Information Systems and Enterprise Integration

Pericles Loucopoulos

Information Systems Research Group
Department of Computation
UMIST
P.O. Box 88
Manchester M60 1QD, U.K.

pl@sna.co.umist.ac.uk

Abstract

The aims and objectives of Enterprise Integration is the development of computer-based systems that facilitate the co-ordination of tasks and information flow across organisational boundaries. This is a key challenging area as increasingly organisations strive for improving the co-ordination between systems and ultimately individuals. The success of this undertaking depends to a large extent on the ability of information systems to provide the necessary technological baseline for achieving the desired results.

This paper discusses some important issues underpinning enterprise integration which give rise to a set of requirements for improved design paradigms. In particular the different facets of requirements engineering and supporting environments for enterprise integration are examined in detail.

1. Introduction

Within the field of Information Systems, there is an emerging body of work that advocates a new approach to developing information systems based on the need for a closer alignment between business policy and system operation [Loucopoulos, Wangler, et al 1991; Bubenko, Dahl, et al 1992; Nellborn, Bubenko, et al 1992]. Such an objective stems from the realisation that the lack of interaction between business practices and functionality of supporting information systems results in high levels of maintenance costs and inability to effectively implement changes to the information system or take opportunities for business re-engineering.

Independent of developments in the field of information systems, a body of work is emerging which recognises the need for comprehensive solutions to organisational problems. This body of work is viewed under the term of *Enterprise Integration* (EI). EI has been addressed from a number of different perspectives resulting in a diversity of solutions (c.f. [Petrie 1992a] for a collection of different approaches). Much of the work in this area attempts to provide support to Computer Integrated Manufacturing (CIM) which naturally, is concerned with issues such as control of the manufacturing process and relationship between suppliers of component parts and manufacturers.

Despite the diversity of approaches, it is nevertheless, commonly agreed that the objective of EI is to improve the performance of large business processes (not necessarily contained within a single organisation) by managing the interaction between different agents participating in these processes. This objective is of course not confined only to Enterprise Integration; indeed there are other well established disciplines that have similar objectives. The only distinguishing feature of EI is that it seeks *improvement through the co-ordination of interacting organisations, individuals and systems* [Petrie 1992b].

It is this particular facet that motivates the central tenet of this paper namely that information systems could and should be used as facilitators to enterprise integration. This is based on the premise that the most critical factor that impedes integration is the unavoidable continuous evolution of enterprises and that information systems represent an opportunity for addressing some of the major problems associated with such an evolution. However, if information systems are to prove to be more than a promise they will need to become much more cognisant of the practices of the enterprise that they will seek to serve. They must provide such services so as to facilitate better, more informed decision making and support for changing business practices within improved time scales than is currently the case.

Complexity in enterprises results from the vast amount of detailed information that is often required to describe the enterprise, including its business ontology as well as its operations. Difficulties in managing changes are normally present because of the hierarchical decision making process of most enterprises (strategic, tactical, operational) which results in long delays in the implementation of decisions and the need for information filtering between the various levels of decision making.

The hierarchical decision making process adopted by the majority of today's organisations could in fact be regarded as a hindrance to effective change management. The hierarchy hides information between the different levels. The hierarchy also introduces major delays in implementing decisions.

In a large organisation, for example, there can be many levels of management. The job of the senior management is to make decisions on the general strategy of the organisation. This however, makes their concerns and objectives necessarily more abstract than the those in lower levels of the hierarchy. If for example, the senior management decides that 'the organisation must be profitable in the next financial year', it is up to the middle management to specify how profitable the organisation will have to be and how this can be achieved. At the next level of seniority the goals will be with regard to the tactics and procedures which will ensure profitability of the organisation.

At the heart of effective EI therefore, is the ability to manage change effectively and timely. This in turn necessitates improved quality of information and its communication between the various levels of the decision making hierarchy. It is argued in this paper that improved quality of information is achievable by understanding the concerns and requirements of different enterprise stakeholders of different decision making responsibilities, understanding the goals and their relationships, understanding the business rules that dictate the functioning of the enterprise and ultimately relating all this knowledge to the process of developing supporting information systems.

The paper examines a number of important characteristics found in EI and develops a set of arguments for the nature of enterprise models as well as the enterprise modelling process itself.

The paper is organised as follows. Section 2 discusses the issues underpinning EI and relates these issues to the domain of information systems. In particular, the role of enterprise modelling is examined and arguments in favour of a more comprehensive approach to specification approaches is advocated. This theme is further discussed in section 3. A related issue to specification is the way that on develops requirements for a new improved support mechanism for EI. This gives rise to the issues of generating and evaluating such requirements, a theme further elaborated in section 4.

2. Enterprise Integration and Information Systems

Enterprise Integration can be considered along three dimensions depending (a) on the relationship among participants in the enterprise, (b) the nature of the object of integration and (c) the infrastructure required for supporting integration [Goranson 1992]. Figure 1 depicts the different types of integration along each one of the three dimensions.

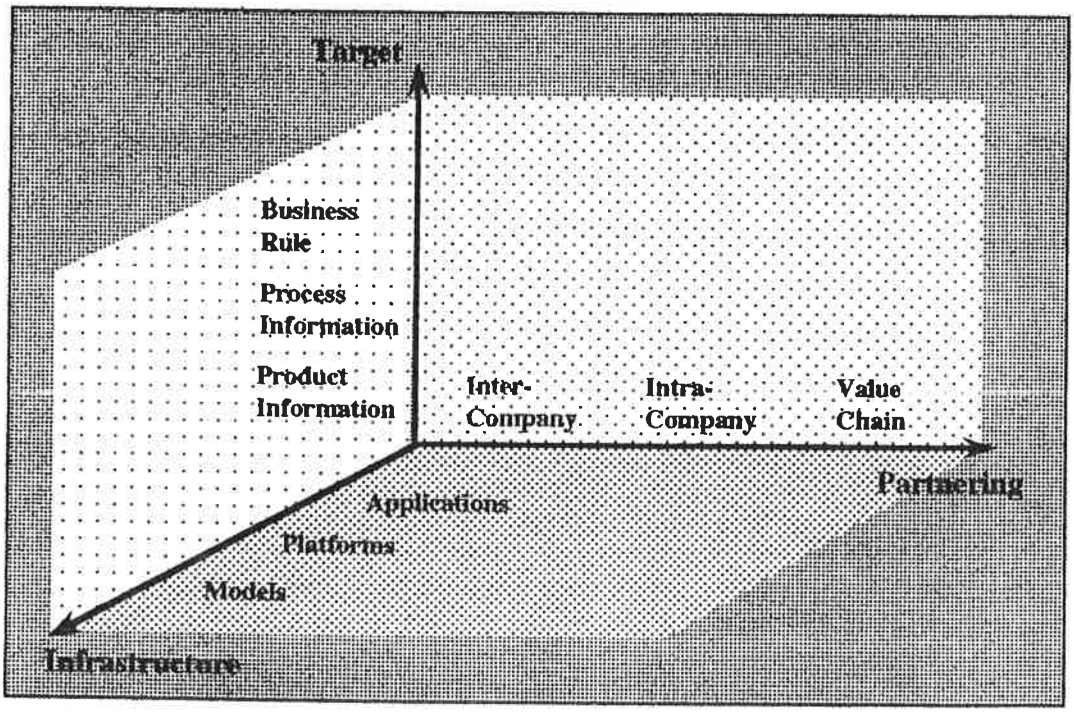

Figure 1: Dimensions of Enterprise Integration

The Partnering Dimension

Intercompany integration is concerned with integration between two or more major partners including the case where some partners may be subcontractors or suppliers. This is normally established for specific, well identified, anticipated enterprise processes for a pre-specified period of time in order to produce some product. *Intracompany* integration is concerned with integration within a single company to support the enterprise. Integration spans different companies within a single corporate enterprise but this type of integration is at the level of infrastructure components. *Value Chain* integration is concerned with the integration of different partners in a dynamically established way which is normally dynamic and short lived in nature.

The Target Dimension

Product Information focuses on information concerning the product being produced by the enterprise. Integration is based on information that is obtained primarily about the product. This type of information is particularly prevalent in intercompany integration since the information about products can be shared between partners without disclosure of the processes that are followed by the partners. *Process Information* focuses on the information about the processes followed in order to integrate different partners. The process of producing some product becomes the dominant characteristic of integration. *Business Rule Information* focuses on the rules that dictate the functioning of the business.

The Infrastructure Dimension

This is the viewpoint with which the field of information systems is most familiar although in their widest sense the terms 'application', 'platform' and 'model' may not be confined to information systems aspects.

A unifying theme within EI is emerging as that of *enterprise modelling*. Understanding the enterprise is considered as the primary objective in EI since such an understanding is used as the basis for developing applications which in turn will be based on various platforms that will facilitate enterprise partnering of one of the three types.

Current efforts to enterprise modelling can be classified according to three different perspectives.

(1) The *enterprise ontology* view. Approaches in this category are concerned with modelling the enterprise in terms of its services and processes. The model of an enterprise's processes can be analysed in order to make recommendations about a better interaction between processes. This view is most appropriate for intra-company situations.

(2) The *computing resource* view. Approaches in this category attempt to model the computing services available to the enterprise with the objective of understanding how different (possibly heterogeneous) computing services can be integrated. There is no intention, using these approaches, to change any of the processes of the enterprise. This view is mostly prevalent in intra-company set-ups although it could be applicable to inter-company situations.

(3) The *decision interaction* view. Approaches in this category concentrate on simultaneous decision making in different organisations or departments. Modelling concentrates on entities and their relationships necessary for interaction between different agents taking part in some common decision making process.

A first observation that can be made about these approaches is that, in terms of an information systems perspective, they tend to rely overwhelmingly on the process and product aspects with little attention being paid to the business rules component. In other words, the 'Integration Target' dimension is only partially addressed. This has the effect that most of these approaches tend to be predominantly used for one or another of the 'partnering' situations but seldom for more than one and hence orthogonality of EI is lost. Furthermore, the absence of clear, comprehensive specifications of the business rules hardly addresses the change management requirement that were outlined in section 1.

More recently there have been proposals that go some way in overcoming these shortcomings [Loucopoulos 1989; Theodoulidis, Wangler, et al 1990; Loucopoulos, McBrien, et al 1991; Loucopoulos, Wangler, et al 1991; Bubenko, Dahl, et al 1992;

Nellborn, Bubenko, et al 1992; Bubenko and Wangler 1993]. The explicit modelling of business rules alongside that of business objects and processes seems not only to go some way in increasing the coverage along the integration target dimension but as Bubenko argues in [Bubenko and Wangler 1993] it provides a natural way of understanding the reason behind the provision of some particular facet of the infrastructure dimension. The field of requirements specification has been extensively covered in the literature, but many of the shortcomings outlined above still persist. Section 3 discusses some improved approaches based on the author's recent work which attempts to overcome these shortcomings.

Requirements specification is one facet of the process of trying to understand and document requirements. In a complex domain such as EI there is a need to adopt methods that generate and assess requirements. This is a direct result of a second observation which states that enterprise modelling, is primarily used in order to devise a course of action to change an existing state of affairs. Typically one models the current state of the enterprise and considers possible future states. As such it constitutes a *design* activity..Design in the EI context is not concerned with traditional engineering in the sense that it does not deal with material artifacts but rather with plans, organisational aspects and the like, in other words with symbolic devices. In this larger sense of engineering, one can consider EI and any supporting information system to fall very firmly in the sphere of activities that constitute what Simon termed "sciences of the artificial" [Simon 1969].

A design process associated with problems in the EI domain involve: (a) a set of requirements to be met by some artifact, (b) the output of the process is some design, (c) the goal of the designer is to produce a design such that if an implementation of this design were to materialise then the artifact would satisfy the requirements and (d) the designer has no knowledge of any design that satisfies the requirements. These properties of the design process are indeed not confined to EI problem domain and some authors are stating that there are aspects of design that are domain-independent [Dasgupta 1991].

This second observation results in an examination of the way that one structures the process of identifying requirements within the EI domain and the effect that this process has on developing information systems.

Enterprise systems fall in the category of the so called "wicked" problems where the space of possible designs is so large that a designer will strive to reach *satisfactory* rather than *optimum* designs. If the design meets the criteria established by the problem stakeholder then the process has reached some (temporary) stability. As the enterprise changes so must the design process recommence to reach some other satisfactory solution. Dasgupta has defined two qualitative laws, the *hypothesis* law and the *impermanence* law that reflect this working practice [Dasgupta 1992]. Dasgupta argues that the design process involves "hypothesis creation, testing and modification" and that "a design in a given state is never guaranteed to remain in that state". In this sense any requirements expressed within an enterprise could be considered as propositions [Vitalari 1992] or plausibility statements [Aguero and Dasgupta 1987; Hooton, Aguero, et al 1988].

Testing a hypothetical design for satisfactoriness requires the capture of appropriate evaluation criteria against which the design must be tested. In enterprise modelling there is therefore, a need to consider the characteristics that the desired artifact must display in the context of *generating requirements* and *assessing requirements*. Requirements generation and assessment are considered in section 4.

3. Modelling Along the Integration Target Dimension

Enterprise modelling along the integration target dimension requires modelling of information about all three components, products, processes, and business rules if the specification is to be supportive of any aspect of integration. The specification emerging from enterprise modelling should consider three interrelated domains, the enterprise domain, the system domain and the system use domain [Katsouli and Loucopoulos 1991]. The relationship between the three domains is depicted in figure 2.

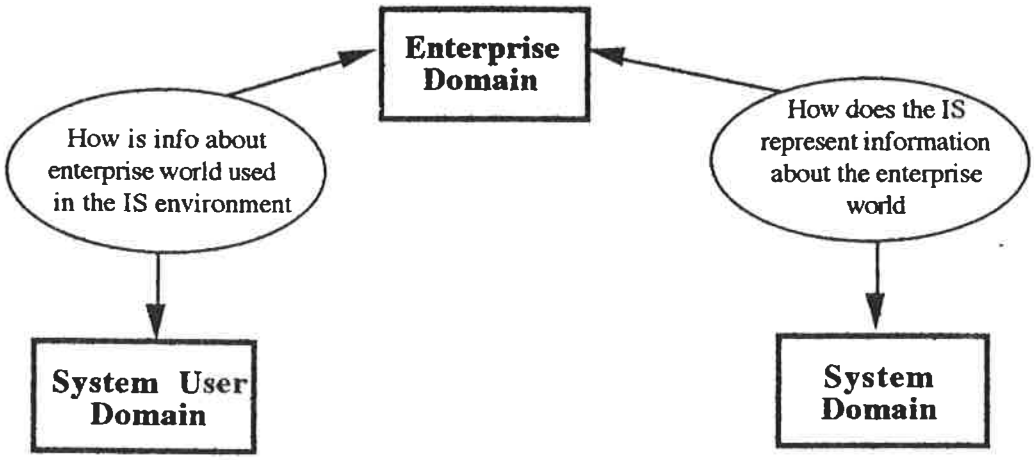

Figure 2: Different domains relating to enterprise modelling

The main objective of enterprise modelling is to build a model which represents both a conceptualisation of what is needed and a statement of the 'enterprise' domain under consideration. There is a natural relationship between this domain and those of 'system' and 'system user' domains.

By modelling the objectives, actors, concepts, and activities, a rationale is established not only about the enterprise itself but also about the infrastructure that supports or will support in the future the enterprise [Bubenko and Wangler 1993]. Bubenko and Wangler present a detailed framework of enterprise modelling which involves a number of different submodels. This approach seems to provide an appropriate methodological backbone to a task that is often extremely difficult to carry out due to uncertainty, vagueness and complexity inherent in this domain.

This section is not so much concerned with the methodological issues raised in [Bubenko and Wangler 1993] but rather on the specification details that are appropriate for the task in hand.

In the *system domain* descriptions are guided by the enterprise model which defines the area of concern (functional requirements) together with any non-functional requirements. The overriding criterion is one of implementation efficiency within the constraints identified at the requirements phase. The level of detail and type of representation of these specifications depends on the development paradigm adopted. Typically, design specifications include conceptual data objects which represent information about the enterprise domain e.g., 'books', 'borrowers', etc., system activities e.g., 'generating daily reports' etc., and *correspondence* between the information maintained by the information system and enterprise domain e.g. 'for every borrowing there will be an entry in the information system' or 'the information system will record only the books currently on shelves'.

The *system user domain* is concerned with the environment within which the system is embedded. Typically these specifications describe the input and output relationships. A developer needs to identify the interaction between the enterprise domain and the information system and the implementation of this interaction within the information system itself.

Definitions

A ***Business object*** is anything of relevance to the enterprise about which information is captured (even transient information). An occurrence of a business object must be individually identifiable. It can be a tangible object (e.g. person), an abstract notion (e.g. cost centre), an activity (e.g. accident) - about which information is kept.

A ***Business process*** describes behaviour on business objects by consuming and producing business objects.

A ***Business event/trigger*** is an observable happening which could be: an external signal, or a control flow (including temporal conditions).

A ***Business rule*** is: a statement that specifies the legal existence of the business objects, or a statement that defines the legal condition for a process to take place.

These definitions are expressed in the 'top-level' metamodel shown in figure 3.

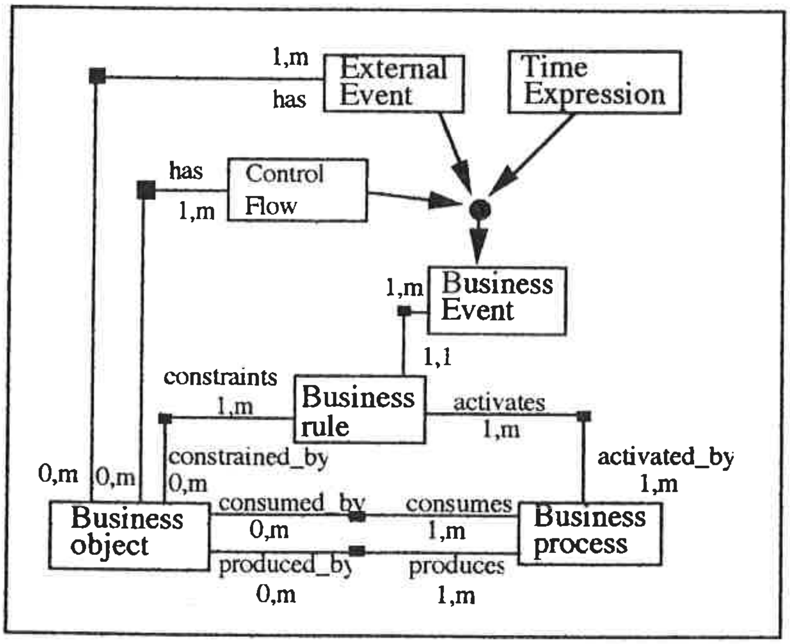

Figure 3: A metamodel for business objects, processes, triggers and rules

An Example

To demonstrate the applicability of the metamodel of figure 3 consider the enterprise of an airline where there are a number of objectives, not only those of the airline itself but also of other external businesses such as such those of travel agencies, security agencies, catering companies etc. This is a typical example of inter-company integration, normally with the focus around the process of 'ticket reservation'. This process however, can only be understood if the business objects associated with it are also understood and the business rules that determine the functioning of this process are modelled. These three cornerstones will be fundamental to determining the application and its required platform. A detailed specification of the business rules is beyond the scope of this paper. Also the relationship between the specification of the enterprise domain and the corresponding specifications at the system and system use domain are not included in this paper but these are detailed in [Katsouli and Loucopoulos 1991].

Consider the typical objectives of different stakeholders in the airline enterprise, samples of which are given below.

Chief Executive

- when a flight is full, VIPs are the first to be upgraded and/or given a seat in preference to others.
- discounted tickets should be offered to politicians since these people make important decisions affecting the airline
- information about frequent fliers and executive club members should not be made available to outside contractors

Chief Security Officer

- the number of bags in the aircraft's hold should tally against the list of passengers on board
- passenger lists should not be made available to the public
- passengers should check-in only once

Catering Section

- the food loaded is dictated by the number of passengers travelling in a particular class, irrespective of the fare they paid
- a predicted number of passengers on a flight must be available 24 hours prior to departure
- passengers requiring special diet must indicate their request 24 hours prior to departure

Travel Agency

- an agent is responsible for holding and cancelling reservations
- tickets offered by an agency have different fares to be negotiated with the sales department of the airline

Airline Sales Manager

- a ticket may only be issued when a fare is paid
- for some fares a reservation can be held and not confirmed
- when a discounted ticket is booked, the normal book-ahead requirements do not apply
- all tickets must carry appropriate endorsements relating to the terms and conditions of the issue of tickets

These are only a small set of goals held by different stakeholders within the enterprise. Typically, in any enterprise there will be many, sometimes contradictory, goals which must be documented, analysed and reconciled if necessary.

It can be seen from even this very small sample of requirements that there is a plethora of business rules that can be regarded as business goals (such as the chief executive's requirements), or detailed statements about the functioning of the enterprise (see for example the airline sales manager's view) or even rules imposed by the outside world (for example the chief security officer's perspective).

The way that these rules could be handled is well documented elsewhere c.f. [Loucopoulos, McBrien, et al 1991; Petrounias and Loucopoulos 1993] for a description of the TEMPORA languages as well as detailed workings with these languages.

Fundamental to the view that it is important to understand the functioning of an enterprise in terms of its business rules as well as in terms of its products and processes is the *teleological view of systems*. According to the teleological view a system (such as an organisation, machine, human etc.) has a set of goals which it seeks to attain. Thus, the teleological view attempts to explain a system's behaviour in terms of its goals. A *goal*, then is a defined state of the system. Since a state is described in terms of the values of a number of parameters, a goal can be alternatively defined as *a set of desired values for a number of parameters*. For instance, if the system is a profit making organisation then one of its goals might be

`to make profits of $1M in the next financial year`

Here, the goal parameter is "profits" and the desired value is "1M".

Goals can vary in their degree of specificity (or else abstraction). In general, the more desired values are mentioned, the more specific the goal is. Thus, the goal

`to make profits of $1M in the next financial year`

is more specific than the goal

`to make profits in the next financial year`

The varying degree of specificity in goals, has a lot to do with the hierarchical way most of the systems are organised. In a large organisation, for example, there can be many levels of management with most of the 'vague', qualitative goals pertaining to the decision making processes higher up in the hierarchy. These higher-level goals do not usually specify 'when', 'how much' or ' how'. The goal `'the organisation must strive for profitability'` without specifying how this profitability will be measured or when it must be attained is an example of such a type of qualitative goal. As it usually happens, high-level goals are decomposed into a number of goals of increasing specificity and appropriately, of a quantifiable nature. In many situations, a subgoal may be instrumental to more than one (super)goals, thus forming a lattice of goal relationships.

It should be noted that goals rarely occur in isolation. People often find themselves in situations in which they possess several goals simultaneously, either because they have initially more than one goal or because the execution of a plan for a single goal impinges upon other goals. Thus, in order to develop a successful plan for EI there is a need to consider the interactions between different goals. This requires an understanding of the nature of goal interactions. Such understanding includes a classification of the possible kinds of goal interactions together with the circumstances during which each type of interaction may arise. Furthermore, knowledge about the consequences of each interaction must be specified and most important knowledge about possible ways of dealing with these consequences must also be considered.

The interactions into which goals may enter can be considered in terms of two perspectives:

(a) Ownership of Goals

In this perspective two classes can be distinguished, *internal goal interactions* representing relationships between goals of the same agent (person, groups, organisation) and *external goal interactions* representing relationships between goals of different agents.

(b) Nature of Interaction

This involves an analysis of whether the interaction is favourable or unfavourable. In the case of internal goal relationships unfavourable interactions are called *goal conflicts*, where a goal inhibits the fulfilment of another goal, while a favourable internal goal relationship is called *goal overlap*, when the goals are similar enough so that a plan for both goals is more efficient than the plans for each goal. When an unfavourable relationship occurs between goals belonging to different agents then there exists a *goal competition*, while a favourable interaction between goals of different persons is called *goal concord*.

The goal interactions which are more interesting are mainly those falling in the goal conflicts class, the resolution of which represents arguably the most difficult and challenging aspect of the EI modelling process.

Goal conflicts cause difficulties in the formulation of a successful plan. These goals are in danger either of failing to be achieved or of being abandoned. In this case in order to solve this problem a modeller will either try to eliminate the cause of conflict, or modify the plan or abandon one of the goals.

Three classes of goal conflicts can be distinguished:

- Resource Limitations. Two goals are in conflict when the plans for the goals require a common resource and there is insufficient quantity of that resource available for both plans. Several classes of resources can be distinguished for the purpose of understanding goal relationships such as time, consumable functional objects, non-consumable functional objects, abilities etc.

- Mutually Exclusive States. These are states involved in the fulfilment of different goals which may simply be incompatible with one another, that is trying to achieve these goals will require mutually exclusive states to come into existence.

- Causing a Preservation Goal. Executing the plan for one goal can cause a preservation goal to come into being creating thus a conflict between the original goal and the preservation goal.

Understanding the business goals, rules, constraints their interrelations as well as their relationship to the business products and processes is a major step forward in beginning to formulate plans for change. Planning concerns the process by which people select a course of action, that is understanding the problem, deciding what goals to pursue, creating plans to satisfy these goals, and eventually executing these plans.

The concept of goal is central to the planning process. Once a goal has been determined then a plan to satisfy this goal must be decided. However, having decided to follow a particular plan there is a need to investigate how this plan may interact with other goals and plans that are being considered. That is, the interaction between this goal and other goals must be detected and the plan modified in order to accommodate the interactions.

This discussion leads on to the appropriateness of the requirements capture and analysis process and in particular the issues of generating requirements and assessing requirements. This was briefly introduced in section 2 and further elaborated in section 4.

4. Generating and Evaluating Requirements

As stated in section 2, the process of capturing requirements for designing an artifact such as an information system to support an enterprise or redesigning the enterprise itself requires the use of approaches (examples of which can be found in [Olle, Hagelstein et al 1988; Olle, Sol et al 1983; Olle, Sol et al 1982]) that go beyond the traditional documentation and representation aspects.

A requirement for an improved system begins with an initial hypothesis which is vague and requires further elaboration in order to produce the desired result. In this sense, requirements analysis involves the generation of hypotheses, and subjecting these hypotheses to a process of disconfirmation. These hypotheses are generated and evaluated against attributes of the intended system which it is anticipated that it will meet a desired state in the enterprise.

The idea of using hypothetical or plausibility statements in the design process was introduced in [Aguero and Dasgupta 1987] in the context of complex designs for computer architectures. More recently, Vitalari argued in favour of adopting a propositional viewpoint in requirements analysis [Vitalari 1992]. Plausibility-driven analysis of enterprise requirements differs significantly from the traditional approach which is instead concerned with 'discovering requirements and documenting them'.

Central to a plausibility-driven analysis approach is the generation of hypotheses (requirements) and the evaluation of these hypotheses.

Hypothesis formulation is based on the vision or belief that the participants in the analysis process may hold about the intended system, its role in the enterprise, its effects on organisation practices, its effects on personal and group status and so on. In short the participants define a set of system characteristics which in their opinion will lead to an improved situation. Hypotheses evaluation is based on the idea that every hypothesis undergoes criticism and is thoroughly examined for its validity i.e. looking for evidence to disconfirm the hypothesis.

Using a plausibility-driven approach implies that a specification describing objects, processes and business rules represents a set of propositions that are considered to be true until proven otherwise. The question is "what mechanisms could one use for testing these propositions?". There are some obvious candidates such as prototyping, although in practice prototyping has proved to be useful for evaluating aspects of the system use domain rather than the enterprise domain. Furthermore, prototypes are very often regarded as the 'almost finished' product and have a tendency to persist beyond the evaluation stage.

A promising approach seems to be that of visualisation. The advantage of visualisation over prototyping is that design decisions will not have to be made prematurely. A requirements specification is likely to change many times before proceeding to design and visualisation should help in testing a succession of propositions which are increasingly closer to end users' perceptions about the target system. In the context of conceptual specifications, visualisation involves the animation of the behaviour of a system as hypothesised at some point in time and a visual interface reflecting the results of events upon the graphical - and where appropriate the textual - components of the specification.

Animation of a specification is the process of providing an indication of the *dynamic* behaviour of the system by walking through a specification fragment in order to follow some scenario. Animation can be used to determine causal relationships embedded in the specification or simply as a means of browsing through the specification to ensure adequacy and accuracy by reflection of the specified behaviour back to the user.

Although, interactive graphics and animation were used successfully in programming, these techniques have not been fully applied in conceptual modelling.

The architecture shown in figure 4 uses the same techniques, with program animation, in order to provide an indication of the dynamic behaviour of a specification. Such an approach has been used on a number of applications and is reported in [Lalioti and Loucopoulos 1993].

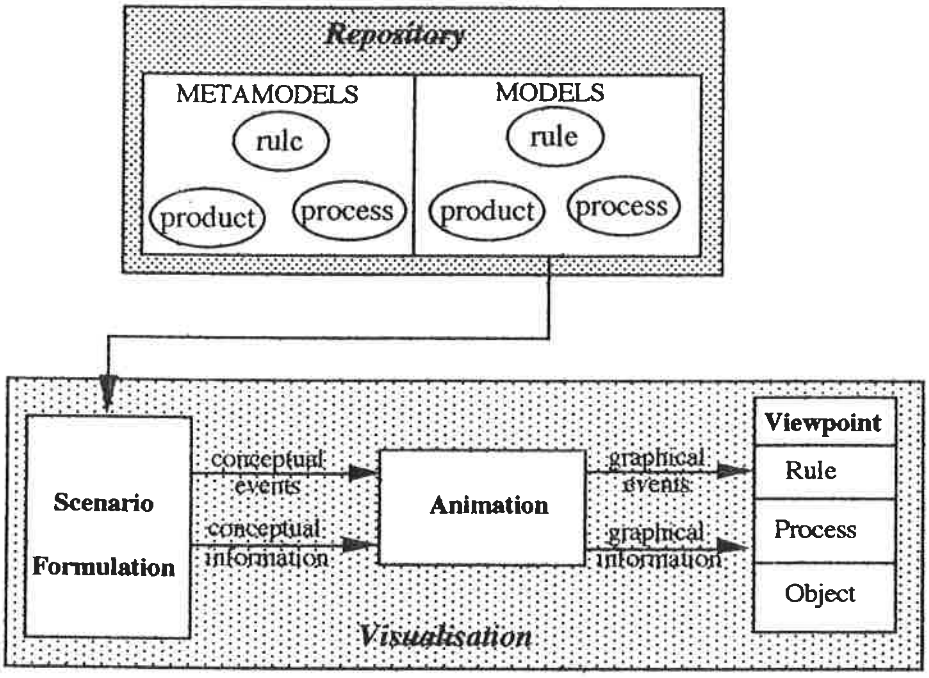

Figure 4: Visualising conceptual specifications

One approach is the use of scenarios which could with the evaluation of propositions contained in a conceptual specification. For example, starting from the functional model one could ask "What If ?" questions. In the architecture depicted in figure 4, the component which is responsible for forming the scenarios and walk through a specification fragment in order to follow the scenario, is the *Scenario Formulation*. The algorithm will start from the functional model and analyse the behaviour of the system by checking the interrelationships of the models. It will also report any events and conceptual information to the second component, namely the *Animation*. More than one

View can be used in order to visualise the various interrelationships between the conceptual models. Each View is responsible for updating its projection on the specification according to the events, and all Views are updated simultaneously. The Views may use colour, movement and animation techniques; thus providing participants to the analysis process with multiple animated views.

5. Conclusions

The field of EI presents many opportunities for improving the functioning of organisations, and improving communication between participants in the enterprise at both corporate and individual levels. Successful implementation of EI strategies relies to a large extent in our ability to understand current environments on the basis of this understanding to devise new solutions for improved enterprise functioning. This paper has argued that the field of information systems has much to offer in this undertaking, particularly in our ability to specify, generate and evaluate requirements for improving the infrastructure supporting the enterprise. Benefits to be accrued from this include:

- **Improving change management** by explicitly identifying and modelling those aspects of enterprises that are liable to change and by developing information systems that go beyond the automation of existing processes.
- **Proving integration of views within an enterprise** by adopting an approach which encourages a co-operative generation and assessment of requirements.
- **Improving communication between the actors involved in an enterprise** by assisting the communication between different levels of the decision making hierarchy.
- **Relating information systems to business strategy** by adopting a specification paradigm that facilitates the modelling of business goals and their realisation in information systems structure.
- **Analysing the impact of changes within an enterprise to the IT platform** by modelling the goals and business rules and their impact on functional and non-functional aspects of the IT platform.

Information systems are entering a new phase moving beyond the traditional automation of routine organisational processes and towards the assisting of critical tactical and strategic enterprise processes. We need to strive for new development paradigms that concentrate on organisational aspects, delivering systems that are closer to the culture of organisations and wishes of individuals.

References

[Aguero and Dasgupta 1987] Aguero, U. and Dasgupta, S., *A Plausibility-Driven Approach to Computer Architecture Design*, Communications of the ACM, 30, 11, pp. 922-931.

[Bubenko, Dahl, et al 1992] Bubenko, J., Dahl, R., Gustafsson, M., Nellborn, C. and Song, W., *Computer Support for Enterprise Modelling and Requirements Acquisition*, SISU, F3 Project Internal Report, 1992.

[Bubenko and Wangler 1993] Bubenko, J.A. and Wangler, B., *Objectives Driven Capture of Business Rules and Information Systems Requirements*, IEEE Conference on Systems, Man and Cybernetics, pp. 1993.

[Dasgupta 1991] Dasgupta, S., *Design Theory and Computer Science*, Cambridge, UK, Cambridge University Press, 1991.

[Dasgupta 1992] Dasgupta, S., *Herbert Simon's "Science of Design": Two Decades Later*, 1st International Conference on Intelligent Systems Engineering, Edinburgh, UK, pp. 1992.

[Goranson 1992] Goranson, H.T., *Dimensions of Enterprise Integration*, Enterprise Integration Modeling, pp. 101-113, 1992.

[Hooton, Aguero, et al 1988] Hooton, A., Aguero, U. and Dasgupta, S., *An Exercise in Plausibility-Driven Design*, IEEE Computer, 27, 7:, pp. 21-33.

[Katsouli and Loucopoulos 1991] Katsouli, E. and Loucopoulos, P. *Business Rules in Information Systems Development*, Workshop on Next Generation CASE Tools, Trondheim, Norway, 1991.

[Lalioti and Loucopoulos 1993] Lalioti, V. & Loucopoulos, P. *Visualisation for Validation*, 5th International Conference on Advanced Information Systems Engineering (CAiSE•93), Paris, June 1993, Springer-Verlag Lecture Notes on Computer Science No 685, pp 143-164.

[Loucopoulos 1989] Loucopoulos, P., *The RUBRIC Project-Integrating E-R, Object and Rule-based Paradigms*, European Conference on Object Oriented Programming (ECOOP), Workshop Session on Design Paradigms, Nottingham, UK, pp. 1989.

[Loucopoulos, McBrien, et al 1991] Loucopoulos, P., McBrien, P., Schumacker, F., Theodoulidis, B., Kopanas, V. and Wangler, B., *Integrating database technology, rule-based systems and temporal reasoning for effective information systems: the TEMPORA paradigm*, Journal of Information Systems, 1, 2, April:, pp. 129-152.

[Loucopoulos, Wangler, et al 1991] Loucopoulos, P., Wangler, B., McBrien, P., Schumacker, F., Theodoulidis, B. and Kopanas, V., *Integrating Database Technology, Rule Based Systems and Temporal Reasoning for Effective Information Systems: The TEMPORA Paradigm*, Information Systems, 1, 1.

[Nellborn, Bubenko, et al 1992] Nellborn, C., Bubenko, J. and Gustafsson, M., *Enterprise Modelling - the Key to Capturing Requirements for Information Systems*, SISU, F3 Project Internal Report, 1992.

[Olle, Hagelstein et al 1988] Olle, T., J. Hagelstein, I. G. Macdonald, C. Rolland, H. G. Sol, F. J. M. Van-Asche and A. A. Verrijn-Stuart, *Information Systems Methodologies - A Framework for Understanding*, North Holland, Amsterdam, The Neteherlands.

[Olle, Sol et al 1983] Olle, T. W., H. G. Sol and C. J. Tully (Ed.), *Information System Design Methodologies: a Feature Analysis*, North Holland, Amsterdam.

[Olle, Sol et al 1982] Olle, T. W., H. G. Sol and A. A. Verrijn-Stuart (Ed.) *Information System Design Methodologies: a Comparative Survey*, North Holland, Amsterdam.

[Petrie 1992a] Petrie, C.J.,*Enterprise Integration Modeling, Proc of 1st International Conferene*, Cambridge, Massachusetts & London, England, MIT Press.

[Petrie 1992b] Petrie, C.J., *Introduction*, Enterprise Integration Modeling, Proc of 1st International Conferene, pp. 1-14, 1992.

[Petrounias and Loucopoulos 1993] Petrounias I. & Loucopoulos P. *Design, Implementation and Evaluation of an Application following the Deductive Approach*, 4th International Workshop on the Deductive Approach to Information Systems and Databases, Costa Brava, Spain, 1993.

[Simon 1969] Simon, H., *The Sciences of the Artificial*, Cambridge, Massachusetts, MIT Press, 1969.

[Theodoulidis, Wangler, et al 1990] Theodoulidis, C., Wangler, B. and Loucopoulos, P., *Requirements Specification in TEMPORA*, Advanced Information Systems Engineering: Second Nordic Conference CAiSE '90, Stockholm, Sweden, May 1990, Proceedings, Berlin, pp. 264-282, 1990.

[Vitalari 1992] Vitalari, N.P., *Structuring the Requirements Analysis Process for Information Systems: A Proposition Viewpoint*, Challenges and Strategies for Research in Systems Development, W. W. Cotterman and J. A. Senn (ed.), John Wiley & Sons Ltd., 163-179.