Developing a Learning Notification Express Delivery Service for Students based on Petri-Net

Tzu-Husan Wu, Rita Kuo¹, Maiga Chang², Jia-Sheng Heh
Department of Information and Computer Engineering, Chung-Yuan Christian University
¹Department of Electronic Engineering, Chung-Yuan Christian University
²National Science & Technology Program for e-Learning in Taiwan
mezi@mcsl.ice.cycu.edu.tw, rita@mcsl.ice.cycu.edu.tw, maiga@ms2.hinet.net, jsheh@ice.cycu.edu.tw

Abstract

This paper designs a learning notification express delivery service for providing students information about their learning status, their next learning activities, and even reminder notes. The learning notification express delivery service uses Petri-net for modeling learning activities. In order to make students knowing what status they are and what activities they need to do during the learning process, a learning planner is also developed to make teachers feel more comfortable in planning their teaching schedule.

Keywords: Petri-net, Computer Assisted Learning, Learning Feedback

Petri-net is a directed, bipartite graph in which nodes are either transitions or states and could be defined as:

\[ P \text{(Petri-net)} = (T, S, A) \]

\( S \) is defined as a set of states \((s_i)\) presenting as a state or a condition. The state number in a Petri-net \( P \) can be denoted as \( n \). When a state is transformed from one state to another, the transformation event is called as a transition, which is denoted as \( t_{n+k} \in T \), where \( k > 0 \)

The last part, \( A \), is a set of arc connecting states and transitions.

This paper develops a learning notification delivery service to give students useful information about when and what learning activity they should do. Figure 1 shows a practical learning activity example, “Delivering Homework”, based on Petri-net modeling. When a token is at \( s_1 \), the student has no homework. After teacher assigns a homework \( (t_5) \), the token would be transited to \( s_2 \) (have homework). If the student delivers his/her homework on time \((t_8)\), the token would go to \( s_4 \) (homework finished). Otherwise, if the student could not finish it before the deadline \((t_6)\), the token would go to \( s_3 \), “Homework Tardiness” state.

Some additional states and transitions are added into the original Petri-net-based learning activity to make the learning notification express delivery service get knowing what kind of information should be used to inform the student. Figure 2 shows additional two states and three transitions are added into the original Petri-net and insert between the transition \( t_{n+k} \) and the destination state \( s_j \), where \( s'r_{n+k} \) is a state to check the notification rule used by the transition \( t_{n+k} \).

If there is no rule here, the token would be able to go to the destination state \( s_j \) through the transition \( t'_{w+k} \). If there is any rule for the transition \( t_{n+k} \), the
token would be sent to the state $s_{n+k}'$ through the transition $t_{n+k}'$ (has notification rule(s) for the transition $t_{n+k}$). $s_{n+k}'$ is a decision state to decide which the notification rule for $t_{n+k}$ should be fired. After the learning notification express delivery service sends a short message to the student via mobile phone, the token then transfer to the destination state $s_j$ automatically through the transition $t_{n+k}'$.

To make teachers feel comfortable to use the service, a learning planner is also implemented. For this reason, some additional variables should be considered. Figure 3 is a complete Petri-net-based learning notification express delivery service. The bold lines represent the main learning process. Each transition (from $t_1$ to $t_4$) would cause additional related states and transitions with normal line. A variable, $f_{n+k}$, is used to indicate whether the notification action should be activated or not (just like $f_5$ ~ $f_9$ in Figure 3). Other useful variables such as check point ($c_3$) is also added in the learning planner marked as blocks with dotted line. This variable is used for teacher setting the triggering-time of each feedback function.

![Figure 3. Learning Process Model](image)

Figure 4 is a learning plan example according to Figure 3. The first column is the parameters needed in the learning process model and the second column is an example. The first two parameters (title and content) are the learning plan descriptions. The learning flow controller uses the three check point settings inside the bold table frame. The rest parameters are notification action flags. If teachers do not want the service delivers learning notification to students at a transition, they could dis-check the correspondent notification action flag.

![Figure 4. Parameters example of learning activity in “Homework” learning plan](image)

In the end of this paper, Figure 5 shows an example for two students passing through different ways after the teacher creating a new learning process based on Figure 4 and Figure 5 on 2006/1/7. Both Alex and Buddy stayed at the state - “No Learning Activity ($s_1$)” from 2006/1/7 to 2006/1/9. When the date 2006/1/10 came, Alex and Buddy passed through the transition “Activity Start ($t_6$)” and stayed in the state - “Execute Learning Activity ($s_3$)”. Because the “Activity Start Notification ($f_5$)” is checked, students would receive a notification notes from the system. On 2006/1/13, Alex had already delivered his homework and went to the state - “Learning Activity Finished ($s_4$)” through the transition “Activity Complete ($t_8$)” without any notification, because the notification action flag $f_8$ was dis-checked according to Figure 4.

![Figure 5. Example of two students’ learning processes based on the predefined learning plan](image)

Buddy didn’t finish the homework on time, so he would be forced to go to the state - “Redeem Activity ($s_3$)” through the transition “Activity Incomplete ($t_6$)”. He would also receive a remind message from the service, because the notification action flag $f_6$ (“Activity Incomplete Notification”) was checked. After Buddy delivered his homework, the system would check the homework delivering status on 2006/1/16 and transferred him to the state - “Learning Activity Finished ($s_4$)” through the transition - “Redeem Activity ($t_7$)”. 