Third
INTERNATIONAL EXPERT SYSTEMS Conference

Third INTERNATIONAL EXPERT SYSTEMS Conference

London

2 – 4 June 1987

Sponsored by

Expert Systems

The International Journal of Knowledge Engineering

Organised by Learned Information (Europe) Ltd
Publishers of *Expert Systems Journal*,
Expert Systems/AI In Business, *Online Review*,
Electronic and Optical Publishing Review,
The Electronic Library, *Monitor* and
Information World Review

Learned Information **Oxford and New Jersey**

Contents

REPRESENTATION TECHNIQUES FOR DATA, INFORMATION AND KNOWLEDGE IN AN INTERACTIVE PLANNING ASSISTANT

F.N. Teskey, A.J. Pearce, Brighton Polytechnic, UK

Abstract: The Alvey PLANIT Club has recently completed the development of a prototype Interactive Planning Assistant (IPA). The IPA embodies a wide range of data, information and knowledge about planning. This paper describes the types of data, information and knowledge that are used and shows how they can be incorporated into a formal model. The model is developed to the stage where it can be used as the basis for a production version of the IPA. The paper concludes with a discussion of the possible applications and advantages of such a system based on the experience of the PLANIT Club.

1 INTRODUCTION

The aims of this paper are:-

- to analyse some of the experiences of the Alvey PLANIT Club in developing a prototype interactive planning assistant (IPA),
- to develop a formal model for representing data, information and knowledge based on this experience,
- to investigate how this model could be used to build a production IPA, and
- to review the likely applications and advantages of such a system.

2 THE PLANIT CLUB

2.1 Background

As part of the Government's Alvey programme a number of Clubs, including PLANIT, were set up to increase industrial awareness of new technologies. The aim of the PLANIT Club is to promote awareness of expert system techniques in the field of project management. To this end the club has designed and produced an interactive planning assistant. The system is implemented on a Texas Instruments Explorer using the KEE software.

2.2 The Role of the IPA

The IPA is intended to act as a tool for planners at all stages of a project, from the overall planning of the project, through the planning of the separate processes down to the scheduling of the individual operations. It is not intended that the system itself should undertake planning but rather that the system should act as an assistant to a human planner. The system should provide answers to questions such as:

- **is the project likely to complete on budget?**
- **how can this part of the project be expedited?**
- **what alternative processes can be used?**
- **can the scheduling of these jobs be improved?**

The IPA has been developed as a general purpose tool. However within the scope of the PLANIT club it has been possible to test the IPA on only one major example, the design and production of a tanker for transporting liquified gas.

2.3 The Tanker Example

The aim of this example is to study all the stages of planning a small project using the IPA. The project is to design and build a liquified gas tanker. The functional requirements for the tanker, volume of gas to be carried, temperature, pressure etc., constrain the design of the tanker but also leave some freedom in the choice of processes to fabricate parts of the tank. The choice of processes will, in turn, constrain the scheduling of the work amongst the available resources. To ensure that the impact of changes at one level of the system can be propagated to other levels it is essential that a uniform knowledge representation is used.

At the time of going to press the results of the test runs of this example are not available. However the results will be presented at the conference to illustrate the working of the IPA.

3 THE PROTOTYPE INTERACTIVE PLANNING ASSISTANT

3.1 The Functions of the IPA

The essential idea behind the work of PLANIT is that plans generated to achieve specific goals are inherently 'knowledge rich'. The process of creating a plan requires knowledge of planning techniques in general and of the business organisation, products and processes in a specific application. The main function of the IPA is to enable such knowledge to be represented explicitly and to be used for the intelligent validation and modification of plans with advice and explanation facilities provided.

A plan drawn up by a human planning engineer represents a sequence of activities or processes. These in turn represent the conclusions of many decisions taken by the human planner. The decisions are based upon his knowledge and experience of the specific planning domain and the general process and techniques of planning. The knowledge content remains within the domain expert (the human planner) whilst the plan produced contains data and information.

As an alternative to the human planner, the automatic generation of goal driven plans is a current research topic (Ref 1). Plans created in this manner contain some of the reasoning behind their construction. The reasoning is conducted using time, resource and cost data plus activity pre and post conditions. The conditions tend to be object oriented such as products. If the post-condition of activity 'x' equals the pre-condition of activity 'y', then the plan generator can reason that the two activities 'x-y' form a feasible ordering.

It may also be true that a second pair of activities 'a-b' form a feasible ordering and that the pre-conditions of 'x' and 'a' are equal and that the post-conditions of 'y' and 'b' are equal. The plan generator can reason that 'x-y' and 'a-b' are valid alternative sub-plans. An attempt can then be made to rank the sub-plans by considering the times, resources and costs involved using pre-set criteria. Plan generators of this type therefore contain and use some knowledge and can be interrogated about the reasoning performed.

Human planners and plan generators use data, information and knowledge to construct a plan. The data and information contained in the plan are derived from, but are not identical to, the data and information required to construct the plan. The differences are in terms of modified data values and selections of available alternatives. New data and information are created as a result of plan construction.

In the same way, new knowledge is created which a human planner adds to his experience and a plan generator can use to modify its control strategy. For example, the reasons why an activity duration was modified or a particular process was selected can be stored for future reference and evaluation, ie, a learning process. It is in this area of planning knowledge and experience that the work of PLANIT is centred.

The IPA is not intended to act as a plan generator, but as a planning assistant. The types of knowledge incorporated into the IPA and its ability to reason about activity ordering models are considerably enhanced. This enables the IPA to validate suggested plans and to advise the user about plan modifications.

3.2 Plan Modification

In traditional project planning, the concept of plan modification is thought of in terms of adjusting activity durations within available float, using alternative or additional resources, adjusting relative priorities in multi project scheduling or changing the plan logic. Where plans have been constructed from a library of sub-plans, it is possible to include a number of equivalent but alternative sub-plans, to achieve a specific sub-goal or key activity.

This type of plan modification can be thought of as a hierarchy of permissable actions and consequences. Increasing a single activity duration within its available float may not affect any other activity and therefore has limited consequences beyond the consumption of additional resources or incurring additional costs. At a higher level in the hierarchy of permissable plan modifications, part of a plan may be replaced by an alternative sub-plan.

This second type of action is altogether more significant than increasing an activity duration. A replacement sub-plan may involve many activities requiring different quantities and/or categories of resources. The human planner may need to consult with other people in order that the plan modification decision can be made. The new sub-plan may involve a different manufacturing process which although 'technologically possible' is not 'normally used'. The planner may have to consult with a process planning engineer to discuss the application of the possible sub-plan in the current case.

Similarly the alternative sub-plan may also be more expensive than the original and the planner may have to consult with management to determine whether the additional cost is acceptable. It is therefore clear that higher levels in the plan modification hierarchy require a wider span of knowledge and a higher level of authorisation before the decision to modify the plan can be taken. The planner has to consult experts of additional domains of knowledge and had to decide which additional domains are relevant to the current decision.

It is also normal that complex actions at high levels in the plan modification hierarchy are used less frequently than simple actions at the base of the hierarchy. The planner is less familiar with seldom used plan modification actions and may need more advice on how to implement them correctly, or conversely he may implement them less efficiently. In this case further plan modifications are necessary to correct the errors introduced.

It is at this level of the planning process that the IPA is of significant help to a human planner. The IPA can make inferences about the state of a project and the correct course of action to advise. It does this by searching the appropriate structures of data, information and knowledge to determine attribute values in rules. It is also possible for the IPA to learn by asking the user to assert attribute values in new cases for use by inference in future similar cases.

4 REPRESENTATION TECHNIQUES IN THE IPA

4.1 Semantic Nets

The representation techniques adopted in the planning domain vary greatly. Traditional techniques represent data and information eg. activities, durations, dates, resource requirements and availabilities, costs, calendar information, process details and activity sequencing alternatives. Records and arrays held in sequential files are quite adequate for storage and calculations using these items. However for the representation of knowledge or for the inferencing processes inherent in the manipulation of knowledge, information and data, these structures do not possess the required characteristics.

The IPA makes use of more appropriate representation techniques. The content and structure of a plan which may be drawn on paper as an activity diagram (CPA or PERT network diagram) can also be thought of as a semantic net. Activity diagram activities and activity sequences are equivalent to semantic net nodes and labelled arcs, where arcs are used to represent only the single relationship of 'succeeding activity'. By attaching attributes to the nodes, all the data and information of the activity diagram is transferred to the semantic net.

This idea can be developed further to include other types of information about the plan. Additional labelled arcs can be introduced to represent other relationships between activities eg. causal relationships such as:-

activity b	'usually follows'	activity a
activity b	'logically follows'	activity a
activity b	'technologically follows'	activity a

These relationships are used to form inferences about the construction and modification of a plan and the results are displayed in the form of interactive advice or explanation facilities.

This approach of identifying a variety of relationships or types of information about adjacent nodes enables us to represent alternatives. One of the options available to a planner in modifying a plan is to make use of alternative resources or activities. Information about which alternatives exist and, to a certain extent, information about which alternative to select, can be represented using labelled arcs in semantic nets.

A further relationship, 'is part of', can be used to represent decomposition, or in planning terms, the use of sub-networks or sub-plans. One node on a semantic net can be expanded to represent all the detailed activities required to achieve the single higher level activity. By combining these two features of alternatives and decomposition, the ability exists for the IPA to give advice and explanation to the planner regarding alternative activities, resources and sub-plans with nesting of alternatives if required.

This type of plan modification activity is one where the human planner gains practical benefits from the use of a planning tool such as the IPA. For example, the planner can use a 'what if?' approach to test if a modification is technologically possible or if a sequence of activities is logically correct. This feature is not available in traditional project management systems and is achieved in the IPA by checking for consistency within sets of relationships.

4.2 Petri Nets

The hierarchy of plan modification actions includes associated levels of required management authorisation. In traditional project planning systems, the authorisation could be viewed as a consumable resource which has to be available for an action to proceed. More realistically the required authorisation is a pre-condition which must be satisfied before the plan modification action may be implemented. Similarly activity resource requirements can be viewed as pre-conditions which must be satisfied before an activity can start.

[The authorisation process itself may be viewed as an example of constraint relaxation, necessary for the accomplishment of the plan goals. The representation of constraints as attribute values is recognised within the IPA but that of constraint relaxation is the subject of current research (Ref 2).]

Other items of data exist about the project such as the required completion date, the target cost, the maximum number of staff to be used, the product surface finish to be produced or the maximum amount of raw material to be consumed. These items of data may also be viewed as constraints which must not be exceeded or as activity post-conditions which must be true if the activity is to be successfully completed.

There are several types of plan knowledge, including activity conditions, which cannot be represented within semantic nets. For these the IPA uses Petri Nets (Ref 3) which have the basic components of Places, Links, Tokens and Transitions. In particular, tokens can represent pre and post conditions of activities and transitions enable the passing of a non-consumable resource from one activity to the next.

In addition to representing activity conditions and non-consumable resources, the properties of Petri Nets allow for iteration and redundancy checking. This enables the IPA to provide further plan validation functions. Iteration checks can be made to ensure that no loops exist within the plan logic. Redundancy checks are used when modifying plans to ensure that only one of a number of alternative sub-plans is included in the plan.

4.3 The KEE Implementation of the IPA

The main method of knowledge representation used in the KEE implementation of the IPA is the frame hierarchy. Frames and slots can be used to implement all the aspects of data, information and knowledge represented as semantic and Petri nets. In common with most other frame based systems KEE supports a single taxonomic hierarchy using the sub-class, or 'a kind of', relationship. While this is adequate for the objects to be represented it is not ideal. In the IPA, the compositional hierarchy (the 'is part of' relation) is as important, if not more important, than the taxonomic hierarchy. For example, before we change the design of a particular component, we need to know all the products of which it is a part.

This information can be represented by additional slots in the relevant frames, however it would be more natural to represent it as a frame hierarchy. The problem is more serious when we come to represent activities. Clearly much of the knowledge in the IPA is about activities and their relationships. As well as the taxonomic and compositional relationships required for objects, the IPA has identified the need for causal and temporal relations between activities. While these can be viewed conceptually using semantic nets as discussed previously, the implementation problems remain.

As an example, it is not possible to bolt two components together until the bolt holes have been drilled. Thus the activity 'bolt together' causally follows the activity 'drill holes'. The relationship 'logically follows' exists between the activities 'drill holes' and 'bolt together'.

In a specific plan which includes these two example activities, 'bolt together' should temporally follow 'drill holes' with the relationship 'successor of' between the two activities. The IPA will validate the plan construction by searching for causal relationships between the two activities, inferring the correct temporal relation and checking that the plan logic does not contradict that temporal relation. These relations are illustrated in the following diagram.

```
                         successor of
drill holes ----<---------------------- bolt together
     |                                        |
     |                                        |
     |             logically follows          |
drill holes ----<---------------------- bolt together
```

In an attempt to rationalise the model we can identify three separate parts. Firstly data, which is specific details such as machine availability or duration of operations, is represented as specific values in slots. Secondly information, which is structured collections of data, is represented as frames and relations between frames. Thirdly knowledge, which is collections of rules for manipulating information and data, is represented by LISP code attached to frames and slots. In the following section we attempt to rationalise these different representations into a single model.

5 A FORMAL MODEL FOR DATA, INFORMATION AND KNOWLEDGE

5.1 Data, Information and knowledge

As we have moved from database management systems, to information retrieval systems, to knowledge based systems, there has been no clear distinction between data, information and knowledge. We suggest that a clear distinction should be made based on the notion of data as directly observable facts, information as structured collections of data and knowledge as methods of using or modifying information. A further distinction needs to be drawn between user level and meta-level.
The ability to reason at the meta-level (Ref 4) has been identified as an important part of information systems, yet in many cases the meta-level knowledge and information used to control the systems is not explicit. There is a need for an explicit and distinct meta-level within the model.

5.2 The Mathematical Model

An information model is a mapping

$A : E \longrightarrow V$

where E is a set of entities,
V is a set of values and
A maps each entity to the value attributed to it.

As in the standard entity relationship model (Ref 5) the entities are regarded as tokens or icons representing some abstract or concrete object or concept in the real world. The values,however, are not regarded as atomic, as in the original relational model (Ref 6) or in many subsequent developments (Ref 7, Ref 8) and others. Instead the value is regarded as an analogic representation of the related object in the real world. Thus, in a simple case the value for an entity representing 'temperature' would be a single scalar value, for a more complex entity such as 'average temperature' the value would be a procedure, or analogue, for calculating the average temperature. It is important to realise that the mathematical model does not place any restrictions on the type of values that are allowed; the values can take the form of any valid mathematical object or formula involving the elements of the model. It is up to the implementation of the system to provide the tools necessary to manipulate complex values and it is up to the interpretation of the model to ensure that the values are meaningful.

For those familiar with category theory (Ref 9) this forms a simple category. As we add more structure to the model at the meta and user level, then the structure of this category becomes richer. A more rigorous definition of these levels could be made within the framework of category theory, though this is outside the scope of the present paper.

This model can be used to formalise the difference between knowledge, information and data that we have already discussed informally. Thus the notion of data as directly observable facts is formalised by the definition of a data element as an entity and its associated value, and information, as a structured collection of data, is defined as relations between entities. Since these represent views of the real world, it is likely that the definitions could be rephrased in terms of physical concepts along the lines proposed by Stonier (Ref 10). Our informal definition of knowledge referred to methods of using information. We would claim that the main use of information is to change one's model of the world; hence knowledge should be defined as a mapping from one model to another.

5.3 Further Developments Using Category Theory

We have seen that in both the IPA and the formal model developed from it, relations are used to represent information and knowledge is represented by rules to modify those relations. We now want to study in more detail how these various relations interact. Consider, for example, the following diagram:

```
           successor of
      A --<---------------- B
      |                     |
      |                     |
      |                     |
 is a ^                     ^ is a
      |                     |
      a                     b
```

Since entities of type B succeed entities of type A we can deduce that 'b' is the 'successor of' 'a'. This is an example of what, in category theory, is called a pull back diagram - the relation 'successor of' can be pulled back through the 'is a ' relation. We shall say that the 'successor of' relation respects the 'is a' relation; note that the converse is false, the 'is a' relation does not respect 'successor of'.

In addition many other standard definitions can be applied to the relations we are using. Of particular interest are the definitions of transitivity and implication. So, for example, the compositional relation 'is part of' is transitive and we can express the law of causality by saying that the causality relation implies the temporal relation. We claim that much if not all of the knowledge in the IPA could be expressed in these terms rather than, as at present, in Lisp code.

6 ALTERNATIVE IMPLEMENTATIONS FOR THE IPA

6.1 The Need for Alternative Environments

The existing environment for the IPA is the KEE/Explorer system. This was used to develop three initial prototypes for the areas of project planning, process control and job shop scheduling. Using the experience gained in developing these prototypes and the expertise of the working group, a functional specification for the IPA was produced (Ref 11). This document does not refer to KEE but does assume the existence of KEE-like facilities in the implementation environment. The final version of the IPA was implemented in KEE using this specification. It was realised, however, that KEE did not provide an ideal environment for a delivery system. It was decided to look at alternative environments in which the IPA could be delivered.

6.2 Limitations of KEE

KEE provides a very powerful environment for the development of intelligent knowledge based systems and it would have been impossible to develop the IPA in the time available without a system such as KEE. However, it has a number of limitations as a delivery vehicle for the IPA.

First, though KEE provides a wide range of functions, it does not at present provide either direct access to any external graphical routines or a direct interface to external programs or data. Thus the graphical display of project networks and other data is limited to the formats predefined by KEE; though these are adequate for a demonstration system more control over the graphics display is required. The lack of external interfaces is more serious. One of the aims of the IPA is to take project plans from other systems and see how the plan can be implemented. For example, it is intended to take the output from the WASP job shop scheduling package and use it as input to the IPA. At present this data would have to be entered manually into the KEE system.

The second problem with KEE is that the user interface has been designed for a knowledge engineer and is not suitable for a non-computer expert. The system offers too many options to the user and the screen can rapidly become overcrowded with windows and menus. The problem is more serious because the IPA user may need to create new classes in the knowledge base and so needs access to the KEE development environment. For example, the introduction of a new type of machine into a job shop would require the creation of a new class.

Finally, KEE, and the hardware needed to run it, are expensive. Even allowing for the most generous discounts, the cost of the hardware and software is in the region of £60,000. Many applications which could benefit from the IPA could not justify this capital cost.

6.3 Requirements for the Delivery Environment

The specification of the IPA assumes the existence of certain functions in the implementation environment. The central requirement is for a method to represent frames, slots and demons. The IPA requires that these are grouped into a number of hierarchies - taxonomic, causal, temporal and compositional. Most AI environments, KEE included, provide only a single taxonomic hierarchy; so the requirement is for a system that will allow the user to define additional hierarchies. As was indicated in previous sections, there is a need for interactive graphics and interfaces to other systems. Though the original requirement was for a system that would run on a small personal computer, there is considerable advantage in having a system that is supported on a wide range of machines and will provide an easy upgrade path. Finally, the system must be affordable.

The KES system (Ref 12) has been chosen as a basis for a PC implementation of the IPA. Work is in hand in developing this implementation and the progress on this will be reported at the conference.

7 FUTURE APPLICATIONS OF THE IPA

7.1 Domain Specific Expert System Shells

The IPA represents a new development in expert systems (Ref 13). The existing expert system shells are all general purpose tools. For each new application it is necessary for a domain expert working together with a knowledge engineer to develop the knowledge base. In any particular domain, however, much of the knowledge base will be common to a range of applications. So if that knowledge could be encoded once and for all then the development of expert systems for specific applications in that domain would be that much easier. By separating the general knowledge about a domain from the specific information about the application it should be possible to develop domain specific expert shells. The IPA is a prototype of such a system.

7.2 Domains of Application

The IPA has been designed for one specific domain - Project Management in an engineering environment. It should be noted, however, that this is much more general than most expert systems. Furthermore the range of expertise in the PLANIT club has contributed to the generality of the system. We can envisage a range of applications for the system.

At the first level the system can be used for planning projects such as the tanker example we have already considered. The important criterion is that the IPA is being used in an organisaiton where a number of different products are being manufactured from a range of standard and semi-standard components. Installing the IPA in such an organisation would involve loading details of the components and processes that are used into the existing IPA hierarchies. The system could be used to help in the planning of new products. Given a top level design of the product, the IPA could then be used to develop and evaluate more detailed lower level plans and monitor the progress of the project.

At the next level, the system could be used for planning other types of projects. We are currently investigating a number of areas including design of aircraft simulators and plant shut down and maintenance. + These applications require a similar structure to the knowledge base, in terms of the basic relations, but differing domain specific information.

Lastly we would like to consider applications of the techniques we have developed for domain specific expert system shells to domains other than planning. However, work on this area will require more experience of the use of the system in the planning domain.

8 CONCLUSIONS

In this paper we have looked at some aspects of the Interactive Planning Assistant developed by the Alvey PLANIT Club. In particular we have identified the role of data, information and knowledge in the IPA and produced a formal model for this. Though much work remains to be done on this model it has formed the basis for a revised implementation of the IPA. It is intended to develop this to a deliverable product. Based on our experience of the prototype IPA, this should provide considerable benefit in a range of applications.

+ This work is independent of the PLANIT Club.

F.N. Teskey and A.J. Pearce, Brighton Polytechnic, Information Technology Research Institute, Moulsecoomb, Brighton, BN2 4GJ.
Telephone 0273-693655 Telex 878422 BUS BRI

REFERENCES

1 Tate, A., Whiter, A., 'Plan generation which accounts for preferred resource usage and resource limits in NONLIN', Applications of AI Conference USA, (1984)

2 Grant, T., 'Categories of constraints in acquiring planning and scheduling knowledge,' Internal Report, Brunel University, (1986).

3 Peterson, J. L., 'Petri Nets', A.C.M. Computing Surveys, Vol.9, No. 3, pp.223-252, (1977).

4 Cunningham, J., 'Comprehension by model-building as a basis for an expert system,' Expert Systems 85 - Proceedings of the Fifth Technical Conference of the British Computer Society Specialist Group on Expert Systems, ed. M. Merry, Cambridge University Press, (1985).

5 Chen, P.S., 'The entity relationship model', A.C.M. Transactions on Database Systems, Vol. 1, pp.9-36, (1976)

6 Codd, E.F., 'A relational model of data for large shared data banks,' Communications of the A.C.M., Vol. 13(6), pp.377-387, (1970).

7 Azmoodeh, M., Lavington, S.H., and Standring, M., 'The Semantic binary relationship model of information,' Research and Development in Information Retrieval, ed. C.J. van Rijsbergen, Cambridge University Press, (1984).

8 Frost, R.A., 'Using semantic concepts to characterise various knowledge representation formalisms: A method of facilitating the interface of knowledge based system components,' The Computer Journal Vol. 28(2), pp.112-116, (1985).

9 Maclean, S., Catergories for the working mathematician, Springer Verlag, New York, (1971).

10 Stonier, T., 'What is information?', Expert Systems 86 - Proceedings of the Sixth Technical Conference of the British Computer Society Specialist Group on Expert Systems, Cambridge University Press, (1986).

11 Smith, J.U.M., Higgins, R., 'Planit Alvey Community Club functional specification of the prototype interactive planners' assistant, issue 2', Internal Report, Systems Designers Plc (1986)

12 Knowledge Engineering System - Knowledge Base author's reference manual for KES release 2.1, Software Architecture & Engineering, Inc., Arlington, U.S.A., (1986).

13 Owen, K.,ed. Alvey conference report 1986,pp. 17, IEE Publishing Dept., Stevenage, (1986).

A Appendix

A.1 The Tanker Planning Example

This appendix gives a more detailed description of the major IPA planning example studied by PLANIT and reports on the progress made since the main paper was written.

The example is a project to design and construct a liquified gas road tanker. The project involves the fabrication of the tank vessel, the procurement of auxiliary equipment such as pumps and the assembly of the vessel sub-system to the chassis to produce the complete vehicle. This example was selected as it includes all three areas of planning activity within PLANIT, namely, project planning, process planning and wor shop scheduling. In addition the network for this project has been used and analysed for many years as a training example and is well understood.

Project planning requirements include producing a logic network for the whole project, analysing the network using conventional critical path methods and applying heuristic rules such as the degree of parallelism to judge the resilience and robustness of the network. Process planning includes selecting from a number of alternative processes for the fabrication of specific components. For example, the dished ends of the tank vessel could be pressed or spun. Spinning involves lower tooling costs than pressing but takes longer to produce the components. Workshop scheduling assumes that other projects exist in addition to this road tanker project, some of which have higher priorities in the workshop. A range of machine types and sizes are available having varying workloads, costs and speeds to consider.

The aim of the project is to produce the end-product, a road tanker, within the target timescale and budget. The IPA is used to modify the network logic, select particular fabrication processes, reduce timescales by increasing resources and reduce waiting time by re-allocating workshop resources to achieve an improved project plan.

A.2 The IPA System Structure

The following diagram illustrates the main system modules. Initially the generic entities and links are entered through the Model Builder into the Domain partition of the knowledge base. Entities include actions, resources and products. Links are the temporal and causal relationships between entities. Generic items describe the standard repertoire of a particular organisation from which a selection may be made to carry out a particular project.

The requirements of the project are entered in a similar fashion into the Task partition of the knowledge base. The IPA performs checks to ensure that task parameters are consistent with domain specifications eg. that the project manager for the task is authorised within the domain partition to manage a project of the size and value specified in the task partition.

A plan for the project can be constructed using the model builder by instantiating generic items from the domain partition of the knowledge base and storing them in the Plan partition. All generic items are held as frames and the Frame Editor is used to add or amend slots and slot values as required. The Link Builder is used to create links between actions in the plan. The IPA validates the proposed links by examining the generic links in the domain partition. If the user has tried to create a link which is temporally or causally unsound, then an error message is generated.

Finally the plan can be Analysed to determine critical paths, the network duration, cost and other features. If the results are not consistent