e-Work: the challenge of the next generation ERP systems

PORNTHEP ANUSSORNNITISARN and SHIMON Y. NOF

Keywords Active protocols, autonomous agent, collaborative information systems, coordination modelling and analysis, computer-supported conflict resolution, enterprise resource planning, task administration protocols, teamwork integration evaluation, viability

Abstract. A rapid development in information technology has transformed not only the way people work and interact with each other, but also the focus of management, from within the enterprise to the relations among business partners. Electronic media, such as e-mail, LAN, intranet and Internet, enable tools that help the enterprise to collaborate on its work and missions within its own organization and with other independent enterprises, including suppliers, customers and even competitors. The work activities that are related to these electronic media are defined as e-Work. This article addresses the emerging and future problems of ERP systems in both the development and implementation perspectives. The characteristics of the next

generation of ERP systems are presented based on conclusions from several of Purdue University's PRISM Center research projects in planning distributed information and production systems.

1. The changing scope of ERP

Over the past decade, there has been a shift in focus from business functions to business processes and further, into value chains. Conventionally, business operations were organized around functions (e.g. accounting, inventory and production) as were traditional information systems. Nowadays, enterprises focus on the effectiveness of the operation that requires functions to be combined to achieve end-to-end business processes (Huang and Nof

Pornthep Anussornnitisarn and Shimon Y. Nof, PRISM Center – Production, Robotics, Integration, Software/Systems for Manufacturing and Management, School of Industrial Engineering, Purdue University, West Lafayette, IN, 47907-1287, USA. E-mail: nof@purdue.edu

PORNTHEP ANUSSORNNITISARN was born in Bangkok, Thailand, and received his Bachelor of Engineering in Electrical Engineering from King Mongkut Institute of Technology, Bangkok, Thailand in 1990. He was awarded a Master of Science in Industrial Engineering from Ohio University in 1995. In August 1998, he began his PhD study at the School of Industrial Engineering, Purdue University. At Purdue University, he joined the Production, Robotics and Integration Software for Manufacturing and Management (PRISM) programme as Research Assistant. His research has involved collaboration-based information technology, partially funded by IBM. Upon completion of his PhD degree, he joined the Department of Industrial Engineering at Kasetsart University, Bangkok, Thailand, as a full-time faculty member.

SHIMON Y. NOF, Professor of Industrial Engineering at Purdue University, has held visiting positions at MIT and universities in Chile, EU, Hong Kong, Israel, Japan, and Mexico. Director of the NSF-industry-supported PRISM Center for Production, Robotics and Integration Software for Manufacturing and Management; he is a Fellow of IIE, Secretary General of IFPR, and current Chair of IFAC CC – Manufacturing and Logistics Systems. He has published over 250 articles on production engineering and information/robotics engineering and management, and is the author/editor of nine books in these areas. In 1999 he was elected to the Purdue Book of Great Teachers, and in 2002 he was awarded the Engelberger Medal for Robotics Education. Professor Nof has also had over eight years of experience in industry positions.

2000, Huang *et al.* 2000). As a result, the ERP system has to adapt itself to support business processes. The business processes, which have been viewed as activities within an organization, have become the processes across the enterprise boundaries, over the value chain. For example, a supply chain or network will include all of the raw material suppliers, manufacturers, distributors and customers. Each supply partner will have its own business processes which interact with the others' on the value chain or network. When a situation occurs, such as a change in product specification, all the related partners may be affected and therefore need to react to that situation promptly and harmoniously. Hence, individual partners must interact cooperatively with one another to ensure the maximum performance effectiveness throughout the chain (Eberts and Nof 1993). The collaboration type of business activities plays an increasingly important role in controlling and maximizing the respective value for each enterprise. Currently, ERP systems are still designed to mainly support internal business processes. Even though several vendors have added computer communication and Internet capabilities to their services, the issues and methods to achieve efficient collaboration across enterprises have not been solved effectively. One way to improve the efficiency of interaction among involved partners is to achieve and maximize the automation of the business collaboration workflow (Kim and Nof 1998, 2000). This automated collaboration activity is defined as a key element of e-Work (Nof 1999b, 2000a, b). The purpose of this article is to analyse the implications of e-Work requirements to the design of future ERP systems. Based on PRISM Center research on distributed production information systems over the last decade (Huang and Nof 1998, Nof and Huang 1999, Nof 1999, 2000a,b, Rajan and Nof 1996, Sauvare *et al.* 1998, Williams and Nof 1998, 1999), the development of

the next generation ERP systems should focus on five main issues, as described next.

1.1. Issue 1: Broader business-process-oriented models and application

Most of the ERP systems today still focus on the function of business activity such as inventory, production, accounting, etc. Each functional view in an ERP system poorly represents the interaction with and among other functional views. As a result, the company has to invest significantly in training its personnel on how to use the system, how to coordinate their task in the system, and how to interact with peers. Many companies have to modify their current business activity/flow in order to use the ERP system. Retraining for new software versions is another challenge. As mentioned earlier, information technology such as intranet and Internet has expanded the boundary of task coordination from the enterprise level to the multi-enterprise level. A task may no longer be completed within a local enterprise (Etzion *et al.* 1995). In order to achieve efficient coordination among independent enterprises, training by itself cannot achieve this goal, because it is practically impossible to train and retrain people from different organizations at the same time with the same objective. The next generation of ERP systems needs to have *activity flow view* of business processes (BP) to control and monitor the entire relevant workflow, including the relevant activities that occur outside of the local enterprise. The activity flow view will not only provide better representation of task coordination, but will also help users to focus on their tasks and related activity in the context of tasks performed by peers (figure 1).

Figure 1. An architecture for a BP-oriented application.

1.2. Issue 2: Temporary cooperation

In distributed and independent business alliances, each individual enterprise has its own priorities, policies and objectives. In order to form a coalition among independent enterprises, each individual enterprise should at least achieve the minimal satisfactory level of its objectives. The concept of *enterprise networking* has been established for several decades; for example, several tiers of suppliers in the automobile industry. In most cases, forming a coalition has taken several months or years, and has required extensive negotiations. Recently, the Internet has become a more convenient communication platform among enterprises. The network of enterprises can be formed temporarily (at least in part) by exchanging the necessary information over the computer network, intranet or Internet (Ceroni and Nof 1999, Ceroni *et al.* 1999, Khanna *et al.* 1998). As a result, the 'virtual enterprises' terminology has been developed for a network of mainly independent, substitutable enterprises that integrate their skills to perform a certain kind of temporary, project-dependent specific task. When there is no shared task, or one or more of the participants are no longer interested in this cooperation, this particular enterprise network can be dissolved, or new participants who can provide the missing skills are added into the network.

Because the flexible (agile) structure of enterprise network is dynamically changing all the time, certain rules or policies may not be established as a permanent coalition guideline. Hence, the challenge for the next generation of ERP systems is: (1) how to formulate the network of enterprises to achieve optimal benefits and (2) how to define and measure the satisfactory performance level of the individual enterprise, which has a dynamic behaviour depending on the market environment (Barad and Nof 1997, Huang and Nof 1999, 2000).

1.3. Issue 3: Standard, and customized coordination protocols

In order to perform successful e-Work, meaning, communicate and cooperate effectively in the Internet/intranet domain (Nof 1999b, 2000a), a certain method/procedure called a 'coordination protocol' must be established. It will guide and support the interaction of participants (human and software agents) in order to successfully accomplish the necessary tasks/services. Accordingly, coordination protocols will be needed to ensure the smoothness and efficiency of the interaction, collaboration and communication (Esfarjani and Nof 1998). Protocols provide fundamental services for integrating high-level tasks/services in a large distributed organization with local autonomy for the individual system platform and process (Anussornnitisarn *et al.*

2002, Peralta *et al.* 2002). Because the coordination process can be considered as a major problem in many disciplines, such as management science, computer science and artificial intelligence, the common research objectives among these disciplines are to identify and to study a basic process involved in the coordination. In fact, the ultimate goal of coordination-related research is to find the most effective coordination scheme that maximizes collaboration effectiveness among participants (e.g. human, machine, computer, etc.). Coordination, in general, is an ill-defined objective due to the problem of heterogeneous, distributed and autonomous platforms, tasks and participants. As a result, a generalized coordination model and scheme do not exist (Anussornnitisarn and Nof 2000).

For the next generation of ERP systems, a group of coordination protocols, some standard and some customized, must be identified as a basic set that can be used to effectively automate the coordination process among independent enterprises.

1.4. Issue 4: Coordination modelling and analysis

The distributed environment limits enterprises in attaining operational efficiency goals (Nof 1995, 1999a). Similar to a multi-agent system (MAS), the independent entities need to exchange information through a communication channel in order to initiate and sustain collaboration. The collaboration of distributed entities can be handled by either a centralized control approach, or a decentralized control approach (Papastavrou and Nof 1996). Because each distributed system is an independent entity, their actions must be coordinated to achieve global solutions and objectives. Typically, the processing capability of each independent entity and the communication channel are the limited resources within the system. Therefore, the flow of information and all agent activities must be synchronized in order to achieve the globally consistent and coherent solution to the operational problems. Inefficient flow of information will certainly not only increase the communication cost but also reduce the agility of the whole network (Ceroni and Nof 1997, 1999). The cost that is incurred during the coordination process can be defined as *coordination cost*. Research in the area of coordination modelling and analysis seeks to discover the best coordination structure that minimizes coordination cost while affording scalability and fulfilling task objectives (Chen and Nof 2000, Chen 2002).

1.5. Issue 5: Scalability of the collaboration network

The multi-ERP problem domain consists of multiple objective functions for each individual enterprise and

network of enterprises. A coordination scheme may or may not be able to resolve, or better yet, avoid conflicts in a large-scale network. Furthermore, each individual enterprise may belong to several supply networks, so that conflicts among members of a network of enterprises occur frequently. The definition of scalability in the collaboration network means that the conflicts in the system are minimized and sustained even when the number of enterprises in the network increases. To provide scalability and robustness for the multi-ERP system, the conflicts within/between enterprises must be prevented or resolved. In analogy to a teamwork environment, many researchers suggested that there is an optimum number of team members for the best performance. More team members do not necessarily contribute to success. The reason is that with more members, the volume of potential conflicts – e.g. conflicting knowledge-based decisions (Lara *et al.* 2000), conflicting time and cost objectives (Auer and Nof 2000, Auer *et al.* 2000), etc. – increases. The optimum team-size can be determined by performance tradeoff analysis between the number of available resources and the number of conflicts. Conflicts must be considered, because they are caused by autonomy, including differences in goals, culture and local expertise. Conflicts can be reduced by improving collaboration among team members. Researchers have proposed several collaboration techniques in terms of ‘coordination’ to avoid conflicts between group members. However, none has verified that in the relatively complex environment, such as multi-ERP systems, the coordination scheme will work effectively.

To design an effective multi-ERP system, the operations and logistics of activities among the network of enterprises need to be defined. As an extension of flexibility, operational and logistics agility can be defined along three dimensions: range, response and collaboration. Range and response are parameters of the internal flexibility of a system, while collaboration defines the capability of a system to seek (offer) support from (to) other systems to accomplish the capacity needs that cannot be fulfilled by each enterprise alone (Huang and Nof 2002). In order to provide scalability and robustness to the multi-ERP system, the conflicts within/between enterprises must be prevented and resolved. Hence, the next generation ERP system needs a conflict management function. Conflict management should not be constrained and delayed by a centralized

approach, but be decentralized, such as in a multi-agent system model. Only if the conflict management system cannot find resolution will human involvement be necessary.

In the next section, specific research results are described, and their implications to the characteristics of ERP functions and the five issues of requirements are discussed.

2. e-Work research at the PRISM Center, and its implication to ERP

e-Work is defined (Nof 1999b, 2000a, b) as collaborative, computer-supported activities and communication-supported operations in highly distributed organizations of humans and/or robots or autonomous systems, and the PRISM lab has investigated fundamental design principles for e-Work effectiveness (Nof and Huang 1999). The premise is that without effective e-Work, the potential of emerging and promising electronic work activities, such as virtual manufacturing and e-Commerce, cannot be fully materialized. A current example is the frustration of users over supply chains with complex ERP and other information systems, originally designed to improve their performance (figure 2).

The following sections describe selected PRISM Center research, which has focused on theories and design principles of e-Work. This research also supports the focus of the next generation ERP systems, as discussed in section 1.

2.1. Viability of autonomous agents (Huang and Nof 1999, 2000, Nof 1999a)

The objective of this study has been to build a platform for future manufacturing systems that are highly distributed, autonomous and collaborative. Based on agent theory, a new model of manufacturing systems, termed agent-based manufacturing systems (ABMS) has been developed. In an ABMS, agents are autonomous and working collaboratively with each other through the regulation of structured dialogues (Task Administration Protocols, TAP). Agents are similar in certain aspects to living systems. Similar to living systems, autonomy and viability of agents have been defined as

Figure 2. The scope of e-Work (Nof 1999b).

Figure 3. A generic model of agent: resources, four agent functions, database (DB), and protocol base (PB) (Huang *et al.* 2000).

new measures for manufacturing and production systems. A theory of agents' coexistence and parasitism has been developed to identify ABMS performance characteristics.

In this research, a complete conceptual structure was constructed, which includes the definition of agent-based manufacturing systems, viability derived both at the agent-level and at the system-level, and four internal functions of an agent (figure 3): internal resource management, a reflexivity mechanism, a goal adjustment mechanism and collaboration management. These four functions enable each agent to interact with other agents and with the environment (modelled by tasks) and to adjust itself to situate in the dynamic environment. This research has developed a parallel simulator called Teamwork Integration Evaluation/Agent (TIE/Agent) (Huang and Nof 2002) for evaluating ABMS. TIE/Agent is programmed with agents on an Intel/Paragon supercomputer. Four computer nodes represent each agent. The information transmission between agents is realized by the Message Passing Interface (MPI) communication library. Agent designers can experiment with TIE/Agent to observe the behaviour of collaborating agents and optimize their design.

The TIE/Agent experiments demonstrate alternative scenarios for applying agents. The results of the experiments indicate, statistically, that:

- (1) processing time of the tasks, rewards associated with the completion of tasks, and centralization or distribution of resources are three significant factors of the system viability;
- (2) inter-arrival time of the tasks and the cost of keeping agents functional are two insignificant factors.

In addition, the experiments indicate that TIE/Agent is an effective tool to study two types of relationships:

- (1) between autonomy factors, including the reflexivity mechanism, goal adjustment mechanism, error recovery, and the system viability; and
- (2) between the number of active agents and the system viability.

Of relevance to future ERP systems, TIE/Agent can also help establish outsourcing policies for an ABMS (Huang and Nof 1998, 1999, 2000, 2002, Huang *et al.* 2000, Kim and Nof 1998, 2000).

2.2. *Coordination protocols* (Anussornnitisarn and Nof 2000, Anussornnitisarn *et al.* 2002, Peralta *et al.* 2002, Esfarjani and Nof 1998, Huang *et al.* 2000)

The major problem of working in a distributed and autonomous environment, such as virtual enterprises, or extended enterprises and their supply chains is the binding constraints (dependencies) among the autonomous participants. Due to autonomy, the participants perform their activities and tasks according to the tradeoff between the local and global objectives (Huang and Nof 1999, Huang *et al.* 2000). Sporadic conflicts with the constraints occur frequently, even when the information may be shared among the participants. To accomplish some form of coordination, the synchronization of activities must be achieved under the given constraints. According to previous research on coordination protocols, these protocols are a predefined pattern of activities, decision-making and communication, and are used to synchronize activities among the autonomous participants, such as software agents. Coordination can also be defined as 'managing dependencies' (Anussornnitisarn and Nof 2000, chen 2002, Chen and Nof 2000, Malone *et al.* 1999). By characterizing the type of dependencies (Khanna *et al.* 1998), the coordination mechanism that can be best used to manage these dependencies can be identified. The task coordination problem-solving scheme is proposed as follows (figure 4): (1) identifying the particular dependencies of a given task, and (2) considering which alternative coordination protocols can be used to manage task dependencies. If a suitable coordination protocol is designed or selected, the participants are expected to perform their activities more harmoniously and more effectively (Anussornnitisarn *et al.* 2002, Peralta *et al.* 2002, Esfarjani and Nof 1998, Williams and Nof 1998, 1999).

To support the development of specific coordination protocols (particularly, task administration protocols), a generalized coordination protocol is selected as a baseline protocol. Because of its popularity and success in previous

Figure 4. Overview of the task coordination problem-solving scheme (Anussornnitarn and Nof 2000, Anussornnitarn 2003).

Figure 5. Conversion of enterprise workflow model (by Petri-net) directly to a Gaussian network (Chen and Nof 2000, Chen 2002).

research, a Contract Net protocol (CNP) (e.g. Sikora and Shaw 1998) is selected as a baseline protocol for comparison with certain task administration protocols. In a case study (Anussornnitarn and Nof 2000, Anussornnitarn 2003), a resource allocation protocol (RAP) has been tested against a CNP. Unlike previous research that mainly focused on the logical aspects of negotiation protocols, this investigation has focused on analysing the dynamic performance properties and behaviour of coordination in a real-time domain. In addition, the focus of previous research was on the performance of decentralized control systems in which CNP was used as a common coordination protocol, against the centralized control approach (traditional approach). Despite the fact that the decentralized approach may often (although not always—see Ceroni *et al.* 1999) provide a better result over the centralized approach, there is no report regarding the coordination efficiency of CNP. As it was found, CNP may not perform well when there are relatively many concurrently active participants for both task managers and resource contractors.

Future ERP systems will require better management of coordination among their distributed, decentralized, users. The coordination protocol that has a coordination

mechanism better matched with the task dependencies (in this case, resource allocation tasks) will perform the coordinated task better than a general or standard protocol (in this case, Contract Net protocol). In addition, the experiments demonstrated scenarios for applying TIE/Protocol, a parallel simulator for evaluating given coordination protocols for a distributed system (Anussornnitarn and Nof 2000, Anussornnitarn 2003).

2.3. Coordination structure analysis (Ceroni *et al.* 1999, Chen and Nof 2000, Chen 2002, Khanna *et al.* 1998)

Gaussian network, which has been applied to describe network topology, has been selected to model enterprise networks because of its two potential advantages: cost reduction and network scalability. Essentially, an enterprise network is a large, distributed system. The enterprise network can be described theoretically by a graph, but since it is different in many aspects from a distributed computing system, for which the Gaussian network was originally developed (Hsu *et al.* 1996), another approach is needed. An enterprise network can be defined (figure 5) by a graph $G = (V, E)$. The node set of V represents the enterprises inside the enterprises' network. The edge set E

Table 1. Comparison of distributed computing and enterprise network.

Comparison aspect	Distributed computing system	Enterprise network
Node in a graph	A computer or CPU with capability of communication	An enterprise with capability to coordinate with other enterprises
Edge in a graph	Communication links between two computers or CPUs	Coordination links between two enterprises
Research issues	Routing algorithms	Coordination among partners

represents the coordination relationship between corresponding nodes.

A comparison between the graphs for a distributed computing system and enterprise network system is shown in table 1. When a Gaussian network is applied to describe the enterprises network, the first question is how to map individual enterprises of an enterprise network into the nodes of the Gaussian cube.

In a Gaussian network, the nodes are used to represent a computer in a network or a CPU in a parallel computer. All of the nodes viewed from the computing point of view are identical or similar. They are also able to perform almost the same operations. Hence, it is not important where those nodes are placed in the developed enterprise network. On the other hand, an enterprise network is quite different. Two issues have to be considered:

- (i) *Coordination capability*: This capability is decided by the degree of coordination defined in subsections 2.2 and 2.3 above. Each enterprise can keep a vector which defines its degree of coordination with other enterprises. Only those with a high degree of enterprise coordination should build up coordination links. The degree of coordination is affected by several factors; it is primarily a function of similarity of production facilities of the multiple enterprises (two in the example of figure 5). It is also a function of similarity of business processes, as well as communication hardware and software available at these networked enterprises.
- (ii) *Coordination relations*: There are two possible directions for an enterprise constructing coordination relationships with other enterprises:
 - *Vertical integration*: The output of one enterprise is the input to another enterprise. When they develop their coordination relationship, they already account for this integration in the cost of coordination. This relationship is relatively long-term and usually can be formed into a relatively more fixed partnership.
 - *Horizontal integration*: Another case is when one enterprise produces as much as what its

customers order. In this case, when maximum capacity is reached, one enterprise might ask another enterprise for assistance. This type of coordination relationship may be temporary and not necessarily predetermined. Usually, the enterprise seeking capacity assistance turns out to be the bottleneck of the enterprise network.

ERP systems need to address both types of coordination relations which can coexist. These relations can be modelled and analysed with a Gaussian cube (Chen 2002, Chen and Nof 2000). An attractive feature of Gaussian cubes and a network is that they inherit the property of scalability. When a bottleneck is formed in an enterprise node, the enterprise network can be reconfigured by adding more edges to the Gaussian cube that the node represents. This property means that the enterprise can ask for coordinated help from other enterprises in case of urgency. Based on pilot experiments, results show that under certain conditions, the coordination cost can be compensated by the profit earned from lower vulnerability cost. The next step is to extend the experiment to more enterprises to test the scalability of enterprise networks represented by Gaussian cubes.

2.4. *Computer-supported conflict resolution in distributed ERP systems* (Avila-Soria and Nof 2000, Lara and Nof 1996a, b, Nof and Huang 1999, Nof and Ceroni, Forthcoming, Papastavrou and Nof 1996, Prytz *et al.* 1995)

As mentioned in subsection 1.5, conflicts often occur during the collaboration among autonomous members of a distributed organization, regardless of the types of available communication channels. By design, information technology does not intend to directly solve the conflict, but can be used to expedite the conflict resolution process. To resolve conflicts in distributed systems would require well-defined interactions and an information exchange scheme, as discussed above on the coordination protocol, subsection 2.2.

Figure 6. A model of reactive problem-solving in manufacturing (Balakrishnan *et al.* 1994).

A possible model of reactive problem-solving in manufacturing is shown in figure 6. In Phase 1, the process begins by monitoring and recognizing the problem that emerges. The problem must then be identified and defined. Following the problem diagnosis, in Phase 2, the causes underlining the problem must be determined. The third phase consists of developing a strategy to resolve the immediate problem and formulating the necessary action plans. The coordination, communication, information sharing and synchronization of decisions and actions are most intense during this phase. Finally, Phase 4, the implementation phase, consists of executing the reactive action plan and reporting the results. The diagnosis, solution and implementation phases generate inputs for root cause analysis and long-term improvement. In addition, implementing the reactive solutions provides the opportunity to learn and improve upon the problem-solving process itself.

In (Huang *et al.* 2000), the impact of a conflict resolution process on the agility of distributed, networked enterprises is investigated. The model in figure 6 may iterate several times until a conflict is resolved. It is assumed that a conflict of the i th need (problem) may be resolved in iteration j with probability s_{ij} and cost C_{ij} . To illustrate the relationship between the probability that a conflict can be resolved, and the number of iterations, table 2 was developed. In the first iteration, there is a probability s_{i1} to resolve a conflict related to requirement i . If the first iteration fails (with a probability $1-s_{i1}$), the second iteration is tried. Then, the probability to reach an agreement is $(1-s_{i1}) * s_{i2}$. Other probabilities can be calculated by following the same approach.

Generalizing Table 2, the probability that the i th conflict can be resolved in no more than m iterations is equal to

$$s_{i1} + \sum_{j=2}^m \left[s_{ij} \left(\prod_{k=1}^{j-1} (1 - s_{ik}) \right) \right]$$

Table 2. Probability to resolve a conflict related to customers' requirements.

Iteration	Probability to resolve a conflict for the i th need
1	s_{i1}
2	$(1-s_{i1}) * s_{i2}$
3	$(1-s_{i1}) * (1-s_{i2}) * s_{i3}$
4	$(1-s_{i1}) * (1-s_{i2}) * (1-s_{i3}) * s_{i4}$
5	$(1-s_{i1}) * (1-s_{i2}) * (1-s_{i3}) * (1-s_{i4}) * s_{i5}$
6	$(1-s_{i1}) * (1-s_{i2}) * (1-s_{i3}) * (1-s_{i4}) * (1-s_{i5}) * s_{i6}$

Suppose ρ_i represent the probability of a conflict occurring about requirement i . The system agility, a , can be presented as shown in equation (1). It is supposed that there are potentially N requirements to be satisfied. In the equation, it is assumed that m iterations are allowed, and n (out of N) customers' requirements are to be met:

$$a = \frac{\sum_{i=1}^n (1 - \rho_i) + \sum_{i=1}^n \rho_i * \left\{ s_{i1} + \sum_{j=2}^m \left[s_{ij} \left(\prod_{k=1}^{j-1} (1 - s_{ik}) \right) \right] \right\}}{N} \quad (1)$$

If a system allows unlimited iterations, the system is assumed to definitely resolve the conflict; this observation leads to a perfect condition $a = n/N$. Unfortunately, the cost cannot be ignored. Suppose every s_{ij} is equal to s , an expected computer-supported conflict resolution cost C_i of the i th requirement can be formulated, as shown in equation (2):

$$C_i = sC_{i1} + (1-s)sC_{i2} + (1-s)^2sC_{i3} + \dots + (1-s)^m sC_{im} \quad (2)$$

A conflict resolution strategy that is efficient and incurs relatively less cost is usually selected as early as possible. Some resolution procedures, however, can be less automatic and less efficient; for instance, when human authorization and approval procedures must be followed.

Figure 7. The cost of conflict resolution increases exponentially when human communications and operations are applied (assuming $q=0.2$); s is the probability of resolution after each iteration (Nof and Huang 1999).

Figure 8. Cost of conflict resolution reaches an upper bound when IT is applied (assuming $q=0.0$); s is the probability of resolution after each iteration (Nof and Huang 1999).

It can therefore be assumed that $C_{ij} = (1+q)C_{i(j-1)}$, where q is a cost increment factor representing such necessary inefficiencies (q is less than 1). Hence, equation (2) can be summarized as equation (3). Equation (3) suggests that $(1+q)$ is a critical factor of C_i , because it may result in an exponential increment of computer-supported conflict resolution cost. On the other hand, because $(1-s)$ is always less than 1, it may result in a convergence; therefore q is a relatively less important factor of C_i .

$$C_i = sC_{i1} \sum_{j=0}^m ((1+q)(1-s))^j \quad (3)$$

For the development of future ERP systems, they have to support conflict resolution. It is noted that if the

iteration of computer-supported conflict resolution is fulfilled by human communications and operations, e.g. phone calls and meetings, q may be a relatively large number close to 1. However, if the iteration is accomplished with the support of information technology, e.g. agent technology, q may be a number closer to 0. This observation can be illustrated by figures 7 and 8. Figure 7 shows that the expected cost of computer-supported conflict resolution may increase significantly when human communications and operations are required, e.g. $q=0.2$. Conversely, figure 8 shows that there is an upper bound in the expected cost, when agent technology is applied, e.g. $q=0$. The comparison of costs under these assumptions indicates that it is recommended to develop information systems support which minimizes q to resolve conflicts.

2.5. Implications from PRISM Center research

According to previous PRISM Center research (summarized in subsection 2.1 to 2.5), the two major ingredients for e-Work effectiveness are *autonomous agents* and *active protocols* (Nof and Huang 1999, Nof 1999b, 2000a, b). Their role is to enable efficient information exchanges at the application level, and administer tasks to ensure smooth, efficient interactions, collaboration and communication to augment the natural human abilities. An *agent* is defined as a computing hardware and/or software system, that is able to autonomously interact with other agents and its environment, act and respond reflexively to external impacts in accordance with its given goals (Nof 1999a). *Active protocols*, in particular, *Task Administration Protocols* (Anussornnitisarn and Nof 2000, Anussornnitisarn 2003, Huang *et al.* 2000) are defined as the logical rules for the workflow control, that enable effective collaboration by communication and resource allocation among production tasks (Nof 1995). 'Active' implies that the protocol can, under coordination logic, trigger and initiate necessary, timely interaction tasks to improve performance. Autonomous agents and active protocols comprise *active middleware*. Using such active middleware will alleviate the conflicts, which occur naturally during the collaboration among autonomous business partners of a multi-ERP system. A conceptual architecture of the next generation ERP systems is proposed and discussed in the next section.

3. The design of ERP systems for e-Work interactions and environments

Recently, enterprise architectures have become relatively more heterarchical than hierarchical. The heterarchical enterprise can provide greater levels of reconfigurability and adaptability in response to competitive, changing markets. These organizations are comprised of multiple, distributed, locally autonomous entities, thus enabling a cooperative approach for global

decision-making. The distributed enterprises allocate their tasks by matching skills and resources from different, autonomous enterprises and synthesizing them as a unified business alliance. The effectiveness of this cooperation depends on how each individual company coordinates its tasks with other parties. Information systems are the backbone of this cooperation. The next generation of ERP systems can be described as a combination, coordination and integration of information resources to plan and serve the requirements of both users and developers. The ERP architecture provides support for distributed computing in the context of large-scale networked, heterogeneous information systems. There are five basic issues in the design and implementation of any coordinated problem-solver system (Gasser 1994), which can also be considered as key issues in the development of the next generation ERP systems:

- (1) How to formulate, describe, decompose and allocate problems and synthesize results among groups of independent entities, such as human, software agent and enterprise.
- (2) How to enable applications to communicate and interact: what information exchange protocols to be used, and what and when to communicate.
- (3) How to ensure that independent entities act coherently in making decisions or taking action, accommodating the non-local effects of local decisions, and avoiding harmful interactions.
- (4) How to enable individual independent entities to represent and reason about the actions, plans and knowledge of other agents, in order to coordinate with them.
- (5) How to recognize and reconcile disparate viewpoints and conflicting intentions among independent entities, and try to coordinate their actions.

Based on recent PRISM Center research projects and other research in computer-supported cooperative work, six major components of the next generation ERP can be identified (figure 9).

Figure 9. Conceptual architecture of the next generation ERP system.

The six major components that will optimize the performance of e-Work interactions in heterogeneous, autonomous and distributed environments (HAD), are as follows:

- (1) *Multi-agent based system* (MAS): Agent technology brings proactive, adaptable functions into ERP, rendering it active.
- (2) *Workflow management system* (WFMS): Integrates and automates the execution of steps that comprise a complex process, e.g. a business process. By using workflow techniques, ERP is able to control and execute the tasks beyond an enterprise boundary and throughout the distributed enterprises.
- (3) *Coordination protocols*: The coordination mechanism that provides specific work steps and roles of participants. Cooperative work protocols are usually designed as software code, in order to automate the workflow system.
- (4) *Decision support systems* (DSS): DSS help software agents to tackle the complicated tasks such as conflict resolution. They also improve the agents' ability to solve unexpected problems.
- (5) *Modelling language/tools*: Used to model the enterprise processes and its specifications. In addition, the business process modelling can be used as a re-engineering tool for optimizing certain processes and workflow.
- (6) *Tasks and activity databases*: Provide task details and activity specification for process modelling.

4. Conclusion and future research

The way people and systems operate, work, communicate and collaborate is being transformed by e-Work. Hence, understanding how to manage and plan effective e-Work is fundamental to productivity and competitiveness. Models of e-Work have been developed to address issues from interaction to coordination, conflict resolution and the management of complex e-Work production environments. Several tools and methods have been developed for this modelling, with the objective of understanding the emerging requirements, limitations and capabilities of e-Work. The modelling challenges include enhancements of the TIE models to include further e-Work functions, which depend on future Internet, intranet and similar technologies. For the next generation ERP system, it is necessary that the focus is on how the ERP application can become an effective tool for the large-scale collaborative work under an e-Work environment. From this perspective, the following research questions are still open to be explored:

- (1) How to simulate and evaluate coordination interactions in a large-scale environment, such as the Internet: what are the measurements that

effectively represent dynamic behaviour of the coordination performance and viability of an individual enterprise and of an overall network of enterprises.

- (2) Managing growth arbitrariness: Growth arbitrariness in a multi-ERP system, alliances and activities may be needed for economic effectiveness, but is prone to errors and conflicts. Modelling and control of scalability for arbitrary growth raises the questions: How to measure it? What are the key factors to enable or disable it? What are the limits of growth, if any?
- (3) Handling errors and conflicts in a multi-ERP system: Errors and conflicts are viewed as major obstacles for interactions and growth of the multi-ERP systems and limit the systems' quality of service. What models of error and conflict detection and resolution (ECCR) are feasible, and what are their characteristics, effectiveness and relative costs? With heterogeneous ERP systems across organizations, what tools for coordination and control of distributed ECCR are needed, if any, and what are their feasibility models, effectiveness and costs?

Acknowledgements

This research has been partially supported by the PRISM Center and by the IBM Corporation. The authors also acknowledge the many contributions by colleagues and research students working at the PRISM Center.

References

- ANUSSORNITISARN, P., 2003, Design of active middleware protocols for coordination of distributed resources. PhD dissertation, Purdue University, West Lafayette, IN.
- ANUSSORNITISARN, P., and NOF, S. Y., 2000, TIE protocol, a Teamwork Integration Evaluator for coordination protocols. *Proceedings of ICPR-2000*, 2–4 August, Bangkok, Thailand.
- ANUSSORNITISARN, P., PERALTA, J. A., and NOF, S. Y., Time-out protocol for task allocation in multi-agent systems. *Journal of Intelligent Manufacturing*, **13**(6), 2002, 511–522.
- AUER, J. E., and NOF, S. Y., 2000, Agent-based prediction of customer requirements for distributed team service systems. *Research Memo 2000-8*, School of Industrial Engineering, Purdue University.
- AUER, J. E., HUANG, C. Y., NOF, S. Y., 2000, Simulation software to model distributed service systems. *Research Memo 2000-5*, School of Industrial Engineering, Purdue University.
- AVILA-SORIA, J., and NOF, S. Y., 2000, Interactive error recovery for robotic assembly using a neural-fuzzy approach. *Research Memo 2000-9*, School of Industrial Engineering, Purdue University.

- BALAKRISHNAN, A., KALAKOTA, R., and OW, P. S., 1994, Designing collaborative systems to support reactive problem solving in manufacturing. S.Y. Nof (ed) *Information and Collaboration Models of Integration* (Dordrecht: Kluwer), 105–134.
- BARAD, M., and NOF, S. Y., 1997, CIM flexibility measures: a review and a framework for analysis and applicability assessment. *International Journal of Computer Integrated Manufacturing*, **10**(1–4), 296–308.
- CERONI, J. A., and NOF, S. Y., 1997, Planning effective parallelism in production operations. *Research Memo 97-10*, School of Industrial Engineering, Purdue University. Published: CERONI, J. A., and NOF, S. Y., 2002, A workflow model based on parallelism for distributed organizations. *Journal of intelligent Manufacturing*, **13**(6), 439–461.
- CERONI, J. A., and NOF, S. Y., 1999, Planning integrated enterprise production operations with parallelism. *Proceedings of ICPR*, Limerick, Ireland, 457–460.
- CERONI, J. A., MATSUI, M., and NOF, S. Y., 1999, Communication based coordination modelling in distributed manufacturing systems. *International Journal of Production Economics*, **60–61**, 281–287.
- CHEN, J., 2002, Modelling and analysis of coordination for multi-enterprise networks. PhD dissertation, Purdue University, West Lafayette, IN.
- CHEN, J., and NOF, S. Y., 2000, Multi-enterprise networking. *Proceedings of the International Conference on Manufacturing Systems: Innovations for the 21st Century*, Ann Arbor, MI, July.
- EBERTS, R. E., and NOF, S. Y., 1993, Distributed planning of collaborative production. *International Journal of Advanced Manufacturing Technology*, **8**, 258–268.
- ESFARJANI, K., and NOF, S. Y., 1998, A client-server model for integration and collaboration in production testing. *International Journal of Production Research*, **36**, 3295–3321.
- ETZION, O., DORI, D., and NOF, S. Y., 1995, Active coordination of CIM multi-database system. *International Journal of Computer Integrated Manufacturing*, **8**, 116–125.
- GASSER, L., 1994, Information and collaboration from a social/organizational perspective. In S. Y. Nof (ed.), *Information and Collaboration Models of Integration* (Dordrecht: Kluwer), 237–261.
- HSU, W. J., CHUNG, M. J., and HU, Z., 1996, Gaussian networks for scalable distributed systems. *Computer Journal*, **39**(5), 417–426.
- HUANG, C. Y., and NOF, S. Y., 1998, Development of integrated models for material flow design and control – a tool perspective. *Robotics and CIM*, **14**, 441–454.
- HUANG, C. Y., and NOF, S. Y., 1999, Viability – a measure for agent-based manufacturing systems. *Proceedings of ICPR*, Limerick, Ireland, 1869–1872. Also in *Research Memo 99-11*, School of Industrial Engineering, Purdue University. Also HUANG, C. Y., and NOF, S. Y., 2000, Autonomy and viability – measures for agent-based manufacturing systems. *International Journal of Production Research*, **38**(7).
- HUANG, C. Y., and NOF, S. Y., 2000, Formation of autonomous agent networks for manufacturing systems. *International Journal of Production Research*, **38**(3), 607–624.
- HUANG, C. Y., and NOF, S. Y., 2002, Evaluation of agent-based manufacturing systems based on a parallel simulator. *Computer and Industrial Engineering*, **43**, 529–552.
- HUANG, C. Y., CERONI, J. A., and NOF, S. Y., 2000, Agility of networked enterprises – parallelism, error recovery and conflict resolution. *Computers in Industry*, **42**, 275–287.
- KHANNA, N., FORTES, J. A. B., and NOF, S. Y., 1998, A formalism to structure and parallelize the integration of cooperative engineering design tasks. *IEE Transactions*, **30**(1), 1–15.
- KIM, C. O., and NOF, S. Y., 1998, Coordination and integration models for distributed and heterogeneous CIM information. *International Journal of Industrial Engineering Applications and Practice*, **5**(1), 68–77.
- KIM, C. O., and NOF, S. Y., 2000, Investigation of PVM for the emulation and simulation of a distributed CIM workflow system. *International Journal of Computer Integrated Manufacturing*, **1**(5), 401–409.
- LARA, M. A., and NOF, S. Y., 1999a, Computer-supported conflict resolution in collaborative facility design. *Research Memo 99-21*. Also in *International Journal of Production Research*, 2003, **41**(2), 207–234.
- LARA, M. A., and NOF, S. Y., 1999b, Systematic resolution of conflicts situations in collaborative facility design. *Research Memo 99-23*, School of Industrial Engineering, Purdue University.
- LARA, M. A., WITZGERMAN, J. P., and NOF, S. Y., 2000, FDL, a computer-based tool for collaborative facility design. *International Journal of Production Research*, **38**(1), 2471–2488.
- MALONE, T. W., et al., 1999, Tools for inventing organizations: toward a handbook of organizational processes. *Management Science*, **45**(3), 425–43.
- NOF, S. Y., 1995, Parallel and distributed models of integration in production systems. *Proceedings of ICPR*, Jerusalem, Israel, August, 77–79.
- NOF, S. Y., 1999a, Intelligent collaborative agents. In *McGraw-Hill Yearbook of Science and Technology – 2000* (New York: McGraw-Hill), 219–223. Also *Research Memo 99-8*, School of Industrial Engineering Purdue University.
- NOF, S. Y., 1999b, Modelling of e-Work with Teamwork Integration Evaluation, a teamwork Integration evaluator. *Research Memo 99-20*, School of Industrial Engineering, Purdue University.
- NOF, S. Y., 2000a, Models and tools for e-Work. *Proceeding of the 5th International Conference on AI and Simulation*, Mexico City, February.
- NOF, S. Y., 2000b, Models of e-Work. *Proceedings of the IFAC Symposium on Manufacturing, Modelling, Management and Control*, Rio, Greece, 521–526.
- NOF, S. Y., and CERONI, J. (eds), Forthcoming, *e-Work Models and Systems* (Amsterdam: Elsevier).
- NOF, S. Y., and HUANG, C. Y., 1999, Enterprise agility: a view from the PRISM lab. *International Journal of Agile Management Systems*, **1**(1), 51–59.
- PAPASTAVROU, J. D., and NOF, S. Y., 1996, Analytic procedures for optimizing engineering task integration topologies. *Decision Support Systems*, **17**, 159–182.
- PERALTA, J. A., ANUSSORNITARN, P., and NOF, S. Y., 2002, Analysis of a time-out protocol and its applications in a single server environment. *International Journal of CIM*, **16**(1), 1–13.
- PRYTZ, K., NOF, S. Y., and ROLSTADAS, A., 1995, Manufacturing integration and learning by corporate memory systems (CMS). *Research Memo 95-12*, School of Industrial Engineering, Purdue University.
- RAJAN, V. N., and NOF, S. Y., 1996, Cooperation requirement planning (CRP) for multiprocessors: optimal assignment and execution planning. *Journal of Intelligent and Robotic Systems*, **15**, 419–435.

- SAUVAIRE, P., CERONI, J. A., and NOF, S. Y., 1998, Information management for FMS and non-FMS decision support integration. *International Journal of Industrial Engineering Application and Practice*, **5**(1), 78–87.
- SIKORA, R., and SHAW, M. J., 1998, A multi-agent framework for the coordination and integration of information systems. *Management Science*, **44**(11), S65–S78.
- WILLIAMS, N. P., and NOF, S. Y., 1998, User manual for the TestLAN simulator. *Research Memo 98-6*, School of Industrial Engineering, Purdue University.
- WILLIAMS, N. P., and NOF, S. Y., 1999, Protocol adaptability in TestLAN-based production environments. *Research Memo 99-1*, School of Industrial Engineering, Purdue University. Also: WILLIAMS, N. P., LIU, Y., and NOF, S. Y., 2003, Analysis of workflow protocol adaptability in TestLAN production systems. *IIE Transactions*, **35**, 965–972.

Copyright of Production Planning & Control is the property of Taylor & Francis Ltd and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.