Introduction: Standards for Knowledge Representation

John Kingston and Austin Tate

A number of articles in this issue of *airing* examine the development and use of standards for knowledge representation. This article outlines the motivation behind AIAI's interest and involvement in the development of these standards.

AIAI has always been strong in the field of knowledge-based planning and scheduling, which has led to the development of rich plan representations which allow for the specification of tasks, actions and events. The main aim of these representations was to provide a framework for integrating different tools and systems. An early example of this was the PLANIT system [2], developed in 1986, which involved 20 organisations in the UK. This project showed how a representation of activity modelling could be used to integrate Metier's Artemis project planner, Jaguar Cars' process planning tools and the Atomic Energy Authority's WASP Job Shop Scheduler; it was an early example of the development of an ontology for enterprise integration. This representation of tasks and actions, which was originally derived from the NONLIN [5] and O-Plan planning systems [6], was later extended to include processes; this enhanced representation was demonstrated in the Excalibur planning system [1].

As AIAI's work developed and diversified, the developers of knowledge-based systems within AIAI became increasingly interested in using a methodological approach to knowledge engineering. The emergence of the CommonKADS methodology for KBS development has focused attention on standardising descriptions of knowledge domain so that these descriptions can be re-used in other problem solving tasks. AIAI has used CommonKADS (or its predecessor, KADS) on over 20 applications, ranging in size from Course Selector [4] (which assisted students at the University of Edinburgh in complying with course combination regulations) up to the current Enterprise project. As a result, AIAI is very concerned with issues of standardisation and reuse of knowledge representations.

AIAI's most recent work has been in the area of business process modelling and workflow management. Here, too, a need has become apparent for representations which are sufficiently standardised to be used to compare models, but also sufficiently expressive to describe processes and workflow fully.

AIAI is participating in a number of current projects which have a major element devoted to the development and use of activity ontologies. The Enterprise project [3] with partners IBM, Lloyd's Register, Logica and Unilever involves the definition of an activity and process ontology within an overall enterprise ontology. This ontology has been influenced by previous work in task and action representations, which has not been set in a business modelling framework before. The article on Ontolingua and KIF in this issue of *airing* discusses the ontological work which underlies this aspect of the Enterprise project.

AIAI is a participant in the ARPA/Rome Laboratory Planning Initiative (through the continuing development of the O-Plan concept) and is one of the main motivators behind the development and adoption of the KRSL plan ontology. Here the ontology being developed allows for the representation of tasks, plans, processes, resources, authority, etc for the external communication of the planning entities. The development of KRSL is described in detail in an article in this issue of *airing*.

AIAI is also participating in the EuroKnowledge initiative, which aims to introduce standards for knowledge-level modelling. This project is in its early stages; an article describing its progress and plans can be found in this issue of *airing*.

REFERENCES

[1] Drabble, B., Excalibur: A Program for Planning and Reasoning with Processes, Artificial Intelligence Journal, Vol 62, No 1, pp1-40, July 1993.

[2] Drummond, M.E. and Tate, A., PLANIT Interactive Planners' Assistant Rationale and Future Directions, AIAI Technical Report AIAI-TR-108, June 1992. This report contains two papers, the second of which was given at the final meeting of the PLANIT Club.

[3] Fraser, J. L, Managing Change through Enterprise Models, Applications & Innovations in Expert Systems II, Proceedings of Expert Systems'94, the 14th Annual Conference of the British Computer Society Specialist Group on Expert Systems, Cambridge UK, 12-14 December 1994. Also available as AIAI Technical Report AIAI-TR-161

[4] Kingston, J. K. C., Pragmatic KADS: A methodological approach to a small KBS project, Expert Systems: The International Journal of Knowledge Engineering, 9, 4, November 1992. Also available as AIAI Technical Report AIAI-TR-110.

[5] Tate, A., Generating Project Networks, Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI-77), Cambridge, Mass. USA, 1977

[6] Tate, A., Drabble, B. and Kirby, R., O-Plan2: an Open Architecture for Command, Planning and Control, in Intelligent Scheduling, (eds, M. Zweben and M. S. Fox), Morgan Kaufmann Publishers, Palo Alto, CA., USA, 1994. Also available as AIAI Technical Report AIAI-TR-136

The EuroKnowledge Initiative

Ian Filby

1 Introduction

EuroKnowledge is an European initiative which aims to coordinate and encourage standardisation activities in the area of knowledge technology. The current focus of the initiative is on establishing recommendations for *knowledge-level* modelling — i.e. the modelling of knowledge at a conceptual level, independent of implementation concerns.

The activities of the initiative are based around a number of Actions with input from volunteers. A EuroKnowledge Project (ESPRIT P9806) has been established to help coordinate the Actions, consolidate the inputs, disseminate the results and liaise with appropriate standardisation bodies. In addition, a non-profit international association, known as the EuroKnowledge Association, has been established to bring together organisations interested in the EuroKnowledge initiative.

2 Rationale and Objectives

Europe has invested substantially in the development of knowledge technology, and has successfully deployed it in industrial and commercial applications over the past ten years. To allow Europe to undertake yet more advanced applications and to protect the investment which has already been made, it is necessary to have the ability to *re-use* and *exchange* knowledge from one system to another. Knowledge reuse enables more efficient and less costly development processes; it should lead to more predictable and reliable results, and it should enable corporate knowledge to be capitalised. Knowledge exchange includes both intra- and inter- organisation knowledge interchange and sharing.

The major aim of EuroKnowledge is to develop standards to facilitate the reuse and exchange of knowledge. To this end, EuroKnowledge proposes to establish recommendations for knowledge-level standards. It is at this level that there is an evident deficiency, and it is also here that Europe has greatest experience and renown. The initiative will address knowledge representation standards in the following three areas:

- knowledge-level representation formalisms: these are characterised by supporting a fairly direct mapping from those concepts, properties and distinctions which humans perceive in a domain to the constructs in the representation formalism. EuroKnowledge distinguishes two categories of knowledge-level knowledge, *domain* knowledge and *task* knowledge; this categorisation is used as the basis for two of the work areas in the EuroKnowledge initiative.
- knowledge-level domain ontologies: these are re-usable conceptualisations of the static knowledge in an application domain.
- knowledge-level problem solving models: these define the dynamics of the problem solving task. These include representations of methods for carrying out tasks, for example, diagnosis, design, configuration, etc.

The relationships between the technical themes are summarised in figure 1.

3 Related Initiatives

In the United States the ARPA backed Knowledge Sharing Effort (see the article on Ontologies in this issue of *airing*) is also tackling the issues of knowledge reuse and

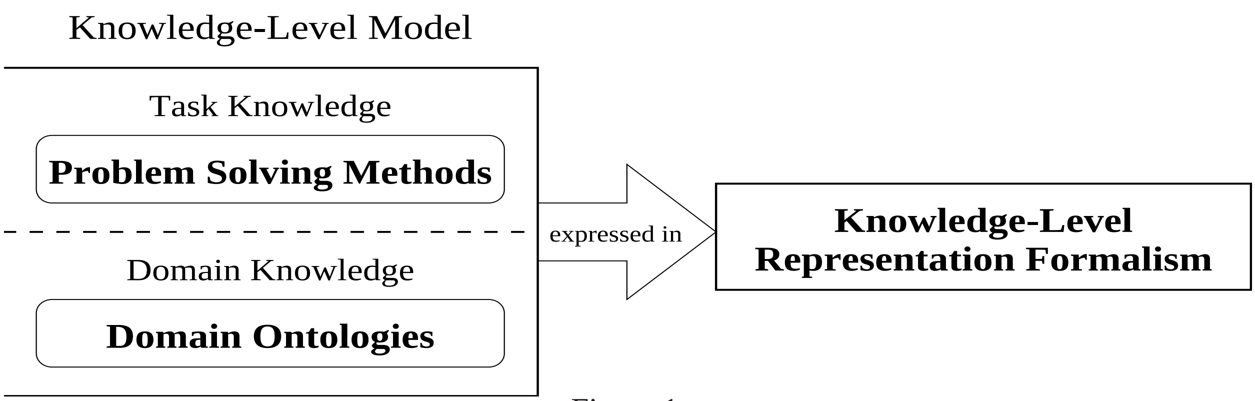

Figure 1

exchange through the development of standards. However, the standards under development are more computationally-orientated in their representation of knowledge than those proposed in the EuroKnowledge approach; their ‘source-level’ representation encodes knowledge which, although independent of implementation languages, can be directly translated into or used within implementations. The ‘knowledge-level’ European standards which represent knowledge at a more abstract level are intended to be complementary to this. One of the aims of EuroKnowledge is to clarify the difference between knowledge-level and source-level approaches to reuse and sharing.

4 Approach

The EuroKnowledge initiative will not undertake new research; it intends to build upon existing results and experience in knowledge technology. It will identify, evaluate, rationalise and report on the current state of the art. On the basis of this it will define practical recommendations on how standardisation can be achieved. The activities of the initiative will centre around focused, short-lived ‘Actions’; inputs describing, evaluating and comparing existing results will be sought from a wide variety of research groups. Inputs on practical industrial and commercial requirements will be sought outside the academic community. The initiative will also establish and maintain communication links with existing standardisation bodies.

For each of the three main standardisation themes (knowledge-level representation formalisms, knowledge-level domain ontologies, and knowledge-level problem solving models) a number of Requests For Action (RFA) will be issued. These will seek input on requirements, evaluation criteria, potential approaches, evaluation of approaches, comparison of approaches and recommendations on standardisation. The majority of responses to RFAs will be through electronic or paper submissions, although some RFAs may involve participation at workshops. The first RFAs should appear early in April 1995. Most will be of a relatively short duration requiring responses within one to three months. During late 1995 and early 1996 a number of workshops will be held at which the earlier results of the project will be presented for discussion, and consensus views on standardisation recommendations will be sought.

5 EuroKnowledge Project

The EuroKnowledge initiative has been in existence since February 1993 on a purely voluntary basis. In February 1995, a 15 month EuroKnowledge project officially started, with partial funding from the Commission of the European Union (ESPRIT P9806). The project is responsible for coordinating the Actions, consolidating and disseminating the results, and liaising with appropriate standardisation bodies. The project consortium is composed of the following members:

- Cap Gemini Innovation (F) (coordinator)
- AIAI, The University of Edinburgh (UK)
- DTK Gesellschaft fur Technische Kommunikation (G)
- CISE Centro Informazioni Studi Esperienze (I)
- Lucas KBEC, Coventry University (UK)
- Swiss Bank Corporation (CH)
- Digital Equipment Corporation (F)
- The EuroKnowledge Association

Cap Gemini Innovation, AIAI and DTK are acting as knowledge technology specialists; they provide the impetus and resources for organising and driving the Actions and consolidating and disseminating the actual recommendations. CISE, Lucas KBEC (Coventry University) and Swiss Bank Corporation have extensive experience of applying knowledge technology in their respective industrial / commercial environments; they take a proactive role in assuring that the standardisation recommendations have practical relevance. DEC assures two-way communication with standardisation bodies. Volunteer activity from EuroKnowledge Association members supplies input to the Actions. Communication and dissemination efforts are partially sub-contracted to AI Intelligence (UK).

The EuroKnowledge project will seek to reflect a consensus view in its practical recommendations on the standardisation of knowledge-level representation formalisms, domain ontologies and problem solving models. The EuroKnowledge initiative (which will continue beyond the end of the project) will seek to establish knowledge technology standards, based upon the project’s recommendations. It must be stressed that the project consortium does not view standardisation as meaning a single approach and believes it is quite possible that the recommendations may be for a set of explicit alternatives. Nor is it expected that any future standards would be immutable although future changes should build upon the earlier standards.

6 EuroKnowledge Association

A non-profit international association, known as the EuroKnowledge Association has been established to bring together those organisations interested in participating in EuroKnowledge activities and those organisations interested in staying abreast of the EuroKnowledge developments. It is envisaged that the EuroKnowledge Association will be responsible for carrying on the standardisation initiative once the EuroKnowledge Project is complete.

7 Dissemination

Dissemination will be achieved via a range of mechanisms:

- The World Wide Web (WWW) as the electronic medium for far-reaching dissemination (URL *http://www.aiai.ed.ac.uk/~euroknow/*);
- In-the-Knews Forum (quarterly newsletter) as the reporting mechanism for the EuroKnowledge Project to the CEU and the EuroKnowledge Association;
- EuroKnowledge News in *AI Watch* as a public printed antenna for EuroKnowledge activities;
- EuroKnowledge Updates in *airing*;
- Presentations at relevant workshops and conferences.

Association-centred workshops will be organised on project topics, and an international workshop will be held at the conclusion of the project, in order to report on results to the knowledge technology community.

During the lifetime of the EuroKnowledge project, a regular update on the initiative will appear in *airing*. For further details about the EuroKnowledge Initiative, Project or Association see the World Wide Web page or contact Ian Filby at AIAI.

The ARPA Knowledge Sharing Effort: KIF and Ontolingua

Mike Uschold

1 Introduction

The ARPA Knowledge Sharing Effort [6] [9] is intended to develop standards within the knowledge engineering community, in order to promote and facilitate re-use. It consists of the following four initiatives:

1. An Interlingua for Knowledge Interchange: This initiative aims to develop an approach to translating between knowledge representation languages. An interlingua has been developed based on first order logic. It is called the Knowledge Interchange Format (KIF). Sharing is accomplished by translating a knowledge base from one language into KIF, and then from KIF to another language.

2. Knowledge Representation System Specification: This initiative aims to build a single language based on the KL-ONE family of knowledge representation languages. It promotes knowledge sharing by removing arbitrary differences in languages within the same paradigm.

3. External Interfaces: The main aim of this initiative is to facilitate the ability of one knowledge based system to access information from another knowledge based system or possibly a conventional database. A query language is being developed called KQML, which will be to knowledge representation systems what SQL is to database management systems.

4. Shared, Re-usable Knowledge Bases: This initiative aims to overcome difficulties in knowledge sharing due to lack of consistency between knowledge bases with respect to vocabulary, semantics, and underlying assumptions. A first step to this end is the Ontolingua project which will allow the definition of portable ontologies (i.e. ways of conceptualising a domain). The language for representing ontologies is based on KIF, and is not committed to any particular knowledge representation system.

The rest of this article describes two of these endeavours - KIF and Ontolingua - in more detail. These will be used on the Enterprise project which AIAI is participating in.

2 KIF

KIF [5] is an interlingua for knowledge bases. The goal is that knowledge bases written in any language can be translated into KIF, which can, in turn be translated into the knowledge representation language of choice. KIF is designed to be state of the art, i.e. able to represent most/all of the important concepts and distinctions available in today’s advanced knowledge representation languages. KIF is based on predicate calculus, but is extended to cater for advanced capabilities such as defining terms; representing knowledge about knowledge; specifying sets; and encoding commonly used facilities for non-monotonic reasoning. KIF includes “model-theoretic semantics for the language and an axiomatization of the primitive object types such as sets, lists, relations, and functions" [2].

The syntax of KIF is Lisp-like. The idea is that KIF will evolve in various layers. There will be a core layer, analogous to the primitives in Lisp. Outer layers will provide facilities for idioms and various extensions that make the language more usable. This is analogous to the set of functions provided in Common Lisp.

KIF is not an implemented knowledge representation language; the intention is that KIF specifications represent knowledge at an implementation-independent level in a clear unambiguous manner. Where practical, key assumptions (e.g. constraints or relationships) are formalised as KIF expressions; otherwise they are expressed in natural language text.

3 Ontolingua

The main goal of the Ontolingua project [2] is to facilitate sharing and re-use of knowledge bases. There are a variety of technical barriers to this goal. The most important ones are:

1. The diversity of existing knowledge representation languages. For example, a knowledge base which includes a rich representation for non-monotonic reasoning will be difficult to use by a system that does not support this.

2. The different assumptions made when conceptualising a domain. All knowledge bases are based on some (usually implicit) conceptualisation of a domain. This conceptualisation is often linked with the purpose of the representation (e.g. diagnosis vs. design).

The work on KIF helps address the first problem. Ontolingua is intended to address the second problem. The conceptualisation of a domain consists of the objects, concepts, and other entities and the relationships that hold between them for a particular domain [1]. An explicit representation of such a conceptualisation, including underlying assumptions, is called an ‘ontology’ [2] [3] [4]. It is worth noting that ontologies are partial specifications; much is necessarily left out.

The term Ontolingua is used both to refer to the formal language for representing ontologies, and the system for processing them. The language is essentially an enhanced form of KIF. The chief addition is a special facility for representing frame structures which are common to most knowledge representation languages.

Sharing and reuse is accomplished by translating the ontologies into one of several knowledge representation languages; currently, Ontolingua supports CLIPS, Generic Frame, Loom, Epkit, Algernon, and (pure) KIF. The output of Ontolingua becomes part of the knowledge base for a particular application. Reuse is accomplished because many similar applications in different languages can use the same ontology. Indeed, one application may use more than one ontology.

The idea is that new target languages will be catered for by development of new translators between Ontolingua and that language. These translators can be developed independently of any other translators which might exist. Ideally, such translators would be fully automatic, but not all Ontolingua statements can be translated into all representation languages. Users are informed by a message when translation is not possible.

The concept of ‘ontological commitment’ is important. It is defined as follows: an agent commits to an ontology if and only if the agent’s actions are consistent with the ontology. In the case of building software, the programmer is the one committing to the ontology in programming the behaviour of the software. The software’s behaviour must be consistent with the ontology. Commitment to an ontology guarantees consistency, but not completeness.

Ontologies are specified using KIF syntax and semantics. In general, there will be much important information that may be neither easy nor worthwhile to formalise. For this, Ontolingua provides a facility for including natural language comments. These are not parsed by Ontolingua, but are passed directly to any documentation ‘place holders’ in the target languages. Ontolingua translates declarative statements into axioms that constrain and manage the defined terms.

Figure 1 shows an example Ontolingua definition from a bibliography ontology.

Facilities available within Ontolingua include a cross referencer, a syntax checker, limited consistency checking, and most recently a sophisticated hypertext editor designed to allow co-development by people at many different locations over the web. This editor is available from [7].

Significant ontologies where there is wide agreement resulting from much careful discussion have great potential of influencing standards. They are likely to be consulted by formal bodies producing standards. Also, if they are publicised and become widely used, they become *de facto* standards.

Ontolingua is available on line. For further details, see [8]. There is also an Ontolingua mailing list; to subscribe, send email to:

ontolingua-request@ksl.stanford.edu

More information on Ontolingua and on related work in enterprise modelling is available in [9].

Figure 1: Example Ontolingua Definition, from a Bibliography Ontology

4 References

[1] Genesereth, M. R. & Nilsson, M. J. (1987). Logical Foundations of Artificial Intelligence. San Mateo, CA: Morgan Kaufmann.

[2] Gruber, T. R. (1993). A translation approach to portable ontology specifications. Knowledge Acquisition, 2, 199-220.

[3] Guarino, N. (1995), Ontologies and Knowledge Bases: Towards a Terminological Clarification, submitted to the 2nd International Conference on Building and Sharing Very Large-Scale Knowledge Bases, 1995.

[4] KACTUS (1995), KACTUS: Modelling Knowledge about Complex Technical Systems for Multiple Use, ESPRIT project 8145. Information available from the University of Amsterdam on the World Wide Web:

http://www.swi.psy.uva.nl/projects/Kactus

[5] KIF manual (1995), Manual for the Knowledge Interchange Formant, version 3. Available on the World Wide Web from:

http://WWW-KSL.Stanford.EDU:80/knowledge-sharing/kif/#manual.

[6] Neches, R., (1994), Overview of the ARPA Knowledge Sharing Effort, Available on the World Wide Web from:

http://www.ksl.stanford.edu/pub/knowledge-sharing/papers/kse-overview.text

[7] Stanford Ontolingua (1995). A Guided Tour to Developing Ontologies using Ontolingua, WWW address

http://www-ksl-svc.stanford.edu:5915/doc/frame-editor/

[8] Ontolingua (1995). Ontolingua home page, WWW address

http://www-ksl.stanford.edu/kst/ontolingua.html

[9] Patil, R.S., Fikes, R. E., Patel-Schneider, P. F., McKay, D., Finin, T., Gruber, T. & Neches, R., (1992), The DARPA Knowledge Sharing Effort: Progress Report, published in The Proceedings of KR'92, The Annual International Conference on Knowledge Acquisition

[10] Uschold, M. et al., (1994), State-of-the-Art Survey of Enterprise Modelling, Part 4. Available from AIAI.

The ARPA/Rome Laboratory Planning and Scheduling Initiative:

KRSL Planning Ontology

Brian Drabble

1 Introduction

The ARPA/Rome Laboratory Planning and Scheduling Initiative (ARPI) is a US funded initiative whose purpose is to provide tools and techniques to support the logistics operations of US military forces. One important aspect of the initiative is the development and use of a planning ontology which can be used to describe each of the various aspects of planning within this domain, e.g. tasks, actions, constraints, processes, etc.

The ontology referred to as the Knowledge Representation Systems Language (KRSL [1]) began development in 1989 and was designed around the knowledge specification document written by James Allen and the KA/KR Issue Working Group. The goal of KRSL is to provide a common set of functions for representing planning related primitives. The aim of the language is to provide sufficient expression to capture domain knowledge while still being independent of the implementation languages (LOOM and CLOS) chosen for the initiative. Individual researchers can then develop translators for their own specific representations. For example, AIAI will develop a translator to convert from KRSL to O-Plan's Task Formalism Language [2], based on the Knowledge Interchange Format (KIF) described in an article in this issue of *airing*. Translators into LOOM and CLOS are also being developed as part of the KRSL development task.

KRSL was originally intended to provide an interchange medium for ARPI systems and a means for specifying shared domain information. However, KRSL has not been exploited as much as desired for at least two reasons. Firstly, it did not provide sufficiently convenient and modular expressive mechanisms for some information, especially in an environment in which extant systems used different subsets or different organisations of concepts. Secondly, for other types of planning information needed by extant and intended systems, it did not provide any standard expressive mechanisms at all.

In order to address these problems a new initiative has been launched within the ARPI to carry through the initial aims of KRSL. The aims behind the new KRSL initiative are to develop an open, modular, clarified, and sharable ontology of planning information for use by ARPI participants and by other efforts (both ARPA and non-ARPA). AIAI is one of the main motivators behind this new initiative.

The rest of this article describes the main structures of the KRSL language and AIAI's background in the development and use of KRSL.

2 KRSL Language Description

The aim of this section is to provide an overview of the main KRSL structures and to show examples of the ways in which KRSL can be used.

The primitive classes in KRSL cover the categories of time, measurement, resources, space and plans. Examples of the categories are now given:

Time:

Time is represented in KRSL in terms of durations, duration bounds, time points and time intervals. These time primitives can either be asserted/retracted or created/deleted.

Measurement:

Measurement in KRSL is represented by unit-types e.g. feet, miles, and bricks, and quantities specify the maximum and minimum values for an object.

Resources:

Resources define the capabilities and capacity of various objects and activities e.g. some objects may have generic characteristics as providers, consumers or producers of resource. For example, a transport plane provides cargo-space and consumes fuel when it flies.

Concepts:

Concepts can be roughly viewed as defining the objects in a domain in terms of their inheritance, attributes, decompositions and the resources provided, required and produced by an object.

Relations:

Relations are the building blocks for creating propositional sentences and can be defined as being temporal or atemporal.

Plans:

Plan primitives are used to specify the behaviour of an application domain and are composed of:

- axioms: general rules of the domain which should always be true.
- causal rules: causal behaviour of the domain in terms of sets of preconditions, effects and likelihoods
- preference relations: a general mechanism for specifying preference ordering between different world descriptions.
- plan: plans are applied to produce some particular effect or set of effects.
- event: primitive actions in the domain and are defined in terms of their effects on the world.

For each type of primitive which can be instantiated, KRSL provides a function for instantiation. This function has the form **make-<x>** where **<x>** is the name of the primitive. A few primitives also allow user class specification (as opposed to simple instantiation); most allow nesting of primitives. All primitives must have a primitive description and must either be an instance of a primitive or a list of the appropriate form to create an instance of the primitive. They are all built upon a class called KRSPrimitive which contains attributes for names, documentation, annotations and modification records and behaviours for book keeping in the KRS.

Details of KRSL are available on line and further information can be obtained from the World Wide Web page:

http://isx.com/pub/ARPI/ARPI-pub/krsl/krsl-info.html

This contains pointers to the current KRSL ontology, reference manual and to the outcome of KRSL workshops.

3 References

[1] ARPA/Rome Laboratory Planning and Scheduling Initiative, Knowledge Representation Specification Language (KRSL) Reference Manual (eds, Allen, J. and Lehrer, N.), version 2.0, September 1991. ISX internal technical report, ISX, 4353 Park Terrace Drive, Westlake Village CA 91361, USA.

[2] Tate, A., Drabble, B. and Kirby, R., O-Plan2: An Open Architecture for Command, Planning and Control, in Knowledge Based Scheduling, eds M. Fox and M. Zweben, 1994. Publisher Morgan Kaufmann, Palo Alto, California, 94303, USA.