The Workflow Simulation System With the Flexible Output Customization Function

Sen'ichi Onoda, Yoshitomo Ikkai and Norihisa Komoda

Department of Information Systems Engineering
Faculty of Engineering
Osaka University
2-1, Yamadaoka, Suita
Osaka 565-0871
JAPAN
E-mails: {onoda, ikkai, komoda}@ise.eng.osaka-u.ac.jp

Abstract: This paper proposes the Flexible Output Workflow Simulation system(FOWS), which enables to model and simulate a business process presented by the workflow model and to freely output optional criteria data evaluating the process with minimal setup effort. The main features of FOWS are "the function of adding optional properties to nodes or work items" and "the function of linking FOWS with an external program out of FOWS". FOWS has been developed with Microsoft Visual C++. In order to demonstrate how easy it is to obtain certain criteria values with FOWS, a sample simulation with FOWS is shown. The existing business process was applied to FOWS, and the result of it shows the serviceability of FOWS.

Keywords: workflow management system, discrete simulation, business process

Sen'ichi Onoda received his BE and ME degrees from Osaka University in 1997 and 1998 respectively. He is currently a Ph.D student at the Department of Information Systems Engineering, Faculty of Engineering, Osaka University. His research area of interest includes workflow systems. He is a member of the Institute of Electrical Engineers in Japan.

Yoshitomo Ikkai received his BE, ME, and Ph.D degrees from Osaka University in 1993, 1995, and 1999 respectively. He is currently an Assistant Professor at the Department of Information Systems Engineering, Faculty of Engineering, Osaka University. He studied at Eindhoven University of Technology in 1999. His research areas of interest include scheduling, supply chain management and knowledge engineering. He is a member of IEEE, the Institute of Electrical Engineers in Japan and of the Society of Instrument and Control Engineering. He received IEEJ Distinguished Paper Award of 1998 from the Institute of Electrical Engineers in Japan.

Norihisa Komoda received his BE, ME, and Ph.D degrees from Osaka University in 1972, 1974, and 1982, respectively. He is currently a Professor at the Department of Information Systems Engineering, Faculty of Engineering, Osaka University. From 1974 to 1991, he was with the Systems Development Laboratory, Hitachi Ltd. He was at the University of California, Los Angeles, as a visiting researcher from 1981 to 1982. His research areas of interest include planning and evaluation of business information systems. He is a member of the IEEE, the ACM, the Institute of Electrical Engineers in Japan, and the Society of Instrument and Control Engineering. He served as Program co-chair of ETFA '96, track chair of ETFA '94, ETFA '97, ETFA '99, ISIE '98, and IECON 2000. He received the awards for outstanding paper of 1986 and the awards for outstanding technology of 1987 both from the Society of Instrument and Control Engineering and IEEJ Distinguished Paper Award of 1998 from the Institute of Electrical Engineers in Japan.

1. Introduction

At present, companies need drastic business reorganization to make the business process over plural departments or companies more efficient to survive keen competition[1]. In BPR(Business Process Reengineering), which is one of the methodologies for drastically re-composing and improving the business process, the workflow management system attracts a great deal of attention [2][3][4]. The workflow management systems perform definition, execution, and management of the business processes.

In introducing a workflow system, a quantitative analysis of the business process is needed for the evaluation of the current business process. The business process is evaluated according to various criteria, for example "working time", "jams of work items in each work", and so on (the work item is the representation of the work to be processed). Until work items are processed in a business process, these criteria values cannot be calculated. Therefore, the criteria values are mainly obtained by simulation.

In most of the workflow simulation systems, the business process is defined visually with GUI(Graphical User Interface) [5]. Though there are some differences among systems, it is of paramount importance to describe the flow of activities, which means minimum working units of process. However, the traditional workflow simulation system prepares only fixed criteria, so that it is almost impossible to calculate additional criteria values, for example "the average working quantity of each worker", which users desire for the business process investigation. On the other hand, although the

simulation system generators such as GPSS can freely develop simulation systems for evaluating all kinds of criteria values, users must learn the new language or tool commands. Moreover, users must also program the module of the workflow model definition. Therefore, a great deal of users' labor is necessary.

To obtain criteria values freely and easily, this paper proposes the Flexible Output Workflow Simulation system (FOWS), which enables modeling and simulation of a business process by the workflow model and to freely output optional criteria data for evaluating the process with minimal setup effort. The variable definition function by the changeable property list and the linkage function of the external program for optional criteria values calculation are realized.

Section 2 explains a workflow model used in FOWS. The characteristic functions of FOWS are described in Section 3. In Section 4, the system composition of FOWS is explained in order of usage. The example of simulation with FOWS, which shows the serviceability of FOWS, is demonstrated in Section 5.

2. The Workflow Simulation

In FOWS, the business processes, that means the flows of works, are described in the workflow model shown in Figure 1. The order of work execution in a business process is graphically expressed by the workflow model. Work items are transferred following the defined model. A "work item" represents the electronic document to be processed in the business process. The workflow model has seven sorts of nodes which are defined by the Workflow Management Coalition (WfMC)[6].

- Activity node

This node represents a unit of work in a business process. The activity node is described by a rectangle.

- Source node

This node is the start node of the business process.

- Sink node

When a work item comes to this node, the business process which the work item represents is over.

- AND-split node

This is a node where a work item is split into two or more items whose IDs are the same.

- OR-split node

This is a node where a work item is passed to one of the two or more output points under the predefined rule.

- AND-join node

This is a node of the workflow where two or more parallel executing activities converge into a single common thread of control.

- OR-join node

This is a node where two or more alternative activity workflow passes are merged into a single common activity as the next step.

Each node has input and output points. The input point is the point which is connected with the previous node by an arc. The output point is the point which is connected with the following node. The six types of nodes, except "activity node" are generally referred to as "the control node". The nodes of the workflow model have some properties which are needed for simulation.

In simulation, the work item is processed by the

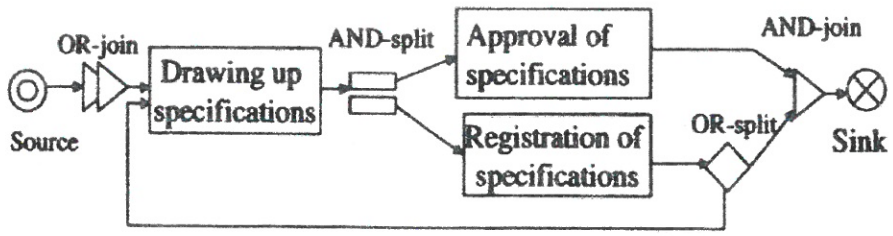

Figure 1. An Example of A Workflow Model

program defined in each node and is transferred in the context of the structured model. At each node, the property values, which belong to the node itself or to the work item, are updated. As a simulation result, the log of work items processing is stored, and the criterion values can be calculated.

3. The Main Idea of FOWS

Figure 2 shows the difference between traditional workflow simulation systems and FOWS. In the traditional workflow simulation systems, when the activity node receives a work item, the execution of the program for the criteria values calculation prepared in the system starts. The executed program updates the same property values (such as property A in work item or property B in activity node in this Figure) in the system. The updated criteria are also fixed.

external program receives arguments values, which consist of properties of a work item or a node (such as property X in a work item or property Y in an activity node in Figure 2). These properties can be added optionally by users. The external program sends the return values to FOWS, and FOWS updates the property values. When the simulation is over, the external program outputs the criteria values to a text file. In this way, users can get the optional criteria values.

For realizing the flexible simulation system, the following two functions are necessary. One is to add properties to a node or a work item. These properties are used for calculating the optional criteria values. The other function is to link FOWS with an external program prepared by users. These two functions are realized as follows:

- The function for adding optional properties to a node or a work item

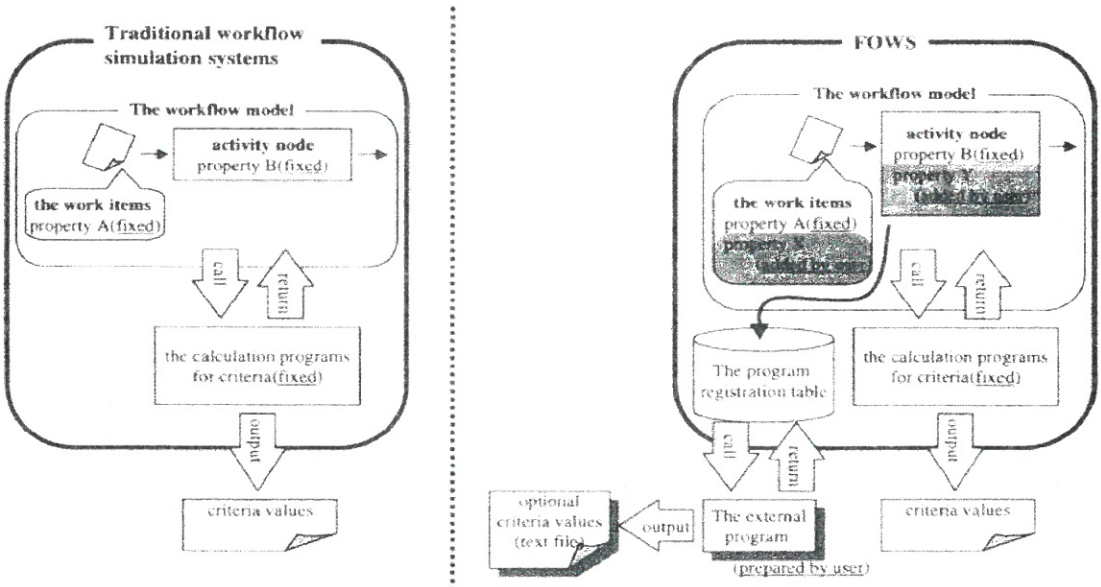

Figure 2. The Difference Between the Traditional Workflow Simulation System and FOWS

On the other hand, in FOWS, the program prepared by the user for the criteria values calculation can be added by the user. It is called the "external program". The external program outputs the pre-prepared optional criteria values and optional property values. Arguments are variables transferred from FOWS to the external program. Return values are variables transferred from the external program to FOWS. When the activity node receives a work item, FOWS calls the external program referring the program registration table, which includes the information about the external programs. The

Nodes and work items can have changeable list type data for optional criteria values. The list type data consist of the names, the type and the value of the properties. The user can add the optional properties to a work item or node by inserting the data into this changeable list.

- The function for linking FOWS with an external program

FOWS which is equipped with the above functions has been developed for Microsoft Windows 95 with Microsoft Visual C++. In order to actualize this function, FOWS uses the function "ShellExecute", supported by

Microsoft Visual C++. The "ShellExecute" function is used for the "open the program" from the application built with Microsoft Visual C++. For using the "ShellExecute" function, FOWS is equipped with *the program registration table*. This table consists of all what is needed for the "ShellExecute" function, for example the external program name, the node type, which calls the external program, arguments, return values, and so on. FOWS is also equipped with the setup window for this information input.

4. The System Functional Configuration

Figure 3 shows the whole system functional configuration of FOWS and interim output files. Each functional part is explained in usage order.

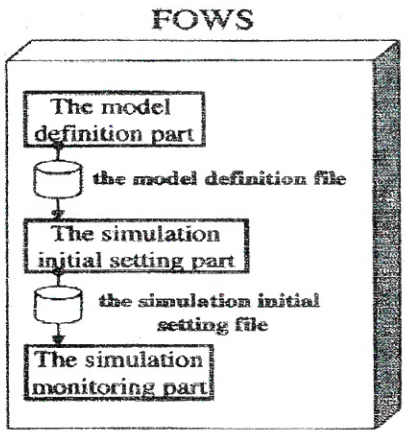

Figure 3. The System Functional Configuration

4.1 Model Definition

A business process, external program registration and optional properties are defined. This part outputs "the model definition file".

1. The business process definition

The definition window shown in Figure 4 is used for editing the process model. In the "Element Selection" window shown in the lower right corner of Figure 4, the inserted node type is selected. The selected node is put in the "Process Definition" window (a larger window). When the user clicks the input and end point, those two points are linked by an arc.

When the arranged node is double-clicked in the business process definition window shown in Figure 4, the node setup window shown in Figure 5 is evoked. In this window, node name, conditions for controlling the flow of work items, and so on, are defined.

2. The external program registration

The window shown in Figure 6, which is evoked by the menu in the business process definition window, is used in order to get ready for linking the external programs and for transferring arguments to the external program. The *Element* field is used for selecting one node type from 7 node types, which should be linked with the external program. As all the program names in a provided folder are displayed in the down menu of *Program Name*, the user only selects one for use. On the right side of this window, the *Additional Property* field is prepared for setup of arguments and return values. If the user wants to set the arguments, first the user checks the radio button, *Argument*. Then, the user selects the one node type which the argument belongs to at *Entry in*. When the user selects the node type or the work item, names of properties which belong to the selected node type or work item appear in the *Registered Properties* field. When

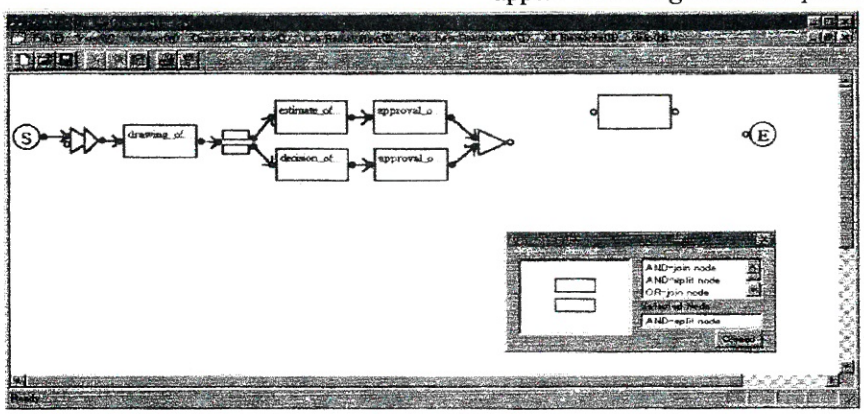

Figure 4. The Business Process Definition Window

the user pushes *Addition*, the selected property is added to *List of Argument*, the bottom left-hand of the window. Similarly, return values are added. If the user prepares the text file described information about the selected program's arguments and return values, FOWS can display it in *information about program*.

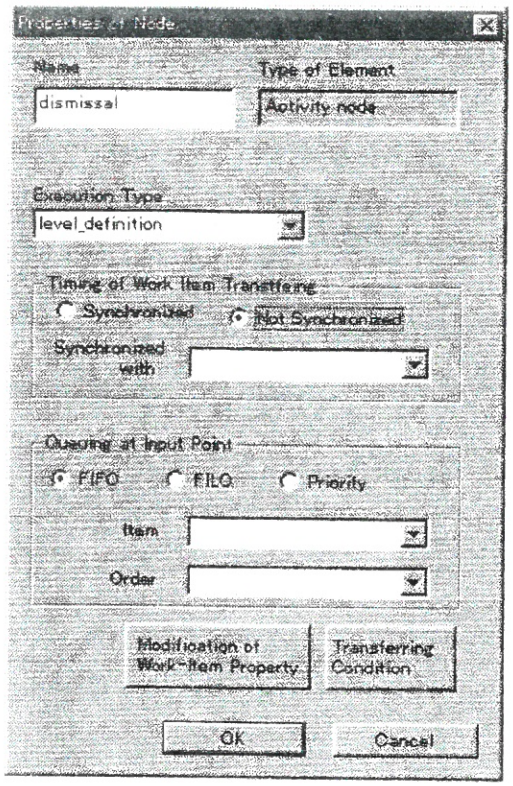

Figure 5. The Node Setup Window

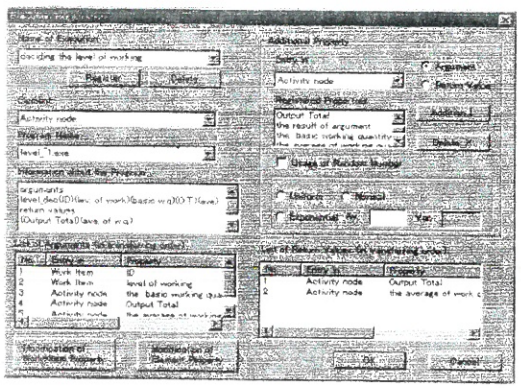

Figure 6. The External Program Registration Window

3. Optional property addition

The optional property addition window shown in Figure 7 is evoked by the menu in the business process definition window or the external program registration window. Nodes and work items can have list-type data for the calculation of optional criteria. The list data consist of *the name of the property* and *the type of the property*. Users can access this list-type data through the window shown in Figure 7.*The name of the property* is entered through the keyboard. *The type of the property* is selected from three types: integer, decimal and character. Finally, by pushing the add button in the window, an optional property is added to the selected element. All properties registered in the selected element are shown in *Registered property*. Once a property is added to the node, all of the same kind of nodes in the model have it.

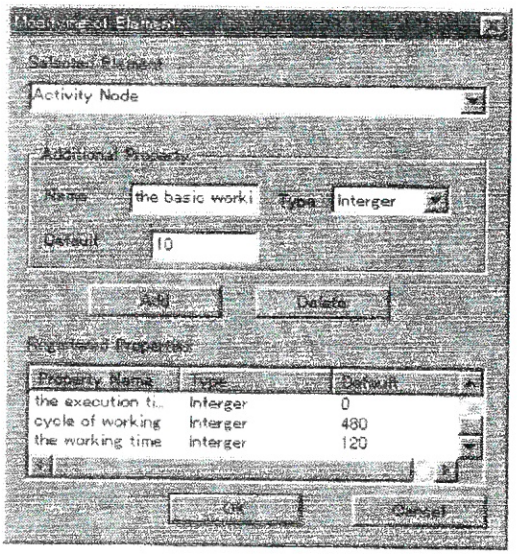

Figure 7. The Optional Properties Addition Window

4.2 Simulation Initial Setting

In this part, the initial setting for simulation is defined, for example the number of simulated work items, the probability distribution of the work item generation intervals, the simulation completion condition, the initial values of a work item's properties, and so on. This part has the main initialization window as in Figure 4. The initial values of the node properties are also defined with the values initial setting window shown in Figure 8. When the node is double-clicked in the main initialization window, the values initial window shown is evoked. At first, the user selects the target property in *Properties*. Then, the user inputs the value in *Property Value*. Finally, by pushing the *Decision* button, the initial property value is set. Moreover, the

branch incidence is also set in this window. This part outputs "the simulation initial setting file".

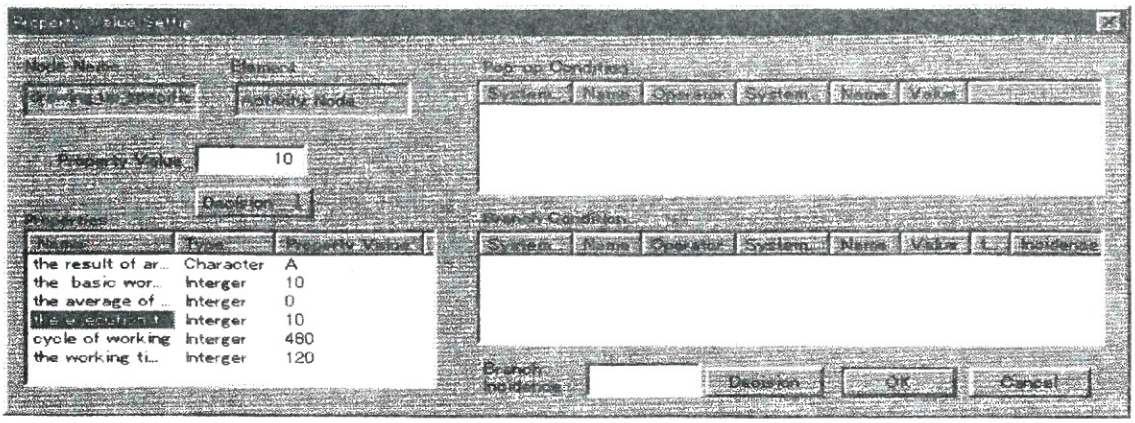

Figure 8. The Value Initial Window

4.3 Simulation Monitoring

In this part, simulation is carried out under the designated workflow model and condition. The user can monitor the state of simulation, as shown in Figure 9. The state of processing and the number of stagnant work items in each node are displayed. The number of generated work items in the source node, and the number of processed work items in the sink node are displayed. The optional criteria values are output as a text file by the external programs.

5. The Example of Simulation

5.1 The Objective Process

The workflow model of "the estimation for the business information system" consists of 7 activity nodes and 7 control nodes as shown in Figure 10. These 7 activity nodes are described in Table 1.

At the OR-split node in this process, the branch condition is defined as follows.

- In case a work item gets approval in both "approval of estimates" and "approval of specifications", the work item is passed to "drawing up specifications for the client", and the process is over. (This case is named Case A.)
- In case a work item does not get approval in "approval of estimates", the work item is dismissed. (This case is named Case B.)
- In case a work item gets approval in "approval of estimates", but not in "approval of specifications", the work item is sent back to "drawing up specifications" for reconsideration. (This case is named Case C.)

In this example, "the average working quantity of each worker", which is defined as the average quantity of processing work item, is required to be evaluated by the simulation. "The working quantity" for checking the load of

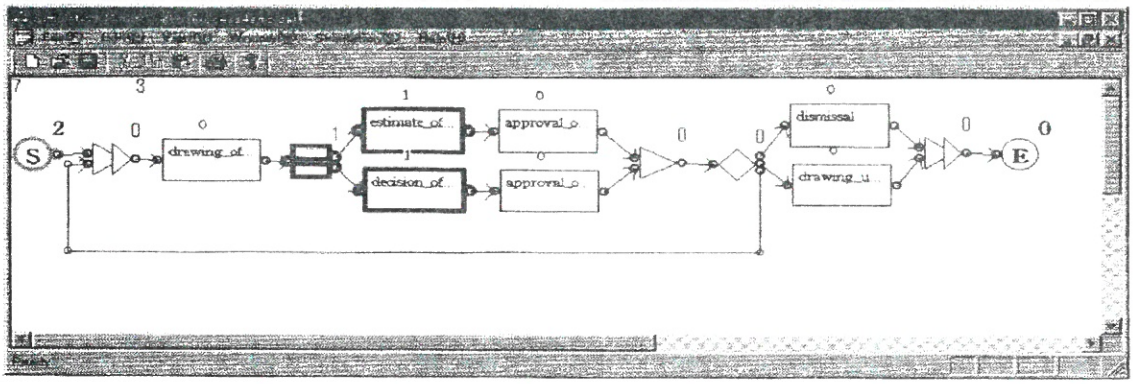

Figure 9. The Simulation Monitoring Window

workers quantifies workers' labor for a work item. It is valued according to the weight belonging to work items. The working quantity is decided according to the level of working, which belongs to the work item, as follows. Each activity node has the basic working quantity, which is the basis of the working quantity decision.

Work Quantity =	{	*the basic work quantity x 0.8*	*:the level of working is 1.*
		the basic work quantity	*:the level of working is 2.*
		the basic work quantity x 1.2	*:the level of working is 3.*

Table 1. Contents of Activity Nodes

Name	Contents of working
Drawing up specifications	Specifications of information systems are drawn up.
Estimation of the cost	The cost for information system is estimated.
Approval of estimates	It is judged whether estimation of the cost is approved or not.
Decision of a responsible person	The responsible person for building the system is decided.
Approval of specifications	It is judged whether the specifications are approved or not.
Dismissal	The work item which is not approved is dismissed. The dismissal is reported to the client.
Drawing up specifications for the client	Specifications for the client are drawn up.

5.2 Implementation of the Workflow Model

Added properties are embedded into the model definition part.

- The work item

The properties shown in Table 2 are added to the work item, using the window shown in Figure 7.

The level of working uses *the program for deciding the working quantity*. When a work item is generated, one of them is decided according to the incidence preestablished in the simulation initial setting part by the user.

- Nodes

The properties shown in Table 3 are added to nodes, using the window shown in Figure 7. This window is called only once, because there is one node of which properties are added to.

The basic working quantity is the basis of the working quantity decision.

- The external program

Further the external program is coded and registered using the window shown in Figure 6. Arguments and return values for the external program are input.

— the program for deciding the working quantity;

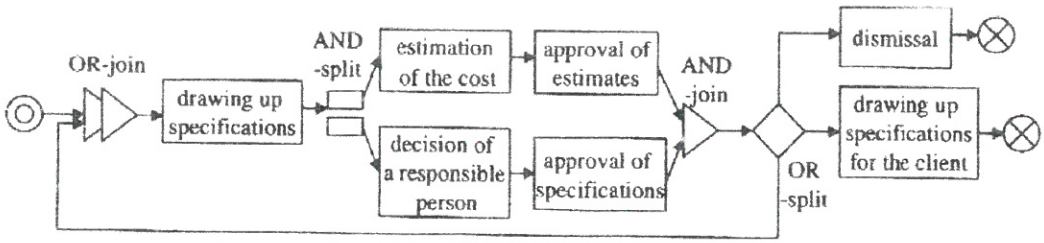

Figure 10. The Process of "the Estimation for the Business Information System"

Table 2. Additional Properties to the Work Item

property name	type
the level of working	integer

Table 3. Added Properties to the Nodes

objective node	property name	type
activity node	the basic working quantity	integer
	the average working quantity	decimal

This program is linked with all the activity nodes in this process. This program decides the working quantity of each work item, and calculates the average of these quantities in each node. It requires 6 arguments and 2 return values. It records the average of working quantities on the text file.

5.3 Simulation Result

20 work items were simulated. The number of simulation work items has been decided according to the log data. As a result, the average quantity of working in each activity node could be obtained as shown in Table 4.

Table 4. The Result of Simulation

activity node name	the basic working quantity	the average working quantity
drawing up specifications	10	9.8
estimation of the cost	15	14.8
approval of estimates	30	29.5
decision of a responsible person	5	4.9
approval of specifications	20	19.7
dismissal	3	2.9
drawing up specifications for the client	25	24.7

5.4 Considerations

In the model definition part, the labor for the model definition was as much as with the traditional workflow simulation system. Considering optional property addition, the optional properties addition window shown in Figure 7 was called only twice (once for the work item, and the second time for nodes). As to the external program setup, the external program registration window shown in Figure 6 was called only once. The external program is programmed in C language, and its steps are 90. It is quite easy for this user to program it.

6. Conclusion

In this paper, we proposed the Flexible Output Workflow Simulation system (FOWS) which enables the modeling of a business process by the workflow model and the output of optional data with minimal setup. In this system, we realized two functions, namely "the function of adding optional properties to a node or work item" and "the function of linking an external program out of FOWS". Because FOWS can link a program developed in any programming language, the external program could be developed with a familiar program language. Therefore, FOWS can save the users' labor in the optional criteria output. FOWS's serviceability in the output of optional criteria values is shown through the result of a sample simulation.

REFERENCES

1. MORRIS, D. and BRANDON, J., **Re-engineering Your Business**, MCGRAW-HILL INC., New York, NY, 1994.
2. FISCHER, L., **New Tools for New Times: The Workflow Paradigm,** Second Edition, Future Strategies Inc., Lighthouse Point, FL, 1995.
3. KOULOPOULOS, T. M., **The Workflow Imperative: Building Real World Business Solutions**, VAN NOSTRAND REINHOLD, New York, NY, 1995.
4. JACKSON, M. and TWADDLE, G., **Business Process Implementation: Building Workflow Systems**, ADDISON -WESLEY LONGMAN LIMITED, Harlow, 1997.
5. SCHOLZ-REITER, B. and STICKEL, E., **Business Process Modeling**, SPRINGER-VERLAG, Berlin, 1996.
6. LAWRENCE, P., **Workflow Handbook 1997**, JOHN WILEY & SONS, Chichester, UK, 1997.