Active Models in Business

R.M. Greenwood, I. Robertson, R.A. Snowdon[1], B.C. Warboys
Informatics Process Group
Department of Computer Science
University of Manchester, M13 9PL
{markg,ir,bobs,brian}@cs.man.ac.uk

Abstract

Many emerging technologies, such as Computer Supported Co-operative Working (CSCW), Workflow systems, and virtual organizations envision the IT infrastructure of the future co-ordinating people and systems in addition to the traditional IT roles of data manipulation and information management.

Central to this view are models of the business process which help relate the process to the supporting IT. Until recently, these models have been passive, helping understanding and analysis only. It is now possible for such models to become much more pro-active in the business process.

Experimentation with active models has been underway for a number of years in the domain of software engineering. These are models having knowledge of the real-time state of the process, and through which users interact with the supporting IT. They can provide a crucial bridge between business process activity and its IT support: managing the complexity of interactions, giving people what they need, when they need it, extending the life of legacy systems and providing a framework for evolution.

This paper describes ongoing research on developing such process models and their use in providing a co-ordinating layer in the IT infrastructure. In particular it advocates 'role' based models where the actions of users (both staff and managers) are placed in the context of their responsibilities in a process.

1 Introduction

> "There are three chief problems of organization engineering: how to educate and train the members of the organization so that each can give the most he is capable of; secondly, how to give to each the fullest opportunity for contribution; thirdly, how to unify the various contributions, that is, the problem of coordination, confessedly the crux of the business organization." (Mary Parker Follett 1927 in [1] page 150)

Information Technology (IT) has made profound changes on organizations in since 1927 but the "three chief problems" remain: organizations need to educate and train their people, they need to provide each with appropriate tools for their contribution, and they must ensure that these individual contributions are co-ordinated. In general, IT has until very recently focussed on the second problem. The future will see a phenomenal growth in the use of IT to address the third problem, co-ordiantion. The dramatic growth of networks already evidences that the basic technology to allow people to communicate through IT is available now. It is expected

1. Also at ProcessWise Portfolio Center, ICL, West Avenue, Kidsgrove, Stoke-on-Trent.

to become dramatically cheaper over the next decade. In addition, organizations are increasingly viewing their processes, that is, how individual work is co-ordinated, as assets to be actively managed and improved.

In section 2 we discuss how IT is now being used as a co-ordination technology. In providing this co-ordination, IT can be viewed as an efficient butler, and must be provided with the information of how it is expected to participate in a process. This can be done by developing an *active* model of the process; a model which represents a view of the current state of the process. We discuss models in general in section 3, and active models in particular in section 4. In section 5 we highlight some experience of this approach from the software engineering domain, which highlights the need for effective methodologies to guide practitioners in this area. In section 6 we briefly overview our ongoing contribution which places active models within the overall context of process engineering. Finally, our conclusions are presented in section 7.

2 Beyond Information Technology - Co-ordination Technology

Traditionally Information Technology has been applied to support individual activity. An insurance clerk queries a database to see if a customer's request refers to work already in progress. An actuary uses a spreadsheet to calculate the premium. A retail store manager queries sales data and uses the results in entering orders for new stock. The basic paradigm is for the computer to perform a data manipulation, or calculation, at the request of an individual. These individual activities however are being performed in the context of business processes, and these processes are performed not by individuals acting alone but by individuals acting in groups. For example, the information obtained by the insurance clerk must be passed to the actuary so that the correct data is used in premium calculation. A store manager and section manager might agree not to reorder when an item is sold out to give space for new stock. The individual participants in these processes not only need to do the activities for which they are responsible; they must also communicate and co-ordinate with others.

Individuals at their computers are becoming more networked than ever before. The Management in the 1990s research programme [2] identified that 'new IT' was radically changing both the economics and functionality of co-ordination. "IT will continue to change over the next decade at an annual rate of at least 20 to 30 percent. This will lead to greater shrinkage in time and distance effects, greater interconnectedness, and better organizational memory with greater capture of organization's 'rules' (heuristics)."[2] It is therefore not surprising that many emerging technologies exploit IT as a way of connecting people. Electronic mail, Computer Supported Cooperative Working (CSCW), and Workflow all focus on the benefits of providing IT to a group rather than to individuals.

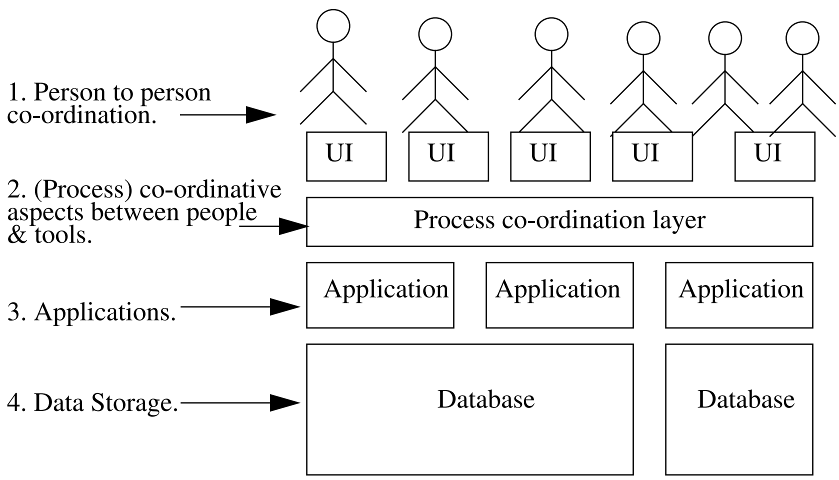
How did groups manage to function when their IT was focussed on the individual's activity? Individuals would organize their own activities, and co-ordinate directly with one another. The insurance clerk who regularly queried three distinct databases would know how the results of the first query became parameters in future ones. The retail store staff would know who to ring up when a delivery does not include all the expected items. The knowledge of who to talk to and how information is passed between computer applications is part of the organizational memory. This knowledge may be supported by work instructions, procedure manuals, and other documentation or just ‘known’. However, the organization is only effective if, in the majority of cases, people know what to do rather than how to find out what to do.

In developing IT to support groups some of this communication knowledge is embedded in the software. Workflow systems are a good example. The general model for these is to replace a form, such as an overseas travel application, which flows from in-tray to in-tray around the organization, with an electronic form which flows from mail-box to mail-box. The individual no longer has to place the form in an envelope and know its next destination, the Workflow system knows the ‘procedure’ which the form is following and can route it to its next destination.

In developing IT systems which are intimately tied with the organization’s business processes, we are effectively moving away from IT systems as calculators to IT systems as partners. If we imagine a future where a large amount of work is performed by individuals isolated at their computer terminals, then increasingly business rules will be defined by the capabilities of the IT infrastructure. Our view is to treat ‘adding business rules’ as providing a new architectural component in a business’ IT infrastructure (see figure 1).

2.1 Co-ordination Layer

FIGURE 1.



A process co-ordination layer will provide facilities which include:

- co-ordination between people - once the actuary has calculated a premium, the insurance clerk will be informed so that a quote can be sent,
- co-ordination between people and applications - when a new request for insurance arrives, the insurance clerk handling it is prompted to query the ‘work-in-progress’ database,
- co-ordination between applications - when the actuary requests a credit rating first an internal database is queried, and if there is no match then an external credit agency service is consulted.

As businesses increasingly strive to develop business processes which exploit the rapidly decreasing cost of connectivity, we believe that a process co-ordination layer will become an increasingly significant architectural component. A key question will be how to design such a co-ordination layer. Our approach is to view it as providing an active model which defines how the IT system participates in the business processes.

3 Models

Models are representations, expressed in some modelling medium, of something of interest. A model stands in some abstraction relation with its subject. The point of creating a model is to provide a way of studying certain features of the subject. A model will thus emphasise certain properties of the subject, by including them in the model, and to suppress others, by omitting them. What is present in a model and what is omitted is a decision of the modeller and is based on the model's purpose.

3.1 Passive Models

Models have traditionally been constructed in a passive modelling medium. That is, the model, once created is independent of its subject. The subject modelled may or may not change, but the model is a representation of the subject as perceived at some point in time and will not change of its own accord. The models produced to date by most IT methodologies fall into this category. An entity-relationship diagram or dataflow diagram represents the modeller's understanding at a particular point in time. If things change then the diagram will be revised; a new model will be created.

The subject of a model may currently exist; the model being an abstract representation of part of the real world. Alternatively the subject of a model may be something which may exist in the future; the model is then an abstract representation of part of a possible future.[2] In the context of business processes we might use a model to capture the current, 'as is', process or to model a potential future, 'to be', process. In both of these cases the modeller deals with two domains: the domain of the business process, and the domain of the model. The modeller will choose the modelling medium depending on the purpose of the model. A car designer might choose a wooden model of a car for a wind tunnel experiment. Most approaches to modelling business processes use a textual or graphical language.

2. Indeed the wooden models of famous ships seen in museums illustrate that it is also possible for a model's subject to be part of the past.

FIGURE 2. A simple model of retail logistics

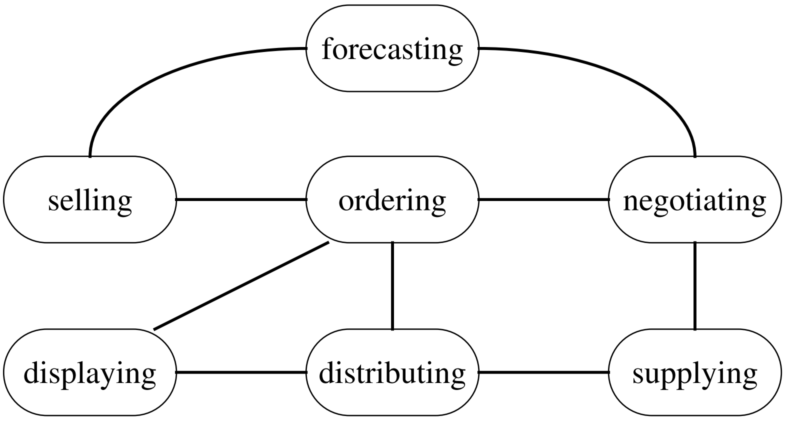


Figure 2 shows a typical high-level passive model of a business process. Its medium is a diagram made up of ellipses and lines where each line links two ellipses. It can be interpreted as an activity model: it shows a number of abstract activities which are involved in the process and connections, or dependencies, between them. Often arrows are used for the connections to indicate the output to input linking between processes.

Figure 2 can also be interpreted as representing a number of interacting roles. Each ellipse represents a responsibility for doing part of the process which is assigned to an individual, team, or division[3], and the connections represent where they interact.

Most graphical process models such as figure 2 are static. Their emphasis is on the structure of the process rather than its dynamic bahaviour. It is often possible to interpret the model to get a view of how the process works in general, for example how a typical order is handled.

3.2 Dynamic Passive Models

Many of the subject systems which we want to model are dynamic. Business processes are a good example. In some situations we want a model of such system which deals with detailed behaviour, how a specific sequence of orders is handled, and particularly how they affect each other.

There is a large discipline of computer based simulation which is based on developing mathematical models and then implementing these as computer programs. By changing parameters is possible to simulate the behaviour of the subject system in varying circumstances. These models are *dynamic* but not *active*. There is no direct link with the subject system. If the retail chain is considering the problem of warehouse location, it might develop a model of the stores, their orders, deliveries and so on. This would be a mathematical model which repre-

3. These can even be in client or supplier organizations. In this example, ‘supplying’ is an abstract role which is carried out by all the retailer’s suppliers.

sented an analysis of the problem. The model's purpose is *not* to provide an abstract representation which is synchronised with its subject system. There would not be a direct link between a purchase in the store and a change of state in the model. The model's abstraction function would be based on patterns of purchases being represented by a statistical distribution which matched historical data. The validity of such representations is a key part of the analysis which must be done in such models. The value of these generalisations is that the simulation is able to compress time; in five minutes it might simulate five years. This enables a large number of different scenarios to be evaluated at a reasonable cost.

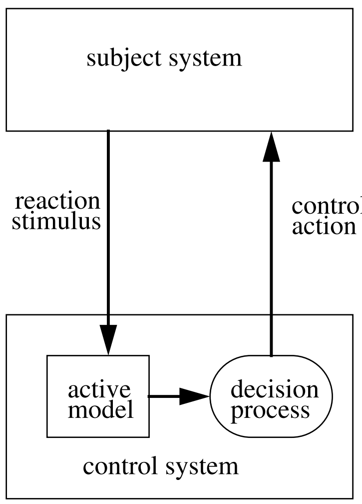
4 Active Models

The idea of an active model is that it is constructed in a modelling medium which allows the modelling relationship to be maintained despite changes in the subject. The active model and its subject are synchronised so that the model reflects the current state of its subject. Whether this synchronisation is in terms of mini-seconds, minutes or days, is part of the abstraction relation between the active model and its subject. Obviously this means that the medium of an active model must be able to change in reaction to some stimulus. IT systems are well suited to play the role of an active model.

An active model can be used for monitoring, that is to provide a high-level view of the current state of its subject. A retail store manager might want an list of the current out of stock items, or a running total of the day's sales for each section of the store.

It is interesting to note that one of the benefits of Workflow systems is the improved monitoring available. Rather than having to phone around the organization to try and find where a particular form has reached, the current status can be obtained by querying the workflow system.

FIGURE 3. An Active Model within a Control System



Active models are often to be found in control systems, for example railway signaling, aircraft collision avoidance, or power station control, where the purpose of the active model is to be a means of monitoring the subject system and to warn of behaviours which breach thresholds (and therefore require attention). Such systems are modelling subject systems which are changing and it is required to understand some, but not all, of these changes.

In a control system there will be a decision process element which uses information from the active model to produce a control ‘action’ which affects the subject system (see figure 3). In a railway signalling system a reaction stimulus indicating that a train had entered a specific section of track might be followed by a control action to alter signals ensuring that no other train enters the same section. In a retail system the sale of an item, a reaction stimulus, might cause a minimum stock level message to be sent to the appropriate person, a control action. If we are using IT to implement an active model based control system, then the active model and decision process will often be closely interwoven. Indeed, the decision process which is required may be a prime determinant of what is modelled.

The addition of a control action through a decision process leads to the model being active in two senses:

- the model actively changes in response to changes in its subject system,
- the model actively affects the behaviour of its subject system.

In both of these the model is maintaining the quality of the modelling relationship. If the purpose of the model is to provide a mechanism for imposing constraints on the subject system, this not only limits the possible behaviour of the subject system but also the scope of the active model.

It is important to realise the scope and generality of these concepts. The word ‘control’ suggests limitation which is perhaps unfortunate. A system such as that depicted in figure 3 is really a feedback system. Feedback systems can be ‘constraint’ systems, whose purpose is to prevent the subject system from going beyond some acceptable threshold. Feedback systems can also be ‘enablers’ of behaviour in the subject system. The retail logistics system may constrain the scheduling of deliveries to ensure that no two articulated lorries will be delivering at the same time. Alternatively, knowing that a delivery has arrived may enable some of its items to be sold before they ever reach the display shelves.

Viewing the co-ordination layer supporting a business process as an active model, we have three issues to resolve:

- What will the modelling relationship be?
- What will be the reactive links between the model and the business process (its subject)?
- What will be the active links between the model and the business process?

For an ‘applying for insurance’ process one set of answers might be briefly:

- the model will record the current status of each application (status can be new, in progress, suspended, quote, in force), and the person responsible for it,
- there will be reactive links for changes of status and of responsible person,
- there will be active links: a) to inform the responsible person when an application has been suspended for 20 days, and b) to inform the new person responsible when an application is assigned, or reassigned, to them.

An alternative set of answers might be:

- the model will record all the company's customers and the current state of all their policies and applications,
- there will be reactive links to update the model when an application is received, a quote is refused, a policy starts, and a policy terminates,
- there will be a active link to ensure that the company's total liability to one customer does not exceed ten million pounds.

These two sets of answers will give quite different active process models. This is to be expected as the purpose of each model is quite different. The former co-ordinates the handling of each application as it passes through the organization; the latter co-ordinates the organization's view of each customer.

We should be clear that the active model's subject system is not necessarily external to the IT which is running the active model. An actuary would probably use a spreadsheet on a networked PC to calculate a premium, and a word processor to phrase a clause to be included into the policy. These are activities in the subject system of the active model. The actuary may also on the same PC receive signals from the active model.

4.1 Implementing an Active Model

In our discussion of active models so far there is one thing which we have omitted: people. The reason for building an active model was to co-ordinate the people involved in the process. To achieve this we must recognise that different people may want different views of the process. There is little point in informing the actuary that a customer has confirmed their official business address if it is the insurance clerk who is waiting for the information. An active model becomes a context manager which knows how the current process state should be interpreted as a current work context for each process participant.

One simple approach is for the active model to provide each person with an agenda of things 'to-do'. Obviously it must recognise when any of these have been done so the agenda is kept up to date.

One view of a business process is as a set of related activities which are distributed among the people involved. This distribution is dictated by different skills, experience and organizational responsibilities. Co-ordination is required when 'personal' boundaries are crossed. This can be because one person must wait for information from another, or because an activity involves more than one person, for example when a decision must be agreed.

It is not recommended for an active model to be directly tied to individual people. If the model depends upon the fact that Marion Sheperd is the logistics expert who allocates deliveries to lorries, then it may come to a standstill if she is sick, on holiday or leaves unexpectedly. A more flexible approach is to recognise an organisational role, say 'logistics managing', which is currently assigned to her. A process' activities are then divided among the roles involved. A person may have a number of distinct roles, and passing a role form one person to another can correspond to delegating responsibility. Experience has shown that it is often useful to have some roles which are explicitly designed to be delegated between people. For example, there might be a role for each application for insurance being processed, which gets delegated between the underwriting clerk and actuary.

When implementing an active model in software we have to deal with three domains. There is the subject system domain, the business process which is being modelled. There is the domain of the active model with its reactive and active links with the subject system. The active model is also the execution of some software. There will therefore be textual (or graphical) programs which are executed to create the active model. The textual description is the third domain. When we abstract the subject system and write the program for the active model we are constrained by the syntax and semantics of the programming language chosen.

4.2 Evolving an Active Model

Consider an active model which has only reactive links with its subject system. What is the value of such a model? It provides an ability to monitor the process but this is only of real value when such information is acted upon. The insurance company manager might notice that most of the company's quotes in a particular area are being rejected. The retail store manager might notice an increasing number of 'stockouts', or that a section's sales are dropping. We would expect such signals to produce some action.

In general there can be three types of action:

- It is decided that the subject system (i.e. the business process) should be changed.
- It is decided that while the subject system does not need changing the active model does.
- It is decided to change resources in some pre-defined way.

Obviously the first two of these are often inter-related. A change of the first type will often also require a change to the active model. Similarly, a change to the decision process associated with the active model may lead to the process' behaviour changing even though it is basically unchanged. Such changes can be viewed as the process and its active model *evolving* in response to changes in its environment. These are sometimes described as second-order changes in contrast to the predicted or first-order changes. However the basic structure of figure 3 still applies.

The ever-increasing pace of business change and the fact that businesses are becoming increasingly dependent on IT means that such changes are inevitable. It is not possible to produce *the* correct active model to support a business process. What is ideal today could become a future legacy system restraining the organization.

5 Experience in Software Engineering

So far we have mentioned only the benefits of active models; for balance we should also mention the difficulties.

Within the software engineering community there has been considerable research on the use of active models to co-ordinate the work of software development teams. The Software Engineering Environment (SEE) is one of the many technologies which attempt to address the ‘software crisis’. Initially SEEs were built around the notion of a database schema description of the product being developed, and a related set of development tools. It was recognised that providing developers with good tools was not, by itself, sufficient to ensure the efficient production of high quality software. Quality products needed quality processes and thus the development process which co-ordinated both users and tools was also vital [3]. This led to Process-Centered Software Development Environments which aimed to yield more stable and predictable software production. The early metaphor used to fashion the development of these environments was essentially mechanistic: they would eliminate the high cost of mistakes by *prescribing* the process to be followed.

Early use of such environments demonstrated that the ‘software factory’ approach was based on inadequate insights into the actual software development process. This has lead to the study of Software Process Modelling (SPM) [4] [5] as a discipline in its own right. Since 1984 there have been nine International Software Process Workshops (ISPW1-9) and more recently a series of European Workshops on Software Process Technology (EWSPT). It also became clear that the environments need not be software specific. They could easily become general purpose Process Centered Environments (PCEs), and suitable for supporting general organizational processes such as financial processes, retail processes etc.

Experience in the use of PCEs has illuminated the inadequacies of the early metaphor and brought into sharp focus the fact that human co-ordination is not merely a matter of written and verbal message passing [6]. It can also depend on what must be communicated, the skills of individuals, the culture of the organization, as well as on available technology. This is true even in seemingly ‘rigid’ organizational processes. As well as complexity in *how* things were done, likewise a new awareness arose around the issues concerning *why* things got done: the complexity of goals (strategic, tactical, operational) placed on individuals, and how they were accommodated into process activity was little understood. In short, there was an emerging need for study at the interface between organization and this new technology.

6 Process Engineering Framework Project

It is against this background that the Informatics Process Group was awarded funding by the Engineering and Physical Sciences Research Council (EPSRC) to explore this area in the Process Engineering Framework (PEF) Project[4].

An ‘engineering’ approach was selected because of the more pragmatic values to be placed on solutions, and the need for rigour, intellectual discipline, theoretical basis, and pooling of solutions. It underscored the need to foster co-operation and communication among practitioners.

The kinds of values seen as appropriate for solutions were that they should be justifiable, reliable, understandable, maintainable, cost-effective, implementable, flexible, and adequate by being satisfactory and sufficient.

4. Further information about the IPG’s research is available from http://www.cs.man.ac.uk/ipg/

The basic objective of the project was to make available to industry a precise and pragmatic guide to the subject which would be suitable for both business professionals and process practitioners. It sought to address the conceptualisation of concerns, the methodological structure for a solution, and an implementation strategy for the solution. This was contained in a framework of cyclical activity to ensure system evolution in harmony with the organization development. Although support technologies are of considerable interest, the IPG does not view active process support as an essential component of every solution. There are situations where responses such as passive models, which allow everyone to see how their contribution fits in, and improved training are much more important than active process support.

There are three threads in the project, of which the first and major thread was Methodology. The Process Analysis and Design Methodology (PADM) in the earlier IOPT Project[5] was further developed with the spectrum of issues addressed ranging from more efficient means of process capture through to minimum risk implementation strategy. In this context our work is being placed firmly on an established theoretical foundation to provide a sound basis for a conceptual framework. Apart from enhancing our understanding of process itself, we sought to identify meaningful characteristics of generic processes and to formulate a sound categorisation schema for such processes.

Another thread was tool development. Automated support is essential to represent organizational processes as a focus for common understanding, to allow for their formal description if desirable, and to analyse and reason about them from alternative perspectives. The group has developed a Process Modeller's Workbench which supports a range of graphical process modelling techniques. Currently, any transformation from a passive to an active model is done by hand and our main active modelling platform is ICL's ProcessWise Integrator. The workbench is however platform independent.

The last thread comprised a series of case studies. These took place in industry and provided essential source information and also provide a suitable test environment for new ideas and methods. This ensured that theoretical work maintained a close relationship with industry concerns.

The deliverables of this project will be in book form, and it is anticipated publication will be in mid-1996.

As well as describing a framework for the engineering of process solutions, PEF provides an initial method of designing such solutions. The technologies are seen as 'intermediaries' between users and existing systems, a way of moderating and structuring the many communications between users and systems, users and applications, as well as between users where appropriate. The active model is used to represent these communications and to actively manage them in real time. PEF advocates a particular approach to capturing processes, based on high level Conceptual Models (figure 2 is an example) and particularly Role Activity Diagrams (RADs) [7]. This gives passive 'as-is' model which is then studied to produce the 'as-wanted' one. An approach is suggested for designing an active model, on the basis of the 'as-wanted' passive one. This is expressed in code and then implemented in an organization.

5. The IOPT Project (Introduction of Process Technology) was a DTI sponsored collaborative project to develop and apply computer support for cooperative work in diverse industrial and government sectors.

7 Conclusions

We believe it to be inevitable that organizations will embed part of their business processes in software systems to exploit the competitive advantages offered by IT networks. This means that a process co-ordination layer will be an increasingly significant component of an organization's IT infrastructure. We propose an active model as an effective approach to designing and evolving such components. An active model approach recognises that IT support for a process will only be partial; the model will emphasise some parts of the process and ignore others. It also recognises that we should expect the process and its model to evolve, and can provide a mechanism to support this evolution. In addition, we believe that technologies must shift their emphasis from manipulating objects to co-ordinating people, and role-based models effectively support this position.

Experience, particularly from software engineering, suggests that designing effective active models is by no means easy, however lessons from the PEF project are reducing the gap between theory and practice, and providing a process engineer with the knowledge, tools and techniques to effectively manage the process assets of the organization. This places designing and evolving active models within an overall methodology.

Acknowledgements

This work has been supported by the Engineering and Physical Sciences Research Council (EPSRC) grant 'Process Engineering Framework.' It has benefitted from collaboration with our colleagues in the IPG and the 'Promoter Working Group', which is funded under the Esprit III programme in Basic Research. R.A. Snowdon's work with the IPG was supported by a grant from the Royal Society and the then Science and Engineering Research Council.We are also grateful to members of the organisations involved in IPG case studies.

References

[1] E. Mumford. "Managerial Expert Systems and Organizational Change: Some Critical Research Issues." In R. Boland and R. Hirscheim, editors, *Critical Issues in Information Systems Research*, Wiley Series in Information Systems. John Wiley and Sons, 1987.

[2] M. Scott Morton, editor. *The Corporation of the 1990s: Information Technology and Organisational Transformation*. Oxford University Press, 1991.

[3] B. Warboys. "The IPSE 2.5 Project: Process Modelling as a basis for a support environment." In N. Madhavji, W. Schäfer, and H. Weber, editors, *Proceedings of the First International Conference on Software Development, Environments, and Factories*, Berlin, 1989. Pitman Publishing.

[4] B. Curtis, M. Kellner, and J. Over. "Process Modeling." *Communications of ACM*, 35(9), September 1992.

[5] A. Finklestein, J. Kramer, and B. Nuseibeh, editors. *Software Process Modelling and Technology*. Research Studies Press Ltd., Taunton, Somerset, U.K., 1994.

[6] D. Wastell, P. White, and P. Kawalek. "A methodology for business process re-design: experiences and issues." *Journal of Strategic Information Systems*, 3(1):23–40, 1994.

[7] M. Ould. *Business Processes: modelling and analysis for re-engineering and improvement*. John Wiley and Sons, 1995.