

Service-Oriented Process-Driven Enterprise Cooperative Work with the Combined Rule Strategies

Wen-an Tan, Yun Yang, Zhenhong Lv and Zhonglong Zheng

Institute of Computer Software, Zhejiang Normal University, Jinhua 321004, Zhejiang, P.R. China jk76.zhonglong_jhlzhxch@zjnu.cn

Abstract. The key to enterprise process simulation and process enactment is to implement enterprise process cooperative schedule, i.e., how to control the execution order of the scheduled activities according to cooperative behavior rules under resource constraints. This paper proposes a dynamic PERT/CPM approach using a compound number, and discusses its applications to dynamic enterprise process scheduling for Computer Supported Cooperative Work (CSCW). By using the compound number, activity's duration, the earliest start time, and the latest start time of activities in a process model can be defined and calculated effectively to facilitate the enterprise process flexible scheduling and process forecast during process enactment. A framework to role-oriented process-driven enterprise cooperative work is proposed, some schedule strategies and the algorithm implementation are discussed, as well as the process-driven enterprise application integration.

Keywords: *Enterprise process engine, CSCW, Dynamic PERT/CPM, Process simulation and enactment*

1. INTRODUCTION

It is very difficult to address the issues of enterprise cooperative work using conventional mathematical programming. Service oriented enterprise architecture via enterprise dynamic modeling, focusing on business process modeling, simulation, optimization, and enactment, is the new paradigm [1]. Using business process simulation, computer-aided solutions can be economically obtained to support enterprise cooperative work in enterprise process enactment environment.

Because of the non-deterministic nature of the enterprise business problem, the business process model has two non-deterministic means, i.e., the non-determinism of the process model structure and the non-determinism of model parameters, such as the duration of activity and the probability of product arrived. Therefore, the simulation of the process model is non-deterministic [2].

In order to describe the non-deterministic of the duration of activities in the project network, Fuzzy number is imported, and bring out the PERT (Program Evaluation and Review Technique) which suggests using three different estimates: optimistic, most likely, and pessimistic value [3]. The durations of all activities satisfy β -distribution. It could be calculated as following formula:

$$\text{Duration} = \left[\frac{\text{Optimistic} + 4 \times (\text{most likely}) + \text{Pessimistic}}{6} \right] \quad (1)$$

As same as using CPM (Critical Path Method) to describe the duration time in project network, conventional PERT is still a static method which could not present the uncertainty during the process enactment. Most of process simulation tools usually use rule FCFS (First Come First Serve), HPFS (Highest Priority First Serve) and RS (Radon Selection) and the combined rules to schedule process activities. It is appropriated for the service industries. But for manufacturing industries and engineering project, these are not suitable and enough, especially in the forecast of project process schedule.

In order to present different behavior characters in the business process to support process flexible scheduling for enterprise cooperative work, compound number is proposed to define the non-deterministic of process model, such as duration, the earliest start time and the latest start time of the activities in process models, by according to the definition of fuzzy centre-number and the expression of complex number [4]. A compound number can be defined as the sum of a deterministic number and a non-deterministic number which expression is as $x=a+bi$ [5]. We have proposed a framework of enterprise process cooperative work and developed 8 kinds of process schedule rules algorithms for enterprise modeling. Using these algorithms to define enterprise collaborative model, a kind of combined rules, the customized model can be flexible simulated according to the characteristic of the industrial sector and the enterprise individual requirement. We can use the proposed techniques to support enterprise process dynamic scheduling with computer supported cooperative work environment.

The rest of paper is organized as follows: Section 2 proposes a compound number to define the non-deterministic of process parameter, such as activity's duration, the earliest start time and the latest start time; Section 3 proposes a framework of enterprise process cooperative work and discusses the implementation of enterprise process collaborative scheduling; Section 4 summarizes the paper.

2 COMPOUND NUMBER APPLICATION

2.1 Compound Number Definition

Definition1. Let $x=a+bi$, then x is called a compound number, if a and b is a deterministic number, and i is a non-deterministic real number and $i \in [-1,1]$. i is called compound factor determined by the practice. The following theorems and properties can be proved easily according definition1.

Theorem1. Addition operation rule: Let $x_1=a_1+b_1i$, $x_2=a_2+b_2i$, sum x of x_1 and x_2 is:

$$x = (a_1 + a_2) + (b_1 + b_2) i. \quad (2)$$

Theorem2. Subtraction operation rule: Let $x_1=a_1+b_1i$, $x_2=a_2+b_2i$, then difference x of x_1 and x_2 is:

$$x = (a_1 - a_2) + (b_1+b_2) i. \quad (3)$$

Property1. Upper limit and lower limit: Let $x_1=a_1+b_1i$, $x_2=a_2+b_2i$, ..., $x_n=a_n+b_ni$,

(1) The upper limit of set $\{x_1, x_2, \dots, x_n\}$ is also a compound number $x = a + bi$, which can be defined as:

$$x = \sup(x_1, x_2, \dots, x_n). \quad (4)$$

and satisfying:

$$a = \frac{\max(a_1+b_1, a_2+b_2, \dots, a_n+b_n) + \max(a_1-b_1, a_2-b_2, \dots, a_n-b_n)}{2},$$

$$b = \frac{\max(a_1+b_1, a_2+b_2, \dots, a_n+b_n) - \max(a_1-b_1, a_2-b_2, \dots, a_n-b_n)}{2}$$

(2) The lower limit of set $\{x_1, x_2, \dots, x_n\}$ is also a compound number $x = a + bi$, which can be defined as:

$$x = \inf(x_1, x_2, \dots, x_n). \quad (5)$$

and satisfying:

$$a = \frac{\min(a_1+b_1, a_2+b_2, \dots, a_n+b_n) + \min(a_1-b_1, a_2-b_2, \dots, a_n-b_n)}{2},$$

$$b = \frac{\min(a_1+b_1, a_2+b_2, \dots, a_n+b_n) - \min(a_1-b_1, a_2-b_2, \dots, a_n-b_n)}{2}$$

2.2 Compound Number Applied in PERT/CPM

The duration D_k , the earliest start time E_k and the latest start time L_k of activity A_k can be described with compound number as following formula (6):

$$\begin{aligned} D_k &= a_k + b_k i \\ E_k &= a_{e_k} + b_{e_k} i \\ L_k &= a_{l_k} + b_{l_k} i \end{aligned} \quad (6)$$

Each compound factor “ i ” is a random number valued in $[-1,1]$, which of D_k is created by the random number generator according to the distribution property (e.g., Constant, Uniform, Normal, Poisson distribution) defined in the activity, and i of E_k is calculated by comparing the real time T_{real} to the plan time of E_k .

$$i = \frac{T_{real} - a_{e_k}}{b_{e_k}}. \quad (7)$$

We can use it for the approximate calculation of L_k in order to dynamically control the schedule of activities by the schedule rule defined in the cooperative model.

2.2.1 The Calculation of the Earliest Start Time

(1) Activity A_k in sequential structure

E_k can be obtained by the rule of addition operation and the property of upper limit. If the activity A_k has only one predecessor activity A_{k-1} , the relationship can be illustrated simply as following structure

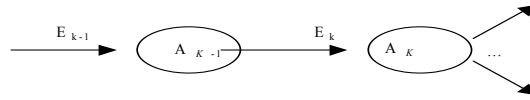


Figure 1. Sequential Structure of Activities A_k and A_{k-1}

E_k of the activity A_k is the sum of the earliest start time E_{k-1} of the activity A_{k-1} and its duration. It can be calculated by following formula:

$$E_k = (a_{e-(k-1)} + a_{(k-1)}) + (b_{e-(k-1)} + b_{(k-1)}) i. \tag{8}$$

(2) Activity A_k in Parallel Structure

Figure 2 illustrated one parallel structure in Process model. If activity A_k has more than one predecessor ($A_{k-1}, A_{k-2}, \dots, A_{k-j}$), the earliest start time E_k of A_k can be calculated as following two steps:

① $E^1_k, E^2_k, \dots, E^j_k$ can be calculated one by one according to the rule of addition operation from the $E_{k-1}, E_{k-2}, \dots, E_{k-j}$ with formula (8);

② E_k of A_k can be calculated according to the property of the upper limit:

$$E_k = \sup(E^1_k, E^2_k, \dots, E^j_k). \tag{9}$$

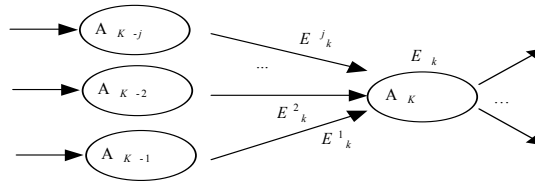


Figure 2. Activity A_k with Multiple Predecessors

2.2.2 The Calculation of the Latest Start Time

We can search out all paths ($path_1, \dots, path_k$) from the sources to the finished products in the process model by process tracking. Let E^r be the finish time of the r -th path.

$$E^r = a_r + b_r i, \quad r = [1..k]. \quad (10)$$

Definition2. Let set $A = \{E^1, E^2, \dots, E^j\}$, $a = \max(a_r | a_r + b_r i \in A)$, $b = \max(b_r | a_r + b_r i \in A \text{ and } a_r = a)$, $A_1 = \{a_r + b_r i | a_r + b_r i \in A, a_r = a, r = 1, \dots, k\}$, $A_2 = \{a_r + b_r i | a_r + b_r i \in A, a_r - b_r \geq \max(E^j), r \text{ and } j = 1, \dots, k, \text{ but } r \neq j\}$, $A_3 = \{a_r + b_r i | a_r + b_r i \in A, a_r + b_r < a + b \text{ and } a_r - b_r > a - b\}$, then all paths corresponding to A_1 are called *dynamic-time-critical-path*, and the static cycle of the project is a . If there are A_2 , then A_2 corresponds with the *absolutely time-critical-path*. $E = a + bi$ is the project dynamic cycle, and the associated path is called *main-time-critical-path*, and all paths corresponding to A_3 are called *hypo-critical-paths*.

Within the enterprise process simulation or process enactment, the *main-critical-path* and the *hypo-critical-path* usually could be transferred each other.

Based on the proposed definition, each A_k 's L_k can be calculated in reverse from the finished products using the rule of subtraction operation and the property of lower limit of compound number.

(1) Activity A_k in sequential structure

In the Figure 3, the sequential structure of process model is illustrated as that activity A_k has only one successor A_{k+1} .

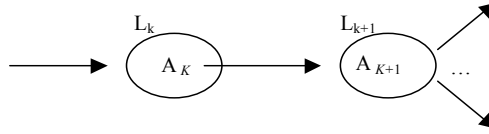


Figure 3. Sequential Structure of Activities A_k and A_{k+1}

The latest start time L_k of activity A_k is the difference of the latest start time L_{k+1} of its successor and the duration D_k of A_k . L_k can be calculated by the following formula:

$$L_k = (a_{l-(k+1)} - a_k) + (b_{l-(k+1)} + b_k) i. \quad (11)$$

(2) Activity A_k in parallel structure

Figure 4 illustrates that the activity A_k has some successors $A_{k+1}, A_{k+2}, \dots, A_{k+j}$. Now, Let $L_{k+1}, L_{k+2}, \dots, L_{k+j}$ are the latest start times of these succeeding activities. $L_k^1, L_k^2, \dots, L_k^j$ are the latest start time of A_k reversed calculated from $A_{k+1}, A_{k+2}, \dots, A_{k+j}$ in the method of sequential structure.

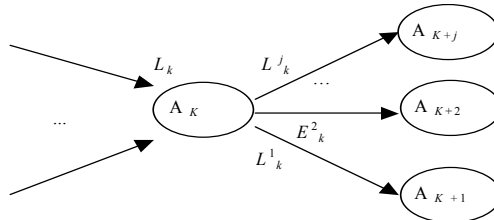


Figure 4. Activity A_k with Multiple Successors

The latest start time L_k of A_k can be obtained according to the following two steps:

- ① L_k^1, \dots, L_k^j can be obtained by reverse calculations from the successors using the rule of subtraction operation with L_{k+1}, \dots, L_{k+j} to the duration of A_k as formula (11);
- ② L_k can be calculated accord the property of lower limit:

$$L_k = \inf(L_k^1, L_k^2, \dots, L_k^j) \quad (12)$$

3. PROCESS-DRIVEN ENTERPRISE COOPERATIVE WORK

The enterprise applications with computer supported cooperative work have three level means. Firstly, it is the definition of the cooperative model. One is that dynamic PERT/CPM technique can be used to support flexible scheduling in process simulation interpreted by VPML [6] according to the cooperative behavior rules defined in the cooperative model. Another is that it can provide the assistant function with cooperative scheduling in enterprise process enactment, i.e., task schedule, which supports enterprise applications integration and process integration.

In the process definition environment, the customized enterprise model can be defined flexibly from process, infrastructure, behavior, cooperative and information. It is a static tool to describe and analyze the enterprise process.

In the process simulation environment, readyEventQueue is a queue used to store the events of activity readiness ordered by the cooperative schedule rules. Process simulation can flexibly simulate enterprise business processes, and activate the ready activities as the requirement of the enterprise in practical work.

In the process enactment environment, the information system will build a role-oriented Task-table for each personnel role defined in the enterprise process model. The Role-Oriented Task-table will store the work task ordered according to the cooperative rules. It can be used to assist each manager to execute the tasks as the requirements of cooperative work.

This section is organized as follows. Section 3.1 proposes a framework of process engine for role-oriented process-driven enterprise cooperative work; Section 3.2 introduces the flexible schedule strategies; Section 3.3 presents the implementation of the proposed strategies; Section 3.4 discusses process driven enterprise applications integration

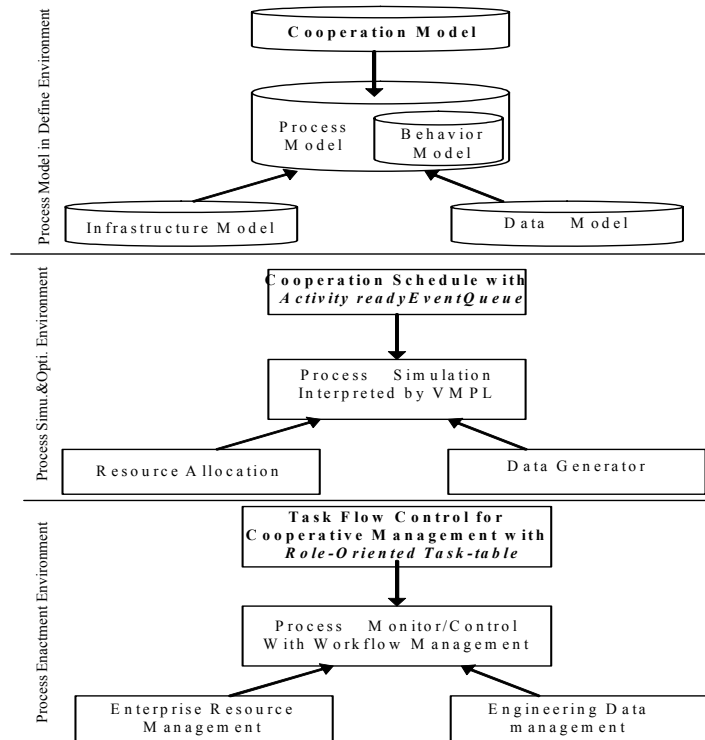


Figure 5. Dynamic Enterprise Modeling for Cooperative Work Framework

3.1 Role-Oriented Process-Driven Enterprise Cooperative Work Engine

Enterprise processes execution is based on event stream mechanism. We can define different process cooperative model, i.e., different combined schedule rules, for the individual processes. And there is an organization need to support a sub-process, i.e., a combined activity, as well as some roles or a resource group are require to support a leaf artificial activity. So, during process simulation, all activities in each sub-process can be activated driven by process schedule engine according to the activityReadyQueue, as well as during process enactment, each role may cooperatively work based on business process model according to the task schedule. Process schedule engine is illustrated as figure 6.

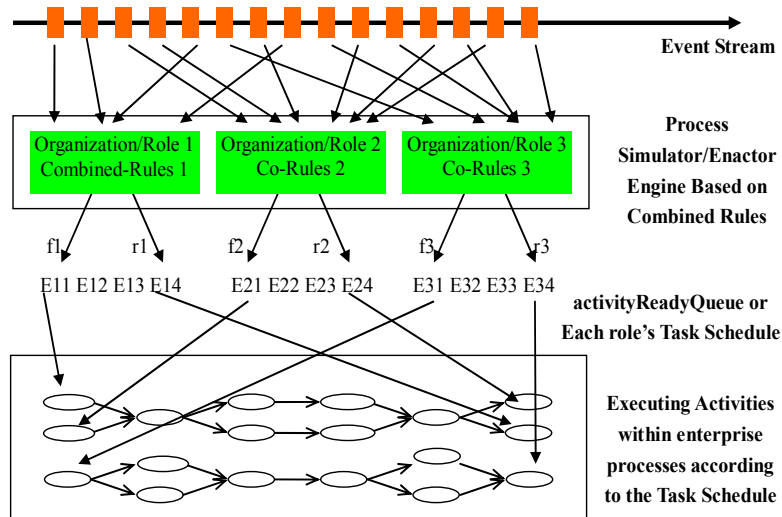


Figure 6. Role-oriented Process-driven Enterprise Cooperative Work Engine

3.2 Flexible Schedule Strategies

While simulating or enacting the enterprise process model, the key is how to control the execution order of the scheduled activities according to cooperative behaviour rules under resource constraints.

Activating activities is based on the queuing model and the discrete event spring mechanism. The events made by activity completing are put in the output-stream. For Output-OR sub-structure, events will put in different stream paths according to the probability of path defined by the modeling engineers.

In the proposed modeling environment, cooperative behavior editor provides a set of 8 kinds of schedule strategies. Process modeling engineers can define their own rules within cooperative behavior editor. The following schedule strategies exist [5]:

- Rule 1: HPFS (Highest Priority First Serve);
- Rule 2: MS (Minimum slack time first Serve);
- Rule 3: FCFS (First Come First Serve);
- Rule 4: SOT (Shortest Operation Time);
- Rule 5: LOT (Longest Operation Time);
- Rule 6: LRPT (Longest Remaining Processing Time);
- Rule 7: SRPT (Shortest Remaining Processing Time);
- Rule 8: SIRO (service in random order) etc.

In the above rules, MS, SOT, LOT, LRPT and SRPT need supporting of dynamic PERT/CPM technique. The combination of these rules can be used to schedule the manufacturing process or the project process.

By this method, all ready activities can be flexibly scheduled. All resources can be allocated to the business activities for which they are needed.

3.3 The Implementation of Schedule Strategies in Process Simulation

3.3.1 Random number generator.

Random number generator is the basic tool used to construct random variables according to the probability distribution. Here, we use compound number to present the non-deterministic process model parameters. The compound factor i of the duration D_k of activity A_k is created by the random number generator according to the distribution property (e.g., Constant, Uniform, Normal, Poisson) defined in the activity attribute. In order to guarantee the independence and reappearance in the multi-simulation of the process model, the proposed system provides two ways for random number generating: random mode and fixed mode. For the random mode, the random variables are created randomly every time, and they are different. For the fixed mode, it is necessary to generate a random number to input a seed variable. If the seed variable is the same, the random number generator will construct one stream using the same random number. Thus, the process simulations can be reappearance.

3.3.2 The algorithms of the schedule strategies

In order to implement all above schedule strategies, we have to modify the schedule's algorithm based on rule FCFS in the original PMSE (process model simulation environment) developed by the Software Engineering Institute at BeiHang University, China.

We have improved and extended some of functions of process simulation environment. A structure variable *ReadyEventQueue* was proposed to store the activity ready events sorted according to the proposed schedule strategies, and these activity ready events will not still stored in the *EventQueue*. But *EventQueue* still store the rest events according to FCFS and HPFS, such as activity active events, activity complete events. The schedule algorithm can process these activity ready events according to the schedule strategies defined by user.

Figure 6 illustrates the cooperative work of the ready activities controlled by 4 levels of combined rules in the proposed strategies. After location-analyzing according to the combined rules, the activity-ready-events will be located in the reasonable position in the pool of activity-ready-events. Then scheduler will allocate resources for the activities required and active these activities orderly.

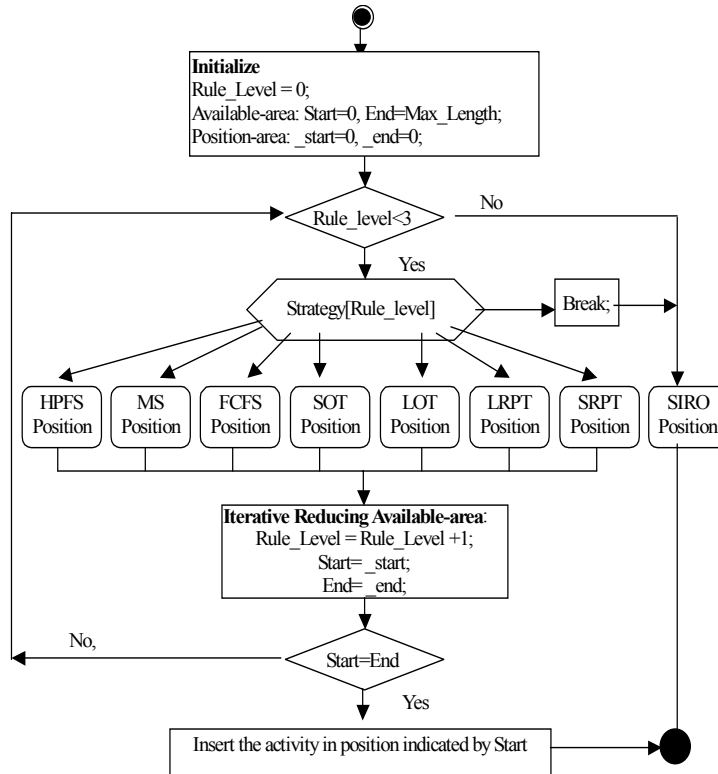


Figure 7. The Location Process of Ready Activity in Ready-Pool

Above these location-analyzing algorithm will reduce the size of position area of activity-ready-events from Available-Area [Start..End] to Position-Area [_start .. _end] besides SIRO-location-analyzing. If pointer _start and pointer _end point to the same position, the activity-ready-event will be inserted there. While the first iterative location-analyzing, the Available-Area is the whole-space of readyEventQueue. SIRO-Position is usually used as the last level rule, so that if the former 3 levels Location-analyzing cannot locate the activity-ready-event, i.e., there are still exist conflicts in the process of resource-allocation, the activity-ready-event will be located in random position in available area in readyEventQueue.

Here give the pseudo-code of MS rule's locate-analyzing algorithm as one example of locate-analyzing algorithm.

Table 1. The Pseudo-code of Algorithm for MS Rule

```

Boolean simReadyPool::reduceByMS(int Start, int End,
int* _start, int* _end, simActClone* act)
{
Boolean Result = FALSE;
//There are not activity which thelastStartTime is
//equal to *act

```

```

double lastTime←LastStartTime (*act);
for(int pos=Start; pos<End; pos++){
double lastTime_pos←LastStartTime(simActClone(pos));
if(lastTime_pos > lastTime) { break;}
if ((lastTime_pos == lastTime) && (!Result))
{ Result= TRUE; *_start = pos; }
//Record _start position
if ((lastTime_pos >= lastTime) && (Result))
{*_end= pos; break; }
//Record the end position _end
}
if (!Result) {*_start=pos; *_end=pos; }
return Result;
}

```

All activity-ready-events are stored orderly in the readyEventQueue. This order is a kind of dynamic priority, so that, readyEventQueue need dynamic updating. Re-calculating the compound factor i of the current event start time E_k and the L_j of all readied activities are required after advancing simulator clock.

3.4 Process Driven Enterprise Applications Integration

Enterprise Applications Integration (EAI) is the set of tools and technologies developed to integrate various dissimilar applications within the organization. Mostly, they consist of flowcharts or maps for invoking applications or sending and receiving messages in specific orders.

Many EAI products have some tools that enable a business to automate business processes by superimposing them across existing applications. BPMS [7] is a mechanism which rapidly develops and evolves custom business automation solutions by directly integrating with legacy and packaged systems as an implementation detail.

One of the goals of adaptive process engineering is to leverage these legacy systems while providing an easy migration path to model-based systems, by accessing legacy systems transparently.

Process models may be deeply nested within many levels. Each model manages its own affairs yet they all work together to solve higher-level business problems. The software architecture which supports the model-based approach consists of horizontal layers rather than vertical applications. The models occupy the middle layer and form the integrating structure of the systems. It is called the process integration server. An example is Biztalk developed by Microsoft Corporation. The model can be used to automatically estimate the time and cost to carry out a manual task [8]. By connecting the models through the Work-Flow-Management mechanism, many activities can be scheduled rapidly and intelligently. The conventional integration methods are usually based on the uniform strategies such as event-oriented FCFS, HPFS. But they do not provide the flexible definition function or selection of cooperation strategies to control the active order of ready activities.

A cooperation model was built within the enterprise model to attack the impact issues. The cooperation model selector provides the selection/definition function of

the proposed multilevel cooperative strategies for control task scheduling, allocating resources to assistant managers doing their work. These are following 8 strategies can be selected: HPFS, MSFS, FCFS, SIRO, SOT, LOT, LRPT, SRPT etc.. By combining these rules, all ready activities can be flexible scheduled, and all resources can be allocated available to the ready activities.

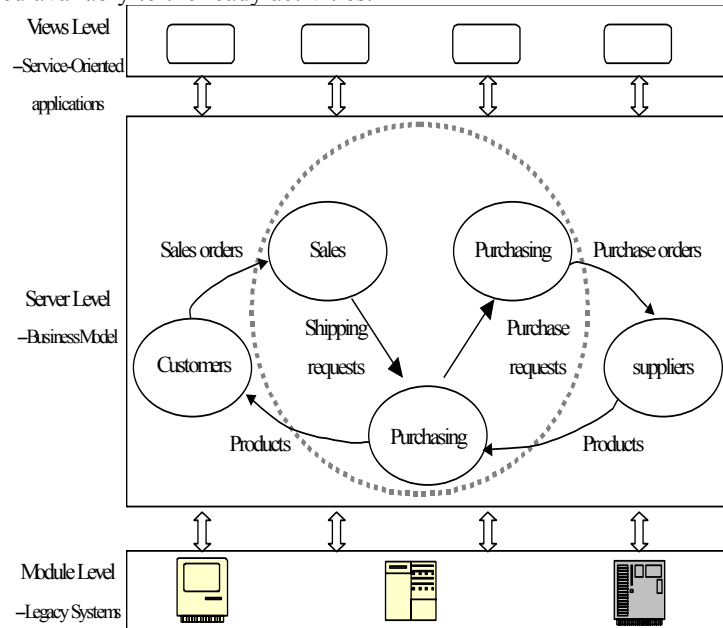


Figure 8. Architecture of Applications Integration for a Distributor Business Model

Within enterprise process enactment, relative services-oriented applications could be provided for the correlation role users [9]. It is very important to protect the investment of IT legacy systems. We usually use Web services to wrap legacy software systems for integration with modern IT systems. Each role user has a *Role-Oriented Task-table* which orderly stores all work tasks associated with the role according to the above cooperative rules. It is similar with aforementioned structure, *readyEventQueue*. The difference is that the *Role-Oriented Task-table* is only one real assistant schedule tool to assist the role users to work effectively, and the final determination and works are still determined by the managers [10].

4. CONCLUSIONS

The flexible scheduling of resources for activities is very important in process simulation or process enactment. In this paper, the dynamic PERT/CPM technique with compound number and its applications to enterprise management with computer supported cooperative work were discussed detail. A framework to process-driven enterprise cooperative work is proposed, and 8 kinds of scheduling rules and the

combined strategies algorithm implementation are discussed to implement an intangible cooperative work relationship in the manufacturing stream-like process or the project-oriented process model, as well as the process-driven enterprise application integration. It is very useful for the enterprise management in the domains of analyzing and optimizing the enterprise business processes.

ACKNOWLEDGEMENTS

The majority of the work presented in this paper was done by the first author during his Ph.D studies in BeiHang University, China. Appreciations should be given to Prof. Bosheng Zhou, the Ph.D supervisor and the team members in BeiHang University. This work was partially supported by the National Natural Science Foundation of Zhejiang province of China (Grant No. Y106039) and the Key Research Foundation for Zhejiang Education of China (Grant No. 20060491).

REFERENCES

1. B.S. Zhou, H. Xu, and L. Zhang, The Principle of Process Engineering and Introduction to Process Engineering Environments, *Journal of Software*. Number 8 (Supplement), pp.519-534, (1997).
2. L. Zhang and L. Wang, Process Simulation Technique and its Support Environment, *Journal of Software*. Number 8 (Supplement), pp.565-575, (1997).
3. C.S. McChahon, Using PERT as an Approximation of Fuzzy Projection-Network Analysis, *IEEE Transaction on Engineering Management*. Volume 40, Number 2, pp.146-153, (1993).
4. X. He, *The Theory and Technology of Fuzzy Number Knowledge* (National Defence Industry Press: Beijing, 1994).
5. W. Tan, B. Zhou, and L. Zhang, Research on the Flexible Simulation Technology for Enterprise Process Model, *Journal of software*. Volume 12, Number 7, pp.1080-1087, (2001).
6. B. Zhou, Visual Process Modeling Language VPML, *Journal of Software*. Number 8(Supplement), pp.535-545, (1997).
7. E.D. Jenz, *BPMS and Web Services: An Unbeatable Team*. www.webservices.org
8. W. Tan, *A Study and Development for Dynamic Optimizing Enterprise Process Technique and its Supporting Environment*. Ph.D Thesis, BeiHang University (2001).
9. D.A. Taylor, *Business Engineering with Object Technology* (John Wiley & Sons: New York, 1995).
10. Y. Rezgui, Role-based service-oriented implementation of a virtual enterprise: A case study in the construction sector, *Computers in Industry*. Volume 58, Number 2, pp.74-86, (2007).