Research on Application of Hierarchy Petri-Net in Dynamic Workflow Modeling

Mortaza Mokhtari Nazarlou

Computer group, Islamic Azad University, Maku Branch, Maku, Iran

Abstract: In modern organizations, business processes are variable and flexible. So the workflow should have the ability to cope with the uncertain factors. The dynamic hierarchy refinement Petri-net-based is one of the methods that can handle the uncertain factors. At build-time with this method, the determining factors compose the basic transition. The uncertain factors are encapsulated to the non-basic transition. Thereby, determining factors and uncertain factors are both taken into consideration at build-time. Besides the thought of software component is introduced into the process modeling; the flexible process is encapsulated to component the reusability and maintainability of the process can be improved. Finally through an example, the author recounts the modeling process of dynamic workflow with hierarchy refinement Petri-net- based approach, verifies the soundness as well.
[Mortaza Mokhtari Nazarlou. **Research on application of hierarchy petri-net in dynamic workflow modeling.** *Life Sci J* 2013;10(1):821-825] (ISSN: 1097-8135). http://www.lifesciencesite.com. 129

Keywords: flexible workflow model; Petri-net; dynamic hierarchy refinement; component

1. Introduction

The workflow is an advanced mean to organize and optimize the process. However, the environment of enterprises is changing constantly; the goals of enterprises are adjusted continuously. Uncertainty and variability have been the internal features of modern enterprise processes. Modern enterprise processes show dynamics much more, such as, the Concrete definitions of part business processes are finished by terminal users at run-time; workflow is adjusted along with the change of internal and external conditions, and the adjustm.ni is fed back to the running instances of workflow, etc. Therefore, in workflow research area, strengthening flexibility and the ability to copy with process change has been a hot issue. Researchers at home and abroad have done a lot of work about how to improve the flexibility of workflow.

Reference [1] adopts the exception handling means, with ECA (event-condition- action), adopts different handling tactics against the changes of conditions. Reference [2] adopts the way of integrating external tools, strengthens the flexibility, openness and the ability of external interaction of the workflow system through integrating external groupware.

Reference [3] [4] provide user primitive or command operation to intermeddle the executing of-workflow conveniently. When workflow system is operated with these primitive, the modified operation can keep consistent with the original workflow through certain algorithm.

Winchester [5] introduces Choice-Merge and XOR- Split etc. in the workflow model. With the method, known process can be defined as many as possible, but the unknown processes cannot be defined. In order to overcome limitations of this method, Sadiq [6] put forward new methods, such as Black Box and pocket etc. That is to say new factors are introduced in the process model. These methods make up the defect that the definition of process is too complicated, the dynamic processes are supported to' some extent; but the process model become more complicated because new factors are introduced in the definition.

For solving above problems, the paper puts forward a hierarchically refining method for the workflow at the build- time; the method of refining in layers from top to bottom is adopted.

Firstly, build the model at the highest layer; encapsulate uncertain process to the non-basic transition. Structural integrities of workflow at the higher layer are maintained. When process instances run to the non-basic transition, uncertain details have been known.

Then refine the non-basic transition at the next layer, thus the flexibility of workflow is maintained. Moreover, in the paper, the thought of software component is introduced in the model, encapsulate the business unit and flexible process to components, and register them to the library of Petri-net model. At the time that design similar workflow, the models in the model library can U* quoted directly, of can be modified to fulfill the modeling for hierarchical workflows.

2. Workflow Nets

Definition1:

[7] A sound workflow nets (WF nets) is a petri-net with places p_{in} and P_{out} that satisfies the following conditions: (1) for each token put in P_{in}, one and only one token eventually appears in P_{out}.

(2)When the token appears in P_{out}, all other places are empty. (3)For each transition (i.e., task), it

is possible to move from the initial state to a state in which this transition is enabled.

WF nets discriminate between enabled and executive accurately. When the input token of the activity satisfy constraint rules, the activity is enabled. If an enabled activity turns to executive really, corresponding trigger mechanism must be satisfied. Trigger mechanism is divided into four categories, which are Automation, Artificiality, Information, time as illustrated TABLE 1.

Triggering way	Automation	Artificiality	Information	Time
Symbol presentation	None			

Table 1: Symbol presentation of triggering way

3. Modeling hierarchically for the WF nets

Introduce the concept of hierarchical workflow into the workflow model. In the present research, there are two ways to build hierarchy Petri-net. One is that the token is refined to the sub-Petri-net; the sub-Petri-net and the main Petri-net compose the hierarchical Petri-net. The other is that the place or the transition is refined to the sub-Petri-net; the sub-Petri-net and the main Petri-net compose the hierarchical Petri-net.

The paper adopts the latter method. There are two kinds of tasks in the workflow. One is the atomic task that is indecomposable and can be executed directly. The other is the composite task that can be refined into a sub- process. Thus the transitions in the Petri-net can be classified into two kinds, one is basic transition (such as T 1, T2) and the other is non-basic transition (such as T3), as illustrated in the Fig.1.

The atomic task is represented by the basic transition, and the composite task is represented by the non-basic transition. The non-basic transition can be refined to a sub- Petri-net. When the non-basic transition is triggered by user, the sub-Petri-net is triggered.

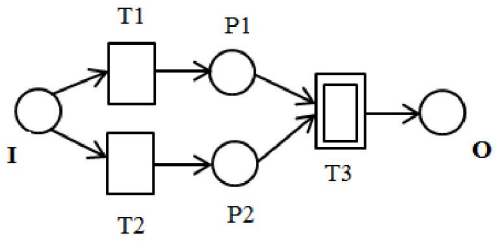

Figure 1: A Petri-net including non-basic transition

The workflow is divided into two stages, building stage and running stage. Model for the workflow at the building stage, interpret model at the running stage.

At the building stage, encapsulate the flexible process to the non-basic transition, Trigger mechanism of the non-basic transition is set to artificial triggering. Thus the main workflow is modeled at the higher layer. At the running stage, the non-basic transition is hierarchically refined to the sub-Petri-net through the TT-Petri-net refining operation [8].

A. Hierarchy Refinement Rules

Refining Petri-net means that the non-basic transition in the main Petri-net is refined to TT-Petri-net [8]. When all the non-basic transitions are refined, the main Petri-net with non-basic transition is translated into combining Petri-net without non-basic transition. And the combining Petri-net is live and bounded. The modeling for dynamic workflow is finished. There are strict limits for refining operation of places or transitions. Firstly, there are only one input places and only one output place. Secondly, when the sub-Petri-net is finished, the numbers of tokens in the input node and in the output node must be equal. In place refining Petri-net, the nodes of input and output must be places, and the place refining Petri-net is named PP-Petri-net [8]. In transition refining Petri net, the nodes of input and output must be transitions, and the transition refining Petri-net is named TT Petri-net. The paper refines the non-basic transition to the TT-Petri-net. The feature of TT-Petri-net is that none of the nodes in the TT-Petri-net are connected to the rest of the nodes outside the TT-Petri-net except the T_u and T_v, as illustrated in Fig.2.

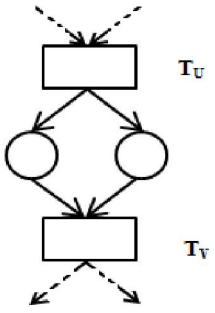

Figure 2: TT-Petri Net

The five basic refinement transformations [17] are shown in Fig.3.

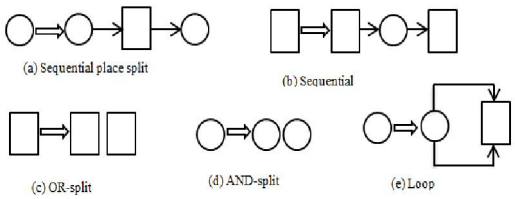

Figure 3: Basic Refinement Rules

There is corresponding relation between Petri-net structure produced by five basic refining transformations and routing structure of workflow, so this method can complete all types of workflow modeling task.

For all the rules displayed in Fig.3, except the first one, we require that they can be applied only if there is at least one input arc and at least one output arc associated with the node to make the transformation valid. Moreover, we presume that all the input arcs of the node are copied to all the resulting entry nodes, and that analogous rule applies to the output arcs. In addition, OR-split must follow OR-join closely, and AND-split must follow And-join closely.

B. Modeling for the sub-process component-based

Introduce the thought of software component into the workflow model, and then model for dynamic sub-process. The goal is improving dynamics of systems and reusability modules. In the software engineering, components are reusable software modules facing the software architecture. Based on this thought, the paper encapsulates dynamic sub-process to component, and adds them to the component libraries. When the similar non-basic transition is triggered, non-basic transition may call sub- process from component libraries. When the similar sub-process is designed, the sub- process may be obtained by existing sub-process in component libraries.

The refining procedure of non-basic transition is as follows.

- Step 1: Analyze the flexible business process, including dividing the node of business process and building structure. Then analyze functions realizing of nodes, operation role, resources and calling application programs. Finally, give the description of flexible business process.
- Step2: In the component libraries, inquire flexible business process described by stepr.tf inquiring is successful, jump to step4, or else executes step3 in sequence.
- Step3: Model for the flexible business process described in the step l, adopting TT-petri-net refining operations. Modeling of the flexible business process is a TT petri-nit. Then register and save the TT-petri-net.
- Step4: verify the soundness of TT-petri-net to ensure the soundness of combining petri-net. If combining petri-net is sound, jump to step6, or execute step5 in sequence.
- step5: Modify the TT-petri-net, and then jump to step4.
- Step6: Replace the non-basic transition with TT-Petri-net.

When the step6 finished, the number of non-basic transitions is minus one in the main petri-net. When the number of non-basic transition is zero, the main Petri net with non-basic transition is refined to combining Petri-net without non-basic transition.

C. Analyze the correctness of the workflow nets

The correctness of workflow model has two layers of meanings. One is the correctness of the structure of workflow model, that is to say the workflow model has neither structural conflict nor deadlock. It can properly end without exception happening. The other is the correctness of the semantics of workflow model, that is to say the practical business process achieves the goal determined in the building stage. In the paper, the correctness of the structure of workflow model is analyzed. H.M.w verbeek and etc. [9] propose a proof method; indicate that the consequence correctness of workflow is equal to flexibility and boundless of its extended network.

Definition 2:

[8] Let $\sum = (N, M_0)$ is a Petri-net.

(1) A transition t of T is live iff $\forall M \in R(M_0), \exists M' \in R(M)$;

(2) $\sum$ is live iff $\forall t \in T$, t is live.

Definition 3:

[8] Let $\sum = (N, M_0)$ is a Petri-net.

(1) A place p of P is bounded iff there exists $k>0$, $M(P) \leq k$, $\forall M \in R(M_0)$;

(2) $\sum$ is bounded iff $\forall p \in P$, p is bounded.

Definition 4:

[8] In the Petri-net (N', M'_0), obtain the sub-net from t_u to t_v, passing by $N_{TT,}$ and increase the transition t_{TT} and arcs set $\{(p,t_{TT}) \mid p \in t_v\} \cup \{(t_{TT}, p) \mid p \in t_u\}$, identification does not changes, get closed TT-Petri-net $(\overline{N}_{TT}, M_{TT0})$

Theorem 1:

[8] Let (N', M'_0) and (N, M_0) are Petri-net, and (N', M'_0) is refined from (N, M_0) through TT-Petri-net refining operation (Ref ($\bar{t}$, N_{TT})). (N, M_0) is bounded iff both (N, M_0) and ($\bar{N}_{TT}$, M_{TT0}) are bounded.

Theorem 2:

[8] Let (N', M'_0) and (N, M_0) are Petri-net, and (N', M'_0) is refined from (N, M_0) through TT-Petri-net refining operation (Ref ($\bar{t}$, N_{TT})). (N', M'_0) is live iff both (N, M_0) and ($\bar{N}_{TT}$, M_{TT0}) are live.

5. Example analysis

In this paper adopt the rule of dynamic hierarchy refinement to model for the treatment process. The treatment process has been characterized by dynamic.

At the build-time, not all of activities but most of them can be defined, such as registration, primary diagnosis, and payment. Only when the doctor gives the result of primary diagnosis, it review process can be defined.

Fig.4 shows the dynamic processes of patient's treatment, noted (N, Mo).

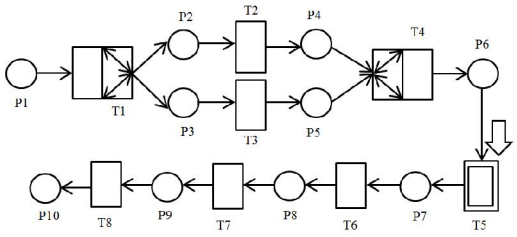

Figure 4: patient's treatment process

In Fig-4, the concrete meanings of places and transitions are as follows.Pl is the start place and p10 is the end place.

There are two kinds of transitions in Fig.4. One kind is the basic transition (such as T l, T2), the other is the non-basic transition (such as T5).

Non-basic transition can be refined to sub-process.

T1: Registering;
T2: creating an electronic medical record;
T3: Inquiring electronic medical records;
T4: Primary diagnosis;
T5: Multinomial medical examination;
T6: Final diagnosing;
T7: payment
T8: Getting the medicine.

According to the patients' diseases, the patients can undergo different examinations. For example, there is a patient suffering from osteosarcoma; his hospital process is as follows.

Firstly, analyze the puncture medical records. If the cause of disease cannot be determined, acquire the slicer, and then test them. In addition, the patients are supposed to receive chest CT detection and bone scanning.

According to the refining converting rules from 2.2, build the sub-process (N_{TT}, M_{TT0}). The Fig.5 shows the petri-net structure of examination process.

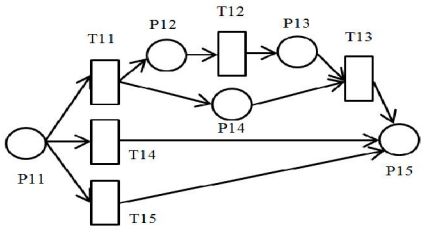

Figure 5: osteosarcoma patient's medical examination processes

In Fig.5 noted (N', M'_{TT0}), the concrete meanings of places and transitions are as follows.

Pl l is the start place;
P15 is the end place;
T11: Analyzing the puncture medical records;
T12: Acquiring the slicer;
T13: Giving primary diagnosis results;
T14: Receiving chest CT;
T15: Bone scanning;

According to the Definition 2, Definition 3 and Definition 4, a conclusion can be drawn that (N_{TT}, $\bar{M}_{TT0}$) and (N, Mo) are bounded and live. Let (N', M'_0) is the Petri-net defined from (N, Mo). According to the Theorem l and theorem 2, a conclusion can be drawn that (N', M_0 ') is bounded and live.

Assume (N_{TT}, M_{TT0}) does not exist in the component library. According to the building strategy from 2.l, when osteosarcoma patients see a doctor, the non-basic transition T5 is defined in the order of step1, step2, step3, step4, step6.

When defining is finished, T5 is defined to sub-Petri net (N_{TT}, M_{TT0}) as illustrated in Fig5. And (N, Mo) is defined to (N', M'), as illustrated in Fig6.

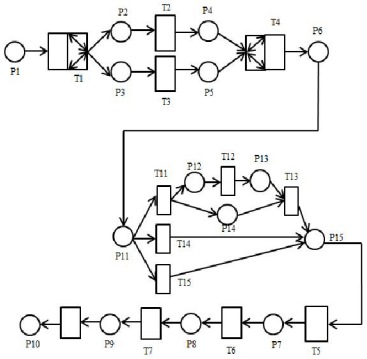

Figure 6: Patient's treatment process

Assume (N_{TT} M_{TT0}) exists in component library.

According to the building strategy from .2.1, when osteosarcoma patients see a doctor, the non-basic transition T5 is defined in the order of step1, step2, step4, step6. (N, M_0) is defined to (N', M'_0), as illustrated in Fig.6. It is clear that the modeling thought of sub process component-based fulfills the reusability of the sub process, and improves the modeling efficiency.

6. Conclusion

In this paper, a modeling method hierarchy refinement Petri-net-based approach for dynamic workflow is adopted; this method can handle both the determining factors and uncertain factors at the same time. The thought of software component is introduced; register the dynamic process into the model library. At the time model for the similar workflow, the prompt in model library can be quoted. In the end, the author model for patient's treatment process, and analyzes the soundness of the model. However, the refining operation of places and transitions has strict demand to the sub-net. At the time that model for the sub-net, the refining rules adopted in the paper only avoid most typical operation of unreasonable structure. In order to guarantee the soundness of model, the refined sub-net and main-net should be validated in the flexibility and roundedness. Therefore, how to improve refining rules of places and transitions to guarantee the correctness of the Petri-net will be the main direction of future work.

Corresponding Author:
Mortaza Mokhtari Nazarlou
E-mail: mortezamokhtari@ymail.com

References

1. Chiu DKW, Li Q, Karlapalem K.A. mata modeling approach to workflow management system supporting exception handling. Information System, 1999, 24(2):159-l 84.
2. Haake JM, Wang WG. Flexible support for business process: extending cooperative hypermedia with process support. Information and Software Technology, 1999, 41 (6):355-366.
3. Casati F, Ceri S, Pernici B, Pozzi G.Workflow evolution. Data and Knowledge Engineering, l 998, 24(3):211-138.
4. Reichert M, Dadam P. ADEPT flex supporting dynamic changes of workflows without 1 oosing control. Journal of Intelligent Information Systems, 1998, 10(2):93-129.
5. Workflow Management Coalition. Interface 1: process definition interchange, process model [Z]. Hampshire, UK: Winchester, 1998.
6. SADIQ S, SADIQ W, ORLOWSKA M. Pockets of flexibility in workflow specifications [A]. Proceedings of the 2gth International Conference on Conceptual Modeling[C]. Yokohama, Japan: Springer 2001. 513-526.
7. Chrzastow ski-Wachtel P, Benatallah B, Hamadi R, O'Dell M, Susanto A. A top-down Petri net-based approach for dynamic workflow modeling. In: van der Aalst WMP, et al., eds. BPM 2003.LNCS 2678, Berlin, Heidelberg: Springer- Verlag, 2003. 336- 353.
8. XIA Chuan-Liang, JIAO Li, LU Wei-Ming. Petri Net Refinement and Its Application in System Design [J]. Journal of Software, Vol. 17, NO.1. January 2006, PP.11-19.
9. H.M.W vetheek, T.Basten, w.M.P.vander Aalst. Diagnosing workflow processes using Woflan [M], The Computer Journal, 2001, 44(4), pp: 246- 279.

Mortaza Mokhtari Nazarlou received his B.Sc. from Islamic Azad University, Tabriz Branch in software engineering and currently he receive M.Sc. from Islamic Azad University, Shabestar Branch in Computer science. also, he is lecturer at Islamic Azad University, Urmia and Maku Branch. His main research interests are Petri net, software testing, Color Petri nets, verification and validation.

12/20/2012