A Tool Set For Enterprise Modelling

Jussi Stader

AIAI-TR-211
January 1997

Published in Proceedings of Interfaces'97, 6th International Conference, held in Montpellier, France on 28-30 May 1997. Published by EC2 & Developpement, Paris, France.

Artificial Intelligence Applications Institute
The University of Edinburgh
80 South Bridge
Edinburgh EH1 1HN
United Kingdom
Tel: +44 (131) 650 2732
Fax: +44 (131) 650 6513
email: J.Stader@ed.ac.uk

Abstract

In this paper we present a tool set for enterprise modelling developed during the Enterprise project at AIAI, the University of Edinburgh. Our approach concentrates on integration, communication, flexibility, and support. We describe the Enterprise Tool Set which uses executable process models to help users to perform their tasks. The Tool Set is implemented using an agent-based architecture to integrate off-the-shelf tools in a plug-and-play style. To ensure effective interchange of information and knowledge between different users, tasks and systems, we developed the Enterprise Ontology which defines terms used in organisations. The project has been successful and valuable insights have been gained and made available.

Keywords intelligent task management, enterprise modelling, agent-based architecture, process modelling

1 Introduction

Organisations are becoming increasingly complex. Competitive pressures require adaptation to rapidly changing markets. This in turn needs to be supported by methods and tools which help to model, analyse and improve various aspects of how a business works and how it is organised. Many such "enterprise modelling" methods rely on pencil and paper. The Enterprise project was aimed at providing an integrated set of computer-based tools to improve on this current lack of support.

In brief, the overall goal of enterprise modelling is to take an enterprise-wide view of an organisation which can then be used as a basis for taking decisions. In order to achieve, use, and maintain such an enterprise-wide view strong facilities for integration, communication, flexibility, and support are required. These can be detailed as follows.

Integration must be achieved for three purposes: to relate information for obtaining different views of the enterprise; to relate tasks to the tools that support them; and to establish connections between the tools themselves.

Communication must be achieved on different levels: between people in order to ensure that enterprise models are shared within an organisation; between tasks so that information can be used where it is relevant; and between the tools used to perform the tasks so that relevant data can be passed between them.

Flexibility is important to allow an organisation to adapt to changes in its environment, its processes, and the availability of tools. It is also important to allow flexibility in the enactment of processes to ensure that people's time is used as effectively as possible, giving people the choice of what to do and when to do it.

Support must be provided to take care of technical details and to ensure that the given flexibility does not result in confusion and that processes are carried out effectively.

This paper describes the Enterprise Tool Set developed by AIAI under the Enterprise project. The Tool Set helps to capture aspects of a business and to analyse these aspects in order to identify and compare options for meeting the business requirements. The Tool Set provides task management support to users by helping them perform enterprise modelling activities and guiding them through the Tool Set facilities. We have concentrated not so much on the detailed modelling of different aspects of organisations, but on the integration of different views and the support for enactment.

The Enterprise project was the UK government's major initiative to promote the use of knowledge-based systems in enterprise modelling, supported under the DTI's Intelligent Systems Integration Programme, Project Number 8032. AIAI at The University of Edinburgh lead the project and the partners were IBM, Lloyd's Register, Logica and Unilever.

This paper describes the approaches taken in the Enterprise project and the workings of the Enterprise Tool Set that was implemented. The paper concludes with the main technological advances achieved and insights gained during the course of the project. For more details on this project's goals and on enterprise modelling see [2].

2 Approach

We developed the Enterprise Tool Set to provide suitable support for obtaining an enterprise-wide view of an organisation. As a basic model of support for enterprise modelling, we decided to use process models. These provide a process-oriented view which can be enacted in a running system. We developed a Procedure Builder which supports the capture of process models.

In most organisations there is a variety of tools in use. We decided to support the integration of available tools with as little change to the tools as possible, rather than trying to replicate existing tools and their interfaces. The framework for this is an agent-based architecture together with a library to support the addition of tools to the system, the Agent Toolkit.

As well as providing support for integrating tools, we provide support for enactment of processes. The Task Manager integrates tools with process models and provides an agenda-style support for enactment.

In order to achieve this high level of integration and to allow all components to communicate effectively, there must be an agreement about the way in which terms are used. We have developed an Enterprise Ontology for this purpose.

In summary, the components of the Enterprise Tool Set are:

- the Agent Toolkit for agent development,
- the Procedure Builder for capturing process models,
- the Task Manager for integration, visualisation, and enactment support, and
- the Enterprise Ontology for communication,

These components combine with a flexible set of integrated, off-the-shelf tools in the working system.

2.1 Agent Toolkit

We investigated a variety of externally available agent-based architecture solutions and concluded that none of them was mature enough to use as-is or met all our requirements. We therefore developed our own agent-based solution which is supported by the Agent Toolkit. One of the essential principles guiding its design was to make the creation of new agents as easy as possible. We wanted to be able to support as-yet unspecified tools as agents without any redesign of the Agent Toolkit or any other component of the Enterprise Tool Set in order to accommodate these new tools.

Keeping in line with emerging standards, the agent communication language we support is KQML (Knowledge Query Manipulation Language, [1]) and messages are expressed using KIF (Knowledge Interchange Format, [3]) and the Enterprise Ontology (see below).

There is also an Agent Registration Tool to make agents (software or human) available to the Enterprise Tool Set, specifying details about the agent, such as its capabilities. The agent-based architecture allows agents to be added to and removed from the overall Tool Set in a plug-and-play style.

2.2 Procedure Builder

The Procedure Builder is a graphical tool for describing and recording business process models. It allows the user to build process diagrams using the notation of Process Flow Networks (PFNs) as specified in the IDEF3 Process Description Capture Method [5]. The Procedure Builder also allows users to specify information which is specific to the needs of task management, in particular the capabilities required to perform a process and input and output specifications of the process. To represent the process information we developed a process modelling language. In developing the language we took account of emerging standards, such as PIF (Process Interchange Format, [4]) and WAPI (the Workflow Application Programmer's Interface developed by the Workflow Management Coalition).

The output from the Procedure Builder can be exported for use directly by the Task Manager. In addition the Procedure Builder can produce reports containing the process diagrams and associated process information.

2.3 Task Manager

The Task Manager is the interface between the user and the Enterprise Tool Set. It directly supports the user in performing their current tasks by helping them to follow the tasks' corresponding processes captured with the help of the Procedure Builder.

The Task Manager plans user tasks and the use of agents. Information about tasks is available from the models specified with the help of the Procedure Builder, and information about agents becomes available when agents are registered. Appropriate agents are identified by matching a task's required capabilities (from the process models) against the agents' registered capabilities. This is done at run-time, so that the most suitable agent can be identified, taking into account which agents are available at that time. The Task Manager also handles outputs generated by tasks. It stores such output information and it can pass it on to other tasks that require the information.

The Task Manager monitors the progress of a task's enactment, keeping track of which tasks are currently active, which have been done, etc. This progress can be visualised in different ways, the visualisation being supported by the process diagrams captured with the help of the Procedure Builder.

The amount of support that the Task Manager gives to the user is flexible and can be adjusted by the user. Advice on what to do next is available on the basis of which tasks are ready to be executed and which have recently been completed. The Task Manager can also help the user to recover from failures, determining alternative routes of action.

The Task Manager effectively puts an extra layer of control on top of the agent services. It coordinates the use of agents at the level of the user's tasks and lets the user participate in this coordination, according to the tasks in which the user is engaged.

2.4 Ontology

We require an effective communication mechanism to achieve integration of a wide variety of tools, both new and old. Independently developed tools are likely to use different terminology which can lead to conflicts and ambiguity. To resolve such issues, a standard terminology is required when tools are integrated. An ontology provides such a standard terminology. For an introduction to the field of ontologies and their development and use see [8].

We developed the Enterprise Ontology for use in the Enterprise Tool Set. It is a set of terms frequently used in enterprises, each carefully defined to conform as best as possible to common usage. We have concentrated on the areas of organisation, strategy, activities and processes, and marketing. We did not try to define all terms that are likely to be required; the terms we defined are ones that we expect to be generally useful. Every organisation will have their own set of terms that they use, so the ontology can be extended to suit the specific needs of the organisation. The Enterprise Ontology is available on the world wide web [7].

Committing to this ontology has the advantage that terms are used consistently and unambiguously throughout the enterprise. The ontology thus provides the basis for communication between agents, whether they are human or software agents.

3 Working with the Tool Set

There are several phases of working with the Enterprise Tool Set. Apart from initial specifications these phases are relatively independent of each other and we expect that these phases will involve different users. The phases are:

- extend the ontology to include concepts specific to the application,
- transform tools into agents,
- register agents with the Tool Set,
- generate process models,
- make process models available to the Tool Set,
- enact process models.

The general architecture of the Enterprise Tool Set which supports these phases is shown in figure 1. In the figure, Tool Set components are shown as rectangles, information like models and data is shown in diamonds and users are shown as ovals. Arrows show information flow.

Tools are turned into agents by agent programmers. This can be done with the help of the Agent Toolkit (not shown in the diagram). Tools that have been turned into agents can be registered with the Enterprise Tool Set by the tool set administrator using the Agent Registration Tool. Registration information includes the agent's type (software or person) and its capabilities. Once an agent has been registered, its capabilities can be called upon by the Task Manager for the enactment of tasks.

Processes are captured by the method expert using the Procedure Builder. Again this is done using the ontology. The processes can be loaded into the Task Manager and are then used as a basis for supporting users and for coordinating the use of agents.

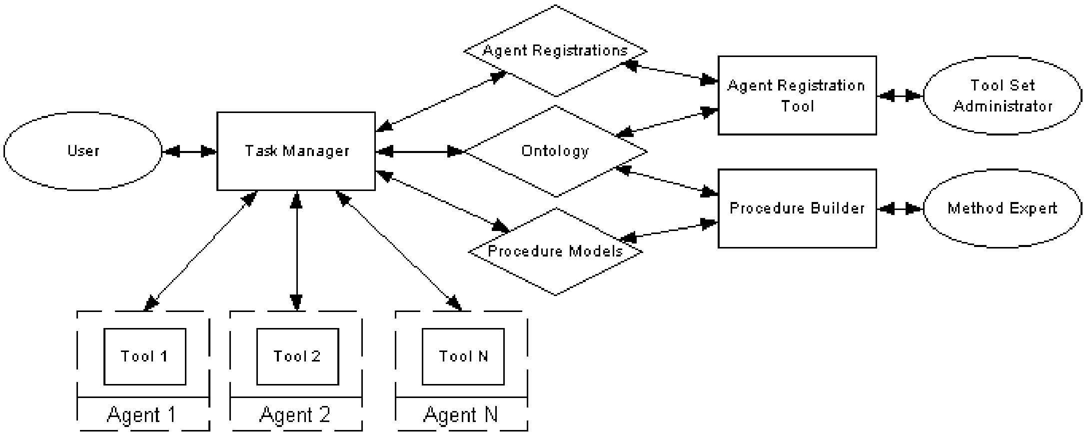

Figure 1: Enterprise Tool Set Architecture

The Task Manager is the interface between the Tool Set and the user. It is the heart of the system and provides the main mechanism for integration and support. It makes use of all other development and modelling efforts, like process capture, agent programming, and ontology development. On the task level, all communication goes via the Task Manager. However, during task execution, an agent may "take over" and use its own user interface to communicate with the user.

The first five phases can be seen as support phases for the last one, but they have their own benefit, independently of their use for the Tool Set. For example, generating process models is a useful exercise in itself to gain insights into structures and processes, and it often shows scope for improvements.

In the rest of this section we consider the last phase (enacting process models) and we concentrate on the Task Manager, as it is the driving force of the Enterprise Tool Set. For this we will assume that the Tool Set is ready to be used, i.e. the system includes results from the first five phases above.

During nearly all interactions with the system the user is in direct control. The level of support can be set such that the Task Manager takes more or less initiative, but in general the user can choose freely what to do. Choices include working on tasks, browsing models or other information, leaving a task in mid-flow and starting another task, going back to a previous task, etc.

Initially, the user identifies tasks that are to be performed during the session (today's work). To identify tasks for which the Task Manager can provide support, the user can browse the process models which are currently available in the Task Manager (from the Procedure Builder).

When a high-level task is identified for execution, the Task Manager plans this task: it first breaks the task down into its full hierarchy of decomposition (sub-tasks). Then it goes through each of these sub-tasks to identify agents that can perform it, checking the agents' advertised capabilities against the task's required capabilities. Agents are then ranked according to how well they can perform the task and the best agent is selected. If there are tasks for which no suitable agent can be identified, this is noted but it does not stop the Task Manager from progressing.

The next planning step is to determine at which level to execute tasks in the hierarchy. If a task is broken down into sub-tasks, the Task Manager can either support the execution of the task itself, or it can help the user to step through the sub-tasks. The Task Manager bases its decision on the availability of agents to perform the tasks and sub-tasks. The level that is best supported by agents will be chosen, and high levels are preferred to low levels.

When the Task Manager has selected agents and levels of execution, it is ready to start running. The Task Manager goes through all tasks to be done and identifies the ones that are ready to execute. A task is ready if its required inputs are available and all tasks that are to be done before it have been completed. The Task Manager then offers these tasks to the user for selection. When the user has selected a task, the Task Manager checks whether an agent has been identified to perform the task and whether that agent is currently available. If there is a problem it again tries to find a suitable agent. If it cannot, it will ask the user which of the currently available agents should be used.

The Task Manager then gathers the information required by the task from its store of information. It asks the agent to start the task, passing it the inputs along with the request. The user may be required to interact with the agent in order for the task to be performed. However, it is up to the agent to organise this interaction through its own user interface.

The Task Manager does not have any information about how long the task will take (seconds, days, months...) so it does not wait for a reply from the agent. It can ask the agent for a progress report if the user requests it, or it can wait for the agent to tell it that it has finished. When the task has been performed successfully, the Task Manager obtains the task's output information from the agent and adds it to its store of information. The task is then noted as "done" and the Task Manager re-evaluates all relevant tasks as to whether they are ready to be executed. When these new doable tasks are offered to the user for selection, any successors of the task just finished will be shown before other doable tasks so that users do not lose their thread.

When a task fails, the Task Manager gives the user the option to re-start the task. Failing that, it looks at the process model to find alternative courses of action to offer to the user (e.g. loading data from a file *vs* entering it by hand).

This process of identifying executable tasks and managing their execution is repeated until no more executable tasks can be found or until the user stops the process. At that point the user receives a report on the overall progress of the high-level task, which can be a note that the task is done, or information about which tasks failed and which are still to be done.

4 Conclusion

The approach taken during the Enterprise project is proving to be appropriate. The Enterprise Tool Set has been implemented and a demonstrator has been built to illustrate the advances made.

The advances in the four areas of work are:

Procedure Builder The Procedure Builder is an advance on earlier software for capturing processes, in that it is able to capture and export models in a form that can be enacted.

Agent Toolkit Like most agent infrastructures, the Enterprise one provides services for communication between software applications. Enterprise goes further than most, in that it provides an Agent Toolkit which makes it easier to add agents.

Task Manager The Task Manager provides the overall integration between process models and tools, passing information between tasks. It is unlike other agent systems in that it puts an extra layer of control on top of the agent services. This lets the user participate in the coordination of agents, according to the tasks in which the user is engaged. It thus provides the user with a better chance to make effective use of the flexibility provided by agent-based technology. The Task Manager is different from most workflow support systems in that it provides flexible and intelligent support on the level of the user's tasks and intentions rather than passing through set sequences of events.

Enterprise Ontology Developing the Enterprise Ontology was not easy. However, we are pleased with the final result and during the project we developed a method for building ontologies (see [8, 6]). The Enterprise ontology is distinct from other ontologies for enterprise information in that it attempts broad coverage to include most terms important to enterprises rather than limited areas. It also exists in the form of a comprehensive, carefully prepared natural language glossary *and* in a formal language. Most other ontologies must be gleaned from various scattered papers, or exist mainly in formal languages and are thus inaccessible to non-technical readers.

The major strength of the overall Enterprise Tool Set is that a process model can be built of any business activity which can then be used without modification to enact the process. The steps in the process are linked directly into the tools required to perform them. This enables the most suitable tools currently available to support the business process. In addition, the tool set provides a visualisation of the process illustrating progress as the process is enacted.

4.1 Future Extensions

Discussions during the first phases of the project and the designs of the Tool Set addressed many important and interesting issues which we had to disregard during the implementation as they were outside the scope of the Enterprise Tool Set. Other ideas emerged later on during the project after we had made some progress. We intend to go back to all these ideas and investigate them further work. Examples of such ideas are:

- A central repository in which enterprise information is kept which would make it possible to generate multiple views (visualisations) of information, illustrating different aspects of the information.
- Translation, both between different ontologies and between different agent communication languages, would provide more flexibility for agent communication and would reduce the demands made of the agent programmers.
- The Enterprise Tool Set has many aspects that are suitable for adding coordination technologies to support groups of people who contribute to a common process. For this, the current approach to task support can be extended to support multiple users.

References

[1] T. Finin, Don McKay, and Rich Fritzson (Editors). An overview of KQML: A Knowledge Query and Manipulation Language. Technical report, The KQML Advisory Group, March 1992.

[2] J. Fraser. Managing Change through Enterprise Models. In R. Milne and A. Montgomery, editors, *Applications and Innovations in Expert Systems II*. SGES Publications, December 1994.

[3] M.R. Genesereth and R.E. Fikes. KIF knowledge interchange format version 3.0 reference manual (logic-92-1). Technical report, Logic Group, Computer Science Department, Stanford University, California, 1992.

[4] J. Lee, M. Gruninger, Y. Jin, T. Malone, A. Tate, G. Yost, and other members of the PIF Working Group. The PIF Process Interchange Format and Framework Version 1.1. Technical report, MIT Center for Coordination Science Working Paper 194, Cambridge, MA, 1996. Also available from AIAI as AIAI-TR-198.

[5] R. Mayer, Cullinane T., P. deWitte, W. Knappenberger, B. Perakath, and S. Wells. IICE IDEF3 process description capture method report (al/tr-1992-0057). Technical report, Air Force Systems Command, Wright-Patterson Air Force Base, Ohio, 1992.

[6] M. Uschold and M. King. Towards a Methodology for Building Ontologies. Technical Report AIAI-TR-183, AI Applications Institute, 1995. Presented at the Workshop on Basic Ontological Issues in Knowledge Sharing Held in conjunction with IJCAI-95 in Montreal.

[7] M. Uschold, M. King, S. Moralee, and Y. Zorgios. The Enterprise Ontology. Available on World Wide Web from URL http://www-ksl-svc.stanford.edu/.

[8] M. Uschold and Gruninger M. Ontologies: Principles, methods and applications. *Knowledge Engineering Review*, 11(2), 1996. Also available from AIAI as AIAI-TR-191.