Modeling dynamic management features of virtual enterprises

Peter Bernus^a, Peter Bertok^b, Laszlo Nemes^c

*^aGriffith University Nathan (Brisbane) QLD 4111,
Australia
email: bernus@cit.gu.edu.au*

*^bRoyal Melbourne Institute of Technology P.O.Box 71,
Bundoora Vic 3083, Australia, email: pbertok@rmit.edu.au*

*^cCSIRO Division of Manuf. Technology, Locked Bag 9,
Preston Vic 3072, Australia email: lnm@mlb.dmt.csiro.au*

Abstract

We investigate essential properties of virtual enterprises and the consequent modeling requirements. Agent autonomy is defined. Virtual enterprises are treated as autonomous entities built out of autonomous entities. Requirements are derived from two types of virtual enterprises, a repetitive and a one-of-a-kind production or service enterprise. Properties that need to be represented in virtual enterprise models include the dynamics of decision making, the negotiations among participants for the delineation of domains of autonomy, authority, beliefs and responsibilities, the mapping of organizational entities to decisional roles, the ability to identify and analyze a variety of conflict types and the existence of conflict resolution paths. These properties have to be analyzed by matching enterprise engineering tools. An ontological theory is also needed to systematize the concepts that must be supported by the protocol languages for parallel distributed planning, scheduling and control algorithms in the virtual enterprise.

Keywords

Virtual enterprise, enterprise modeling, co-ordination

1 INTRODUCTION

A virtual enterprise is a temporary alliance of companies for the lifetime of a common project, solution for a problem, or joint production of service or product. The rapid development of communication and networking infrastructures gave new impetus to the development of virtual enterprises, because new ways of interactions between participants have eliminated the time and space gap between partners.

Virtual enterprises are such entities which, from the point of view of their service to the customer, appear to be one entity, but in fact are formed from several autonomous entities, or partners. The property that differentiates a virtual enterprise from an ordinary value chain is the fact that there is a single locus which takes full responsibility for the entire value chain of its product or products, even though the task is carried out by many participants.

Several disciplines have treated the problem of how to build a larger autonomous system out of autonomous components. The artificial intelligence literature developed blackboard systems and co-operative dialogues to externalize mental states and beliefs (Erman et al 1980, Minsky 1996, Winograd-Flores, 1990), the planning and scheduling area developed co-operative planning (Balasubramanian 1997) and holonic manufacturing cells (Deen 1993, Mathews 1995, Tamura 1995) similar techniques were used for computer supported co-operative work (Schmidt 1996), management science developed dynamic forms of organization, e.g., matrix organizations (Lawson 1986, Wall 1984). Co-ordination science attempts to develop methodologies and paradigmatic models for the same purpose (Malone, 1993). All have an application of the basic approach: building agents out of agents.

There is a new interest in how virtual companies can be created easier than using traditional methods. Enterprise modeling holds a promise because it takes out some of the trial and error component from creating a new, better managed value chain. Enterprise modeling languages have been developed and used to describe and simulate business processes, but the development of viable structures, good quality reusable models for virtual enterprises is far from trivial. We investigate some of the properties that virtual enterprises need to have. These properties must then be predictable from the models which we use to design such enterprises.

Typical virtual enterprise formations are:

- One of a kind engineering endeavors such as designing and building large petrochemical plant, roads, major facilities, organized as major projects - often on an international scale
- Consortia for production, service or for research, with an alliance of partners limited to a common mission.
- The creation of virtual enterprises is mainly done by business executives, or highly paid (and stressed) project managers who have little more help than their experience as to how, under time pressures, successful virtual

and methods, once a design is in place the change of such a virtual organization is often lagging behind the needs.

Our goal is twofold:

- We want to analyze, and identify what are the desired *types* of properties that management should be able to design and verify at the initial creation of a virtual enterprise, and to continuously assess and control during its operation;
- We want to derive needs, or requirements, based on this analysis, as to what types of modeling tools and languages are, or would be especially suitable for the support of the virtual-enterprise engineering process. Hopefully an appropriate combination of existing tools, or some extension to such combination, will be able to answer to the so identified needs.

2 RESEARCH METHOD

We draw conclusions from previous case study experience, in the Globeman 21 consortium (on project enterprise); and use GERAM (Generalized Enterprise Reference Architecture and Methodology) as the theoretical framework for the analysis (Bernus, Nemes, 1996). The method is selected because GERAM is an open, very wide scope life-cycle architecture with enterprise creation and change being its main focus. We also use the theory of GRAI Grids which are the foundation for the decisional modeling capability of GRAI-GIM (Doumeingts, 1994) (a GERAM compliant enterprise reference architecture and methodology). Further, the division between the management architecture and the mission fulfillment architecture corresponds to PERA (Williams, 1994).

We identify typical virtual enterprise types, and essential types of properties that should be accounted for in the Enterprise Models (EMs) describing them. Enterprise modeling languages (EMLs) and appropriate enterprise engineering tools (EETs) can be selected for the support of virtual-enterprise engineering based on their competency to support the description and analysis of the above properties.

We hope that through the use of suitable modeling tools business executives will get help to better design multi-company ventures, e.g., to be able to analyze and optimize, through explicit models, the global behavior of their virtual enterprise.

Because the management system of virtual enterprises needs to be different from incorporated ones (due to looser connection between partners) it has to be studied and understood what special needs virtual enterprise designers face, and only after that attempt to select the modeling methods and languages to model them. Although the types of views through which to model virtual enterprises can be predicted (based on CIMOSA (Vernadat,

purpose of representing these views needs to be gauged to the problem domain.

2.1 Available methods for enterprise modeling

Business process modeling has been extensively used to improve the material and information flow in a company, ameliorating the process from some respect or another (cost, quality of product, time to market etc.). Similar modeling exercises are expected to be applicable (with like expected results) in the context of virtual enterprises improving the information and material exchange among the parts of the process as implemented by the participating partners. In other words such models help integrate the supply chain. Special care must be taken with virtual enterprises in the information exchange and management of the chain (or product life-cycle), to avoid misinterpretation and rework. The specification of the interfaces using semantically rich schema languages (such as Express) is used in the extensive efforts to define STEP protocols for product data exchange (Kahn 1995)

CIMOSA advocates the modeling of the enterprise's control system such that the resulting models, when mapped to the implementation and operational levels, can be used to control business processes. The resulting paradigm is called model-based control and is an important technique of supply chain integration, because of its ability to integrate the operational control level. In virtual enterprises the same operational control integration is applicable, only the integration infrastructure must be distributed over a wider area.

Traditional enterprise modeling (applied extensively in many industries in the design of flexible manufacturing cells, job shops, banking business processes, or integrated circuit development processes) separates the design of the business process from the management and operation of the process. i.e., first design (or re-design) the processes of the enterprise, institutionalise them and then operate them.

Virtual enterprises are dynamic in nature (eg a project enterprise starts operating its project management component before the actual project gets fully designed) and enterprise design and change is a function of the operation itself (more specifically it is part of its management component).

Virtual-enterprise design needs various level interactions with partners who take part of the virtual enterprise's business functions. Thus the design process involves various interactions between partners. But because this interaction is also part of the enterprise operation, enterprise modeling must extend to the determination of protocols among partners which are suitable for negotiating about enterprise design.

3 CHARACTERISATION OF VIRTUAL ENTERPRISES

3.1 Virtual enterprise

A virtual enterprise, as any enterprise, must have a clear goal and a clear mission. It also should demonstrate a goal seeking behavior in its autonomous decision making, act upon its set of beliefs, and interact with its environment (O'Leary 1997). In general it is expected that a virtual enterprise makes plans, and its activities follow those plans to achieve the objectives. The term 'agent' is used to describe this type of behavior, in the traditional AI meaning of planning agent. From the point of view of the applicability of this model it is immaterial if planning indeed takes place or not, and if missions and goals are made explicit; what matters is that an external observer can describe the enterprise as (if it were) a planning agent.

Since the partners that make up the virtual enterprise are also agent-like, in the followings we make an attempt to treat a virtual enterprise as an agent, or collection of agents, which contribute to the virtual enterprise's mission but 'keep their autonomy'. Since what this autonomy is not defined yet, we placed the statement between inverted commas – autonomy being defined later in this paper.

Modeling an enterprise has to consider the *function* of the enterprise, from its mission down to operational level procedures. The level of modeling detail should correspond to the level for which there are suitable human organizational, hardware or software entities that can perform each elementary activity modeled. The description not only needs to give a static picture, the dynamics (e.g. the expected life history) is also to be accounted for.

The model must also reflect the actual *structure* of the enterprise, including its resources (such as human and machine), so that the behavior, reactions and operation (i.e. the functions) can be understood in relation to the structure. This latter type of relationship becomes an important issue when autonomy is considered.

In the following we concentrate on the modeling of the management and control of the virtual enterprise.

Functionally, enterprise management can be described as carrying out three main types of functions (a) managing the products, (b) managing its resources, and (c) co-ordinating (a) and (b) (Doumeingts, 1994). Product management covers all product-related activities from product planning to production planning and related services. Resource management includes human, machine, financial and other resources that are needed to perform the operations that fulfill the enterprise's mission, and is responsible for operational level control. Co-ordination is resolving the conflicts between the objectives of (a) and (b) as well as gives them direction or mission. These functions make up the decisional system of the enterprise.

Figure 1 Interaction in a Traditional Value Chain

Structurally, enterprise management can be described as a management organization, consisting of the agents¹ (people and groups of people) who take the decisions. The above mentioned relationships among these individuals and groups are determined by a) the decisional roles they take and the dynamics of this system (b) intangible social relationships. It is expected that at least the tangible relationships are brought to light in the form of a model, such that organizational conflicts be apparent and can be designed-out of the system.

In traditional value chains one problem is that links between participating agents are limited to the operational or executional levels. This is represented in Fig.1.

3.2 Service level agreements / outsourcing

One approach that has been adopted by many large organizations is to decompose what was once one organization into several autonomous parts. This is done through giving increased autonomy for the individual components, and choosing a co-operation method (e.g. from 1-3 above). The technique is commonly known in today's management as service level agreements and outsourcing (depending on the level of autonomy of the components).

The method is trying to avoid conflicts by setting out the rules in the enterprise design phase and requiring clarification or explicit agreement on these rules before any decision is taken. The co-ordination among the organizational partners is implemented in a hierarchical manner (method 3), and upper level decisions in the domain of non-autonomous functions are observed at lower levels ('level' refers here to strategic, tactical and operational). This is not to say that information exchange and feedback is not preceding such hierarchical decisions.

The strategic level interactions define the rules, (or redesign them) and the component agents have high autonomy in the design dialogue. The strategic level dialogue ends with the definition of autonomy domains and non-autonomy domains. The co-ordination starts on the strategic level, where the

domains of autonomous actions are determined for each participant and the interfaces for communication and co-operation get defined. This sets then the environment for co-operation at all other levels. The tactical and operational levels will interact obeying those rules.

The planning and operational levels are working within this framework, the interfaces between participants comply with the settings defined on the strategic level. Mechanisms are provided for feedback to the level above, so the efficiency of the system can be continuously monitored. Policies (i.e. the rules) are set at the strategic level, and they can be modified if the system's behavior deviates from normal significantly. The feedback mechanisms can be used to modify the structure of the whole framework, adjust interfaces or domains of autonomous actions, etc., in order to improve the system's overall performance.

The system is organized in a hierarchical manner, meaning that at any one time when a decision is taken the co-ordinating agent (via method 3) can determine the contents of the decisional objectives passed down to the lower level agent. However, the co-ordinating agent's tactical level does not have authority to change the decisional structure, so this hierarchical relationship (and loss of autonomy) is limited in scope.

The more the tactical decision frameworks encroach on the lower level's autonomy the more the amount of communication between the levels. This additional load put on the communication system will eventually impact on overall system performance. The functional responsiveness of the system may be slowed down, as the agreements take time to evolve and the upper, decision making level receives the information through a preliminary filtering of the lower ones.

Designers of virtual enterprises must be able to model the dynamics of decision making and the efficiency of responsiveness of the different design variants.

Evaluating this architecture from the enterprise modeling aspect the system has the advantage of having a clear, well-organized and easily traceable structure. Functions at all levels are distinctly defined, the interfaces are handled without ambiguity. As we stated earlier, agents are providing solutions that satisfy certain criteria, but are not expected to provide strictly optimal solutions. This can lead to performance deficiencies. Each level has certain autonomy, but its degree varies at different levels. The higher level entity does not have in fact full autonomy, because it has responsibility for the lower level, e.g. it must observe constraints that are legitimately (according to the rules) submitted as information by the lower level agents.

The lesson from this analysis is that the protocols which must be set up as a part of the design of the enterprise must be suitable for the definition of authority and obligation, and responsibility, in a formal enough manner, so that the resulting set of beliefs, responsibilities,

For example, the decision framework set by the higher level restricts the lower levels' autonomy. At the same time, if the coupling between the higher and lower levels is not close enough, then the limited autonomy of the lower levels can result in conflicts, or lack of harmony, between the higher and the lower levels' objectives. This distance between the decisional centers can be a serious issue, and indicates the need for tighter coupling, albeit curtailing lower level's autonomy. In fact, a reasonable balance between tight coupling and local autonomy can be easier to establish than trying to maintain local autonomy at all costs and going through several steps of iteration to achieve an acceptable outcome.

3.3 Project enterprise

Project enterprises are typical forms of virtual enterprise, used in one-of-a-kind production or other one-of-a-kind human endeavors. Project enterprises are built on the basis of the knowledge of the life-cycle of the product the design and, or building of which is the mission of the enterprise.

A particular property of project enterprises is that the management structure is designed and operated before the rest of the enterprise is set up. Therefore the project enterprise is self-building in nature. It is not necessarily adaptive (the project enterprise is usually built after the pattern of earlier projects used as blueprints), because adaptivity assumes that the pattern itself is capable of being changed by the enterprise itself in response to changes in the environment.

However, the dynamics of the organization and its adaptability is a system property which would be of interest to investigate by designers of project enterprise.

Unlike in virtual enterprises, which are created by breaking up large organizations into autonomous parts to form a better quality value chain, virtual project enterprises are assembled according to a unique business process tailored to the particular product or service in question. The individual entities (partners, or agents) can usually be selected on the basis of competencies, geographic location, and other characteristics, including price of service or product, strategic alliance possibilities etc. In project enterprises often hundreds of participants must be co-ordinated, but the selection of potential partners may be quite limited for any particular service and product to be made available at the time, in the desired quality and at the desired location. In addition, the time pressure on enterprise design decisions is usually great, either because it is part of a tendering process with short deadlines, or because it is part of the project itself. Structurally, this type of brokerage is simple to represent, but *protocols should be defined (definable) and an ability to create a system of management of resource types is needed.*

For large projects to run smoothly it is necessary to have a clearly delineated responsibility structure and protocols are needed to set up and follow these.

design standard interfaces between the respective levels of participants to ease the burden of negotiations by project management and make it more automated.

On the strategic level, commitments are sought and obtained with the precision of usual planning negotiations. It is necessary to be able to prove of any virtual enterprise model that the types of information necessary for parallel distributed planning will be available through the decisional frameworks passed among the partners. For this to be a viable analysis possibility *an ontology of distributed parallel planning should be defined and be made available for the enterprise modeller.*

Similarly the ability of the virtual enterprise to carry out parallel distributed scheduling necessitates the definition of information requirements (and interfaces through which to supply them).

Finally, operational level interfaces could be analyzed as for their ability to support an enterprise-wide monitoring of state.

The ontology (i.e. the formal definition) of concepts which must be supported by the protocol languages could be derived from studying various parallel distributed planning, scheduling and control algorithms.

4. MODELLING NEEDS

4.1 Conflicts in the decisional model

The decisional subsystem of an enterprise implements the goal seeking behavior. It is navigating the enterprise within the constraints of the environment while trying to pursue an enterprise level goal, and adjust and react to any changes in this set-up. An ideal decision making structure would work in this framework for the sole benefit of the organization, and would do all to find the optimum solution in any situation. In practice, however, finding or even defining the optimum can be a too time consuming or sometimes impossible task. Therefore, the participating agents are usually not required to produce optimum decisions: if the outcome satisfies a set of criteria, it is accepted.

Another, even more important limitation is the possible conflict of interest between the decision making system and its components. These factors indicate that any decision structure needs to be examined for efficacy and conflict of interests. As decision making is performed by a system of individuals, the objectives and striving of the human as an individual can not be neglected. Decisional roles taken by an individual may be in conflict with a) the individual's role in another part of the system, b) with the individual's personal aspirations, or c) with the interests of another individual in the same organization. The same is true of group interests and aspirations.

This can be done in a formal manner, once appropriate tools or frameworks are set up for verification of the decisional structure.

When two or more enterprises are combining their efforts for a common project, a new type of conflict emerges: that between the organizational structures. The common goal is supposed to be the main driving force for the participants (with regards to the joint action), but the interests of individual companies may interfere with that of another participant.

It is necessary to be able to demonstrate that no conflict is present between the interest of the virtual enterprise as a whole and that of its components, and if conflicts arise, as they eventually do, that there is suitable conflict resolution path provided for the virtual enterprise to resolve them.

The harmonization of the mental models and beliefs of the individuals (e.g. the assumptions on trust, or what is the 'way things are done here') makes up a company culture and precisely by developing and explicitly displaying such mental models and beliefs can a company culture be ingrained. When a virtual enterprise is set up, partners will have slightly (or extremely) different cultures, so the harmonization of the mental models and beliefs to create a culture of the virtual enterprise needs special attention.

In virtual enterprises where participants change and structure is never constant, it is necessary to be able to explicitly represent agreements and common beliefs that will govern interaction on the lower levels. This is an important part of the 'enterprise engineering process'.

It is expected that autonomy, responsibilities, obligations and beliefs should be able to be analyzed from a structural point of view to reveal conflicts among decisional roles and conflicts that arise from the way they are assigned to organizational entities.

Figure 2 shows an example how the decisional roles can be represented using a GRAI Grid, while Figure 3 shows a mapping of organizational entities to decisional roles.

Figure 2 Decision Roles Represented in GRAI Grid

4.2 Autonomy

A virtual enterprise is intended to work for the benefit of its participants, while endeavoring to harmonize the individual interests. This requires the co-operation of autonomous systems in a conflict-prone environment. Considering each participant in the virtual enterprise to be an agent, there is a need to co-ordinate the autonomous behavior of these agents. Various approaches to this problem are:

1. Agents make decisions in their own environment, detect and try to resolve conflicts locally when implementing those decisions;
2. Agents interact with one another and make decisions in co-operation with the others;
3. A dedicated agent is assigned to representing the interests of the whole (virtual) enterprise and co-ordinates the activities and decisions of the others.

Co-operation of autonomous entities always requires careful consideration. In optimistic control where only individual goals are pursued in the hope that there will be no conflict in most of the cases, the conflicts can become unresolvable when they do occur. It is very important to define the domain where autonomous decisions should take place, and clearly identify the points where harmonization of individual interests must occur. In other words:

Autonomy is assumed only in a well-defined domain of functions (decisional or operational), and it is limited both in space and in time. The models of virtual enterprise thus must take into account the need for explicit conflict resolution, as a type of co-ordination activity.

Figure 3 Organizational Entities and Decisional Roles

5 CONCLUSION

Virtual enterprises are becoming increasingly popular for addressing and solving special problems, for organizing common projects or for joint production. To understand their operation and suggest efficiency improvements, the methods of modeling virtual enterprises were examined in the paper. We identified a number of requirements that virtual enterprise models need to satisfy. The requirements were based on real-life problem cases encountered in the Globeman consortium. In particular, we identified

- autonomy at all levels
- comprehensive conflict handling and resolution and
- agent-like behavior of components

as key issues. In future, these requirements will have to be translated into technical terms, so that the expressive power of the proposed enterprise modeling languages can be evaluated, and necessary adjustments can be made.

6 REFERENCES

- Balasubramanian, S.; Maturana, F.P.; Norrie, D.H. (1997), Multi-agent planning and co-ordination for distributed concurrent *engineering Int.J. of Coop. Inf. Sys.*, 5 (2-3) pp. 153-79
- Bernus, P.; Nemes, L. (1996) A framework to define a generic enterprise reference architecture and methodology, *Computer-Integrated Manuf. Systems*, vol.9, no.3, pp. 179-91

- Deen, S.M.(1993) Co-operation issues in holonic manufacturing systems, *IFIP Tr.B (Appl.inTechn.)*,B-14, pp.401-12
- Doumeings, G.; Fenie, P.; Chen, D. (1994), GIM (Grai Integrated Methodology): a methodology to specify and to design advanced manufacturing systems, Proc. ILCE '94. Integr. Logistics and Conc. Engineering, pp. 346, 271-80
- Erman, L.D.; Hayes-Roth, F.; Lesser, V.R.; Raj Reddy, D. (1980) The Hearsay-II speech-understanding system: integrating knowledge to resolve uncertainty *Computing Surveys*, 12 (2) pp. 213-55
- Kahn, H.J. (1995) STEP methodology – an overview, *Object Technology and its Appl. in Engng.*,p. v+178, 66-75
- Lawson, J.W. (1986) A quick look at matrix organization from the perspective of the practicing manager *Engineering Management International*, 4(1) pp.61-70
- Malone, T.W.; Crowston, K.; Jintae Lee; Pentland, B. (1993) Tools for inventing organizations: toward a handbook of organizational processes, Proc. 2nd Worksh. on Enabling Technologies Infrastructure for Collab. Enterp.pp.72-82
- Mathews, J. (1995) foundations of intelligent manufacturing systems: the holonic viewpoint *Comp. Integr. Manuf. Syst.* 8 (4) pp. 237-43
- Minsky, Marvin Lee (1996), First person : the society of mind, Voyager Company.
- O'Leary, D.; Kuokka, D.; Plant, R. (1997) Artificial Intelligence and Virtual Organizations, *Communications of the ACM* vol.40 /1 pp.52-59
- Schmidt, K.; Simone, C. (1996) Co-ordination mechanisms: towards a conceptual foundation of CSCW systems design, *J CSCW* 5(2-3) pp. 155-200
- Tamura, S.; Luh, P.B.; Oblak, J.M.; Watanabe, S., (1995) A planning and scheduling architecture for holonic manufacturing systems, Proc 1st World Congr. on Intell. Manuf. Processes and Sys., vol.2 pp790-801.
- Vernadat, F.B. (1994) Business process and enterprise activity modeling: CIMOSA contribution to a general enterprise reference architecture and methodology (GERAM), ICARCV '94 vol.1 pp. 378-82
- Vernadat, F.; Drira, K.; Azema, (1996) P. An integrated description technique for distributed co-operative applications. CESA'96 IMACS. *Comp. Eng. in Sys. Applications*, pp. 794, 608-12
- Wall, W.C., Jr. (1984) Integrated management in matrix organization *IEEE Trans. on Eng. Mgmt*, 31 (1) pp. 30-6
- Williams, T.J. (1994) The Purdue Enterprise Reference Architecture *Computers in Industry*, 24, (2-3) pp. 141-58
- Winograd, T; Flores, (1990) Understanding computers and cognition: a new foundation for design, Addison-Wesley