

Integrating AI Planning techniques with Workflow Management System

MD R-Moreno¹, Paul Kearney²

¹Departamento de Automática. Universidad de Alcalá. (Madrid). Spain
mdolores@aut.uah.es

²BTexact Technologies, Aadastral Park, Martlesham Heath, Ipswich,. UK.
paul.3.kearney@bt.com

Abstract

There is a variety of applications that can benefit from the ability to find optimal or good solutions to a proposed problem automatically. The Artificial Intelligent (AI) community has been actively involved in efficient problem-solving in complex domains such as military or spacecraft problems with successful results. In this paper we describe the integration of AI planning techniques with an existing Workflow Management System. We show how these techniques can improve overall system functionality and can help to automate the definition of business processes. The work is based on a short study carried out at BT Research Laboratories as part of a larger programme that aims to provide technologies for a new generation of business support systems.

Keyword: AI planning, Workflow Systems

1. Introduction

Every organisation tries to shape its processes to optimally suit the market and offer the best service to the customer. When an organisation is analysed with the purpose of identifying possibilities for optimising its routines and procedures, three basic facets are outlined:

- A task or activity describes what should be done.
- An organisation model describes who should do something.
- An information model describes which information is needed to perform an activity.

From an historical perspective, the first issue that companies focused on was the design of organisational units. In the years to come, control logic (when should something be done) is set to play a central role in connection with optimisation of business processes.

Numerous issues need to be considered when designing business processes [6,18] and implementing them in IT systems. These include: reusability of past processes, accessibility from the different agents, consistency of usage, and selection of the right model.

In recent years, a new class of software infrastructure product to support business processes has emerged: Workflow Management Systems (WfMS) [4,17,25]. A

WfMS can provide active support to a business process by controlling the routing of work around the organisation automatically. This is done based on input describing the flow, the decisions, the exceptions, the resource to be used, etc. It co-ordinates user and system participants, together with the appropriate data resources, which may be accessible directly by the system or off-line to achieve defined goals by set deadlines. The co-ordination involves passing tasks to participants' agents in correct sequence, and ensuring that all complete their tasks successfully. In case of exceptions, actions to resolve the problem can be triggered, or human operators alerted.

Prior to WfMS, many enterprises created special-purpose bespoke applications to support their processes. The advantage of WfMS-based solutions is that the workflow representation is explicit, and separated from the application code. This means that a WfMS can be customised quickly to support a new business or process, and that workflows are relatively easy to modify should a process change. Current WfMS do not address all aspects of the problem, however. Specifically, they do not deal with scheduling or resource management/allocation. Similarly, while they provide means of generating exception events when things go wrong they do not have a built-in re-

planning function. They do, however provide interfaces so that application-specific modules performing these functions can be integrated.

Recently, there has been considerable interest in the application of Artificial Intelligence techniques to Workflow Management systems. The lack of maturity that the area of Workflow Management presents due to its short history can be addressed by introducing techniques from other fields. Some researchers have seen the advantages of the integration of this approach, as shown by the existence of a Technical Co-ordination Unit of the European research network on planning and scheduling, PLANET [28], on applications of planning and scheduling to workflow. This has led to some exploratory work reflected in a roadmap and some published papers [16,21,23,26]. Although the MILOS project [7] of the Artificial Intelligence Group at the University of Kaiserslautern and the Software Process Support Group at the University of Calgary or the AI group at Edinburgh University in the TBPM project [15,31] have addressed the problem, to date very few tools have been developed using these ideas [24].

In this paper we highlight the improvements that a legacy system can gain by incorporating AI planning techniques into its day-to-day operation. We first introduce the phases that both systems have in common. After this, COSMOSS, a purpose-built legacy workflow application in use at BT is described. Then we review contingent planners, an AI technology that addresses issues found in the COSMOSS application. After this, the similarities between both workflow management and planning are presented. We conclude with an example, based on a COSMOSS scenario that illustrates how ideas from the two fields may be merged.

2. Phases

To provide a frame of reference, we identify four stages in Workflow Systems as in [26], although some authors only identify three since the Monitoring phase is included in the Enactment phase [14,30]:

- **Process Modelling:** is the stage where the user designs, models, optimises, simulates the organisation's processes. We include in this stage design of the process templates that can be instantiated and enacted by a workflow system.
- **Process Planning:** is the stage where the activities required to achieve a user goals are instantiated in a determined order, resources assigned, and preliminary scheduling performed.
- **Enactment/Execution:** in this stage the agents (software and humans) carry out the activities, with the workflow system co-ordinating execution.
- **Monitoring:** this is conducted concurrently with Enactment/Execution. The system enacting the workflow is monitored, with status information

being made available to human operators. Exceptions, such as deviation from the plan, and subsidiary processes initiated to rectify problems.

In AI Planning systems the following phases can be identified:

- **Domain modelling:** in this phase the user introduce the knowledge to the system, that is, the operators, the initial conditions and goals. Each planner has its own syntax although lately it is been an effort to unify the syntax in a unique language: the Planning Domain Definition Language 2.1 (PDDL2.1) [22].
- **Process Planning:** the plan is outlined as a set of instantiated actions in a determined order. Commonly, plans do not contain information about resources, so in some problems planning and scheduling can be separated. In other cases, this idea has to be abandoned and mechanisms to treat resources through constraining equations must be integrated to solve the problem as in O-Plan [8].
- **Execution:** this stage is concern with the actions' execution.
- **Monitoring:** the results of the actions execution can differ from the actions expected results so monitoring must take place to anticipate events or re-plan if the initial plan can not be achieved.

3. A Legacy Workflow System

COSMOSS (Customer Orientated System for the Management Of Special Services) [5] provides support for progressing orders concerning provision of private lines. It was built at the beginning of 90's and it handles about a dozen main product families divided into digital and analogue categories.

The business processes start with a customer contacting a call centre to place an order. The representative in the call centre gathers information from the caller about the customer and the service required in the form of a Service Order. This is passed automatically to COSMOSS for completion where it becomes a job. The job is decomposed into activities by matching against a set of job templates. Target times are then derived for these activities based on information stored in the job and activity templates and the customer requirements. These activities, with target start and completion dates, are passed to other OSSs¹ where they are allocated and enacted. Progress is reported back to COSMOSS.

The main COSMOSS modules are:

- **Order taking module.**
- **Product Register.** This interface to the portfolio database - PDB – which holds information on 90% of BT's product range. This is also used by other systems.

¹ Operational Support Systems

- Customer and site database. This holds information on customers and their premises, and is basically specific to the COSMOSS system.
- An engineering database.
- Job Management module.

All the information that COSMOSS can handle is organised in templates. Each time new products are introduced, new templates are created. The *Process Modelling* stage corresponds to design and creation of the Job and Activity Templates. In AI Planning systems, each activity template corresponds to an operator.

A service order (with its parameters) is used to create a job, which is decomposed into activities linked by dependencies.

Activity templates may have conditions to be met for them to complete. Activities also have 'input criteria': these are similar to pre-conditions. They can be used to prune branches of the process tree that are no longer potentially relevant. When a process is initially instantiated, it generally is under-specified, including alternative branches, only, some of which will be used.

Conditional processing controls which activities actually become part of the job by prompting the users with questions that they must answer. Different answers will cause different activities to be created in the job. Sometimes the same question may be repeated in some activities; the reason to repeat the questions is to avoid redoing all the design if the user answers the question incorrectly by an oversight. However, the system allows an Automatic conditional response, that is, the system will assume that the question is correct and will jump it.

This corresponds to the *Process Planning* phase, that is, the appropriate template is identified, instantiated, and the instance is elaborated in sufficient detail to be executed. For Process Planning in the AI sense, we obtain an instantiated plan without having in mind temporal/resources constraints, just activities linked by dependencies. As we describe below, the resource assignment is done separately.

After template selection, the software constructs a schedule for a job by trying to meet either the Customer Requirement by Date (CRD) or the Target Completion Date (TCD), whichever is the later. COSMOSS uses a common algorithm – critical path analysis to apply date rules to a template to adjust the overall dates of the order and the window lengths of the activities within the template, to ensure the Job is completed by the CRD. It is worth noting that when the job is created it will be scheduled according to the content of the whole template. Once the job and the corresponding activities have been selected, they are assigned to 'owners' (or rather queues) for its completion. Usually an owner is an organisational unit, but it could be a queue for another system.

At this point, the activities are assigned to agents (bearing in mind the computer programs and the human and material resources available in the system as well as the roles they can hold), and the *Execution phase* begins. During execution, completion of activities, and delays and other problems must be detected and reported; this is the *Monitoring phase*. In COSMOSS this information is sent to a human manager in the customer service centre. This manager can take the appropriate measures to rectify the situation, or at least to try to ensure that the situation does not recur: in some cases, the problem can be resolved adding some templates. In other more drastic cases, a new job template must be required (that is, a new plan).

Readers knowledgeable about AI planning will recognise many of the issues addressed here, though the terminology may be unfamiliar. The designers of COSMOSS developed an *ad hoc* and domain specific solution without knowledge of the great body of planning research that might have enabled a more elegant and manageable solution. As an example of the improvements that could have been achieved, we consider how AI planners can help to avoid the 'oversight' problems in the template answers and speed up all the design templates.

4. Contingent Planners

To automate COSMOSS job template design we need a planner that could have different outputs depending on the action that is required to complete the service.

Most classical planners use the assumption that there is no uncertainty in the world: every action has a predictable output. A contingent plan is a plan that contains actions that may or may not actually be executed, depending on the circumstances that hold at the time. A contingent planner must be able to produce plans even if the initial conditions and the outcomes of some of the actions are not known. Several contingent planners can be used to automate Activity Template selection in a Job Template: Cassandra [29] is a partial-order, contingency, domain-independent problem solver architecture based on UCPOP [13]. SGP[32] is an extension of the Planning Graph Analysis algorithm Graphplan [2] to support uncertainty in initials conditions and actions. CNLP [27] uses the basic SNLP algorithm to construct contingents plans, PLINTH [12] is a total order plan very similar to CNLP in its treatment of contingency plans. SENS_p [10] like Cassandra is based on UCPOP but differs in the way it represents uncertainty. All have in common the way they represent operators, based on STRIPS representation. An operator consists of preconditions (the conditions that have to be true to apply the operator) and the *post*-conditions, with add-list (the new conditions after applying the operator) and delete-list (the conditions that are not any more true). These conditions are represented by a set of logical formulae,

the conjunction of which is intended to describe the given state. Another type of planner that we considered was the probabilistic contingency planner exemplified by C-BURIDAN [9]. However, the fact that those planners are based on a probabilistic model make them unsuitable for use with COSMOSS.

For the purpose of this study we have used the planner Cassandra, but any of the others mentioned could be used. To construct plans in Cassandra, all the possible uncertainty outcomes of actions must be known *a priori*, that is, the planner must be able to enumerate these contingencies. Each single operator may introduce any number of sources of uncertainty with mutually exclusive outcomes. Every source of uncertainty is a decision to make. Figure 1 shows an example of a syntax operator of the Cassandra planner [29] corresponding to the example introduced in the last section.

```
Action (InstallingL ?line)
Preconds:(:and (required-service ?client)
              (not-occupied ?worker))
Effects:(:when (:unknown ?spare-available Yes)
              ;uncertain effect
              :effect (connect ?line))
         (:when (:unknown ?spare-available No)
              ;uncertain effect
              :effect (built ?line))
```

Figure 1. Representing the action of installing a line

As explained in the last section, each Activity in COSMOSS may have conditions that define possible user answers. All the possible answers are known at design time. These answers will cause different Activity templates to become part of a job at execution time. This is also how Cassandra works: if a decision cannot be made in advance due to lack of information, the agent will choose which branch to follow when the information becomes available during execution of the plan.

In addition to the domain theory, one has to provide the problem description in terms of initial states and goals. Those states are represented by a logical formula that specifies a situation for which one is looking for a solution. The initial state specifies the starting situation of the posed problem. Goals are often viewed as specifications for a plan. They describe the result that successful execution of the plan should produce: what one would like to be true at the end of the solution of the problem. Figure 2 shows an example of some initial and goal conditions in Cassandra.

```
Initial:(and (required-service Mary-Th)
            (installed-telephone Mary-Th No)
            (spare-available Yes)
            (card-available No)
            (not-occupied Smith))
Goals (and (agree Mary-Th Yes)
           (installed-telephone Mary-Th Yes))
```

Figure 2. Initials and goals of installing a line

5. Points of Correspondence

To understand how a contingent planner could be integrated with COSMOSS, let us introduce the following relationships: (for a high level description of merging AI planning techniques and Workflow, go to [24]).

Inputs of the planner:

- **Domain theory:** actions are represented by so-called operators. Each operator will be used to represent each Activity in COSMOSS. The pre-conditions of the operator are the pre-conditions of the Activity, and the post-conditions or effects are the expected results after completion of the Activity. If the Activity has conditions (that will be used to prompt to the user at execution time), each of the possible answers will be used to represent each source of uncertainty in the operator.
- **Problem:** in COSMOSS the problem is to determine a process that will result in a service that will satisfy the customer's requirements. These requirements will be used to specify the initial state and goals. Actually in COSMOSS, the design of the process is done manually by a user who chooses which Activity template must be part of the Job in order to complete the service.
- **Initial state:** is determined by the information in the customer order, that is, the user location, the existence of a line that can be reused, the urgency of the work, etc.
- **Goals:** The service the user requires and that is offered by COSMOSS, e.g. the on-time installation of the telephone line.

Outputs of the planner:

The planner generates a sequence of instantiated Activity Templates that will be part of the Job. As Cassandra is derived from UCPOP, it also allows actions that can be executed in parallel.

In the next section, we present a simple example that illustrates all the concepts introduced in the preceding sections.

6. A Simple Example

A customer contacts BT customer service (over the phone, in a BT shop or via the Internet). Let us say Mary Thompson contacts BT to ask for a new telephone line. At this point the business process starts (if the user agrees to the terms and conditions). The customer details are needed to see if she is an existing customer and already has a line with BT (in that case, a discount will be applied to the second line). If the customer is asking a line for the first time, a spare pair of wires must be available from the house to make a connection from the Distribution Point (DP, e.g.: telegraph pole). If no pair is available then a new cable

must be built (and the customer must be notified that the delivery date will be delayed).

Afterwards, it is necessary to check that there is a spare line card available in the exchange (in that case it is reserved/allocated). If none is available installation must be arranged. Installation involves making the connection at the DP (connecting a drop wire to the pair of wires that lead back to the exchange).

Then, someone must:

- contact the customer to arrange a visit to the house to fit new NTE (network terminating equipment, that is, the box on the wall that the phone is plugged into);
- arrange for an engineer to turn up on the right day/time to test the line end to end and install the NTE;
- allocate a telephone number to the new line and configure the exchange;
- update the exchange, line plant and customer records;
- and of course, check with customer that he/she is happy with the service.

Figure 3 shows the plan that Cassandra builds for this particular example. The decision-steps in the plan and all possible outcomes of uncertainty are represented in same way that the authors of Cassandra use in [29]. This particular plan has three sources of uncertainty: the existence of a line, the existence of spare pair and the availability of a card spare. The operator/activities are underlined.

We start with an incomplete portion of the plan with two uncertain effects: the availability of the line or of the availability of the spare. If the line is not available, we arrive at another incomplete portion of the plan with two possible situations: the availability or non-availability of the spare. If the spare is not available we just need to build the cable. Once the cable has been built (in the case is needed) or checked the availability of the line or spare, the third incomplete portion of the plan depends on the card availability. If it is not available, the card must be installed. After this, the rest of the plan does not present any uncertain effect and the control follows a linear flow.

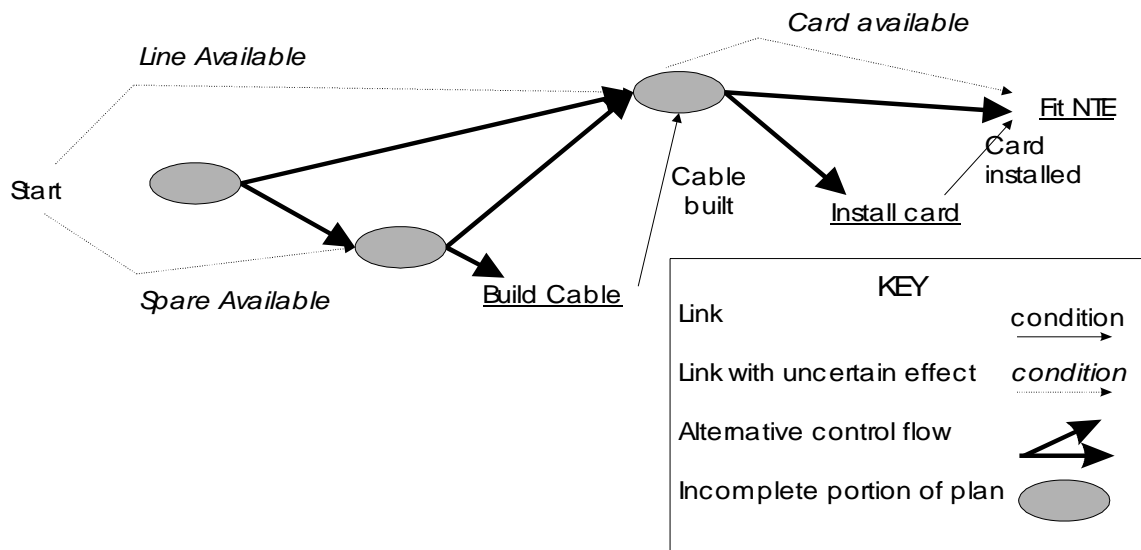


Figure 3. A partial plan showing the sources of uncertainty for providing a new line.

Each of the possible actions/operators in Cassandra will be matched to a particular Activity Template in COSMOSS. Figure 4² shows the Job template with its activities for the example of Figure 3. The code represents each possible situation (activity templates) and the arrows show the different paths depending on the user answers in each activity template.

So LIAV01 represent the template for the AAvailable LIne with two possible paths (arrows) for the two user answers: Yes or No. Each answer will add different activity templates to the job template. If the answer is Yes, the AAvailable CArd template will be added or AAvailable SPare template in the other case.

² The template shown in the Figure has been created for this particular example, and does not exist as such in COSMOSS

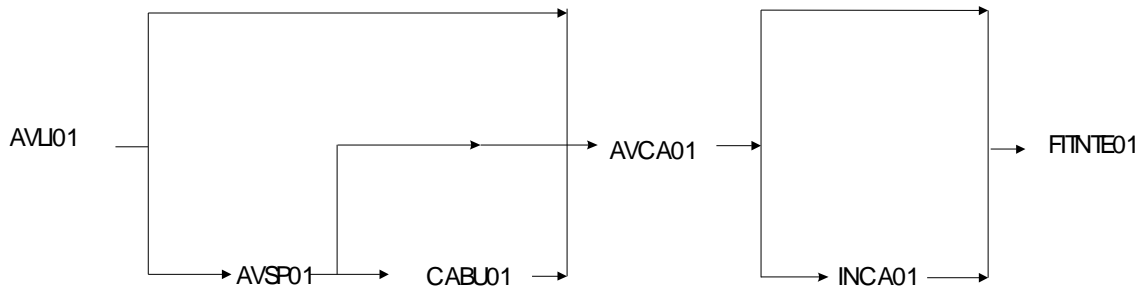


Figure 4. Partial Job Template corresponding to the partial plan of Figure 3.

If we consider the problem of Figure 2 Cassandra would generate the following plan:

```

InstallingL line501 - InstallingC
card327 - FitNTE Mary-Thompson -
Test&Install line501 - AllocateNumber
line501 - UpdateL line501 - UpdateC
Mary-Thompson - CheckCustomerOrder
Mary-Thompson.
  
```

7. Related Work

Several research groups have integrated AI techniques with Workflow Management Systems (support the modelling and the enactment phase of the software engineering process). Some have applied planning tools in the enactment phase and very few have integrated planning techniques with BPR tools during the modelling phase.

The SWIM system [3] integrates process instantiation, task allocation, execution, monitoring and optimisation for improvement in the schedule. There is a Process Library that provides the correct process definition to the Dynamic Process Manager. New processes can also be added to this library.

The MILOS project [30] is a process modelling and enactment approach that provides project planning and management support like resource allocation and time scheduling for tasks in the project. The MILOS workflow engine allows the model and plan to be changed during project enactment and provides support for process restarts whenever necessary. The library of process models contains descriptions of best practices in software development for a given company. The project manager selects processes and methods from the library creating an initial project plan. The plan is uploaded to standard project management tools as MS-Project.

The TBPM project is based on work carried out in the Enterprise project [31] and centres around an intelligent workflow engine that includes an interactive process planner. The planner uses AI techniques based on O-Plan [8] to assist in task planning, while permitting the user to participate in planning decisions. An agent-based architecture supports the execution and

co-ordination of the planned process among multiple participants and distributes processes across computer networks. The user is able to plan a task by assembling fragments and then to refine it, generating a hierarchical model of the process to be performed. For more flexibility, the user is able to edit the process model before and during its enactment, in order to specialise it.

In all these systems the modelling phase is based on a process library but there is no automatic generation as we have outlined in this integration of COSMOSS and Cassandra. Each time new products/templates are created, there is no need to create a library; the planner will generate the correct model.

8. Conclusions

We have shown the potential of applying AI planning techniques within workflow management systems. This benefit will be realised as much by introducing workflow specialists and software engineers to planning concepts and representations as by direct application of planning software.

We also want to outline the issues that AI planners can gain with this approach. Generally, to specify the domain theory, a deep understanding of the way AI planners work and its terminology is needed. However, if we use an existing system, the description language is closer to the user (at least quite familiar for the COSMOSS user). Therefore, we try here to solve the planning domain modelling task by using BPR representation models and technology. In fact, it is really a symbiosis in the sense that, once we have defined the domain using a WfMS, AI planning and scheduling technology can help in the automatic generation of process models.

In this paper we have focussed on how the contingent planner, Cassandra, can help to automate the design of appropriate templates in a legacy system used to support the business processes at BT.

9. Future Research

Particular areas worthy of attention that can take advantage of AI techniques are: scheduling and resource allocation, monitoring [19] for (and anticipation of) deviations from plans, and re-planning in the event of such exceptions. Learning techniques [20] could also be applied, allowing e.g. optimisation of processes over time.

10. Acknowledgements

The first author wants to thank all the Intelligent Business Systems Research Group for their help and collaboration shown during the period of time at BT Adastral Park, specially Paul O'Brien, Jamie Stark and Simon Thomson. Also to Simon Martin the comments on this paper and Nigel Irons for all the support that he has given us with COSMOSS. We also want to thank Daniel Borrajo.

References

- [1] Austin Tate, James Hendler, Mark Drummond. *A Review of AI Planning Techniques*. *Knowledge Engineering Review* Volume 1, Number 2, pages 4-17, June 1985.
- [2] Blum and M. Furst, *Fast Planning Through Planning Graph Analysis*. *Artificial Intelligence*, 90:281--300 (1997).
- [3] Berry P.M., Drabble B. *SWIM: An AI-based System for Workflow Enabled Reactive Control*. In the 16th IJCAI99 Workshop on Intelligent Workflow and Process Management: The New Frontier for AI in Business. Mamdouth Ibrahim, Brian Drabble and Peter Jarvis editors
- [4] C. Mohan. *Recent Trends in Workflow Management Products, Standards and Research*. In Proc. NATO Advanced Study Institute (ASI) on Workflow Management Systems and Interoperability, Istanbul, August 1997, Springer Verlag, 1998. http://www.almaden.ibm.com/cs/exotica/exotica_apers.html
- [5] Chandler J. *Management of special services: designing for a changing world*. *BT Technological Journal*, Vol 15, No 1, January 1997.
- [6] Davenport, T. and Short, J. *The New Industrial Engineering: Information Technology and Business Process Redesign*. *Sloan Management Review*, pp 11-27. 1990.
- [7] Dellen, B. Maurer, F. and Pews, G. *Knowledge Based Techniques to Increase the Flexibility of Workflow Management*. *Data and Knowledge Engineering*, North-Holland. 1997. <http://www.wagr.informatik.uni-kl.de/~comokit>
- [8] Drabble, B., Tate A. *The use of optimistic and pessimistic resource profiles to inform search in an activity based planner*. In Proceedings of 2nd Int. Conf. AI Planning Systems, 1994.
- [9] Drapper D., Hanks S., & Weld D. *A probabilistic model of action for least-commitment planning with information gathering*. In Proceedings of the Tenth Conference on Uncertainty in Artificial Intelligence, pp 178-186 Seattle, WA, Morgan Kaufmann.
- [10] Etzioni O., Hanks S., Weld D., Draper D., Lesh N. and Williamson M. *An approach to planning with incomplete information*. In Proc 3rd International Conference on Knowledge Representation and Reasoning, pp 115-125. Boston, MA. Morgan Kaufmann.
- [11] Fikes R. E, Nilsson N. J. *STRIPS: a new approach to the application of theorem proving to problem solving*. In *Artificial Intelligence*, 2:189-208.
- [12] Goldman R.P. and Boddy M.S. *Conditional Linear Planning*. In Proc. 2nd International Conference AI Planning Systems, pp 80-85. Chicago, IL. AAAI Press.
- [13] J.S.Penberthy and D. Weld. *UCPOP: A sound, complete, partial order planner for ADL*. In Proc. 3rd Int. Conf. On Principles of Knowledge Representation and Reasoning, pages 103-114, Oct 1992.
- [14] Jarvis P, Stader J, Macintosh A, Moore J, & Chung P. *A Framework for Equipping Workflow Systems with Knowledge about Organisational Structure and Authority*. In Proceedings of the Workshop on Systems Modeling for Business Process Improvement (SMBPI-99), University of Ulster, County Antrim, Northern Ireland, UK, pp 205 - 219.
- [15] Jarvis P. et al. *What right do you have to do that?*. In ICEIS - 1st International Conference on Enterprise Information Systems. Portugal, 1999. <http://www.aiai.ed.ac.uk/project/tbpm/>
- [16] Karen L. Myers and Pauline M. Berry. *At the Boundary of Workflow and AI*. In AAAI-99 Workshop on agent-Based Systems in The Business Context. Brian Drabble and Peter Jarvis Cochairs.
- [17] Leymann. F., and Roller. D. *Business Process Management with FlowMark*. Proceedings of IEEE Comcon, March 1994.
- [18] M.Hammer and J. Champy. *Reengineering the Corporation*. Harper Business Press, New York, 1993.
- [19] Manuela Veloso, M. E. Pollack, and M. T. Cox. *Rationale-based monitoring for planning in dynamic environments*. In Proceedings of the Fourth International Conference on AI Planning Systems, 1998.
- [20] Manuela Veloso. *Learning by Analogical Reasoning in General problem solving*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, 1992.
- [21] Markus Hannebaeur. *From formal Workflow models to Intelligent Agents*. In AAAI-99 Workshop on agent-Based Systems in The

- Business Context. Brian Drabble and Peter Jarvis Cochairs.
- [22] M. Fox & L. Derek. PDDL 2.1: An extension to PDDL for expressing temporal planning domains. <http://www.dur.ac.uk/d.p.long/competition.html>
- [23] MD R-Moreno, P. Kearney and D. Meziat *A case study: using Workflow and AI planners*. Proc. of the 19th Workshop of the UK Planning & Scheduling Special Interest Group. PLANSIG2000. The open University, UK
- [24] MD R-Moreno, D. Borrajo, D. Meziat. *Process modelling and AI planning techniques: A new approach*. 2nd International Workshop on Information Integration and Web-based Applications & Services. IIWAS2000. Yogyakarta.
- [25] Medina-Mora, R. Winograd, T. and Flores P. *Action Workflow as the Enterprise Integration Technology*. Bulletin of the Technical Committee on Data Engineering, 6(2), 1993. IEEE Computer Society.
- [26] Paul Kearney and Daniel Borrajo. *An {R&D} Agenda for AI Planning applied to Workflow. Management Proceedings of the eBusiness and eWork Conference 2000.*
- [27] Peot M.A and Smith D.E. *Conditional Nonlinear Planning*. In Proc. 1st International Conference AI Planning Systems, pp189-197. College Park, Maryland. Morgan Kaufmann.
- [28] PLANET: European Network of Excellence in AI Planning. <http://planet.dfki.de/>
- [29] Pryor L, Collins G. *Planning for Contingencies: A Decision-based Approach*. Journal of Artificial Intelligence Research 4 (1996) 287-339.
- [30] Sigrid Goldmann, Jürgen Munich, Harald Holz. *Distributed process planning support with MILOS*. To appear in Int. Journal of Software Engineering and Knowledge Engineering, October 2000.
- [31] Stader J. *Results of the Enterprise Project*. Proc. 16th Int. Conference of the British Computer Society Specialist Group on Expert Systems. Cambridge, UK, 1996.
- [32] Weld D., Anderson C, Smith D.E. *Extending Graphplan to Handle Uncertainty and Sensing Actions*. AAAI98. 1998 .