc Thesis. Imperial College London, Department of Computing. 2001.

[Sergot et al. 1986] M.J.Sergot, F.Sadri, R.A.Kowalski, F.Kriwaczek, P.Hammond, and H.T.Cory. *The British Nationality Act as a logic program*. Communications of the ACM, 29(5):370-386, 1986.

[Shanahan 1997] M.Shanahan. *Event Calculus Planning Revisited*. Lecture Notes in Computer Science 1600, Pages 409. 1999.

[Shanahan 1999] M.Shanahan. *The Event Calculus Explained*. ECP. Pages 390-402. 1997.

[Strawson 1959] P.Strawson. *Individuals: An Essay in Descriptive Metaphysics*. Routledge, London. 1959.

[Thematic roles] J.Sowa. Notes related to "Knowledge Representation: Logical, Philosophical, and Computational Foundations", Brooks Cole Publishing Co., ISBN 0-534-94965-7, 1999. http://www.jfsowa.com/ontology/thematic.htm

[Yip & Cunningham 02] A.Yip, and J.Cunningham. *Some Issues on Agent Ownership.* In "The Law of Electronic Agents: Selected Revised Papers", Proceedings of LEA Workshop, CIRSFID, University of Bologna, July 2002, pp. 13-22. 2002.

[Yip & Cunningham 03] A.Yip, and J.Cunningham. *Ontological Ontological Issues in Agent Ownership (Reasoning About Agent Ownership).* In Proceedings of the second Law and Electronic Agents Workshop, pages 113-126. Norwegian Research Centre for Computers and Law. 2003.