A flexible Workflow System for the enactement of pedagogical scenarii

Thomas Vantroys*,†, Yvan Peter*

*Université des Sciences et Technologies de Lille
Laboratoire TRIGONE – Équipe NOCE
Cité scientifique, Bât B6
59655 Villeneuve d'Ascq - France
{thomas.vantroys,yvan.peter}@univ-lille1.fr

†Archimed SA
49 Boulevard de Strasbourg
59042 Lille - France
t.vantroys@archimed.fr

Abstract

Open and Distance Learning platforms require mechanisms for the enactment and coordination of pedagogical modules and learning activities. For that purpose, they can use workflow management systems. These systems formerly reserved for highly structured procedures can be used in dynamic and reactive environments such as virtual campuses platforms, thanks to a enhanced flexibility in the execution of models and in the management of exceptions. In this article we shall present COW our flexible workflow engine dedicated to open and distant learning. We shall see how it is possible to organize the pedagogical modules and the learning paths to answer the expectation within the framework of individual courses (lifelong learning orientation) or within the framework of group courses (closer to the traditional face to face learning).

1 Introduction

Managing the schedule of the activities performed by students can be a daunting task for tutors and teachers especially if there are a large number of students. They have to keep track of each student's progress to provide them with new activities. Workflow systems are dedicated to this kind of work, however, even in case of virtual classroom style of distance learning, one would like to be able to adapt the learning path of a particular student to better fit his needs. This kind of flexibility is unfortunately not well handled by current workflow systems. For this reason, we have decided to start the development of a flexible workflow system dedicated to distance learning. In this article, we will present the features of our workflow system and its interest inside a virtual campus. We will describe how pedagogical modules and learning paths can be organized to support both individual learning paths (lifelong learning orientation) and group based courses (closer to the traditional face to face learning).

2 Flexibility and Workflow systems

2.1 Systèmes de workflow et formation ouverte à distance

À notre connaissance il existe assez peu de plates-formes d'éducation à distance utilisant des moteurs de workflows. Pour illustrer, nous en verrons deux qui visent des publics différents. La première, issue de travaux de recherches du DSTC, est utilisée dans une université australienne. La deuxième plate-forme, qui est industrielle, sert de cadre à nos travaux.

2.1.1 Flex-eL

The Flexible e-learning system (Flex-eL) has been realized by the Distributed Systems Technology Center (DSTC) [11] and the University of Queensland in Australia. It consists of a distance education platform supported by a flexible workflow system. The focus of this work is on supporting the learning process rather than technology with the aim of designing a new learning environment to support new ways of learning. The project started in march 2000 and the environment is currently in test at the University of Queensland in a course of a Master of Information Technology. According to the authors the first results seem positive since there are few abandons. Flex-El is particularly dedicated to the field of lifelong learning

where the student builds his own learning path dynamically and does not handle temporal constraints.

2.1.2 Campus Virtuel

The virtual campus platform named " Campus Virtuel "^{TM} [1] developped by Archimed SA also integrates a workflow system to handle the steps of a course module (figure 1).

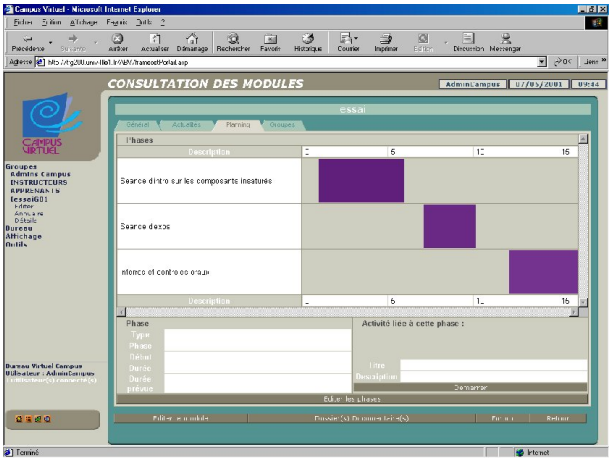

Figure 1: Exemple de modélisation dans le " Campus Virtuel^{TM} "

A teacher or tutor can break his course into steps and associate the documents to use to each step. The system is backed by an Electronic Documents Management System to manage and provide the documents in the different steps. The system is limited by the fact that it can not handle individual work since every students in a group must have terminated the current step to be able to access the next one, even though some of the steps could be realised more individually

3 Cooperative Open Workflow

L'objectif de la plate-forme COW, pour *Cooperative Open Workflow*, est d'offrir un support d'exécution pour des workflows flexibles adaptés à l'éducation à distance. Nous voulons gérer un groupe d'étudiants effectuant un module de formation tout en permettant une possible personnalisation du cheminement d'un étudiant au sein de ce module. L'autre aspect important pour nous est la collaboration entre étudiants et avec l'enseignant.

Pour gérer les instances de processus et pour les modifier dynamiquement, nous disposons d'un outil graphique (figure 2). Il nous permet notamment de modifier graphiquement les modèles des instances de processus.

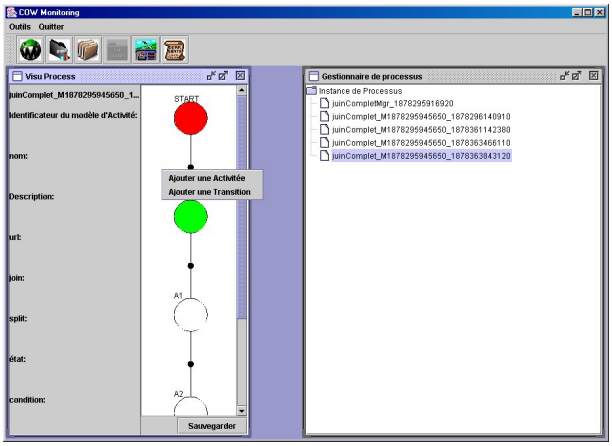

Figure 2: Outil de gestion des instances de processus.

4 Course Modelization and pedagogical modules

4.1 Pedagogical modules

The main function of the worflow system is to schedule the steps of a pedagogical module. Such a module is attended by a group of students (ranging from 1 to n)

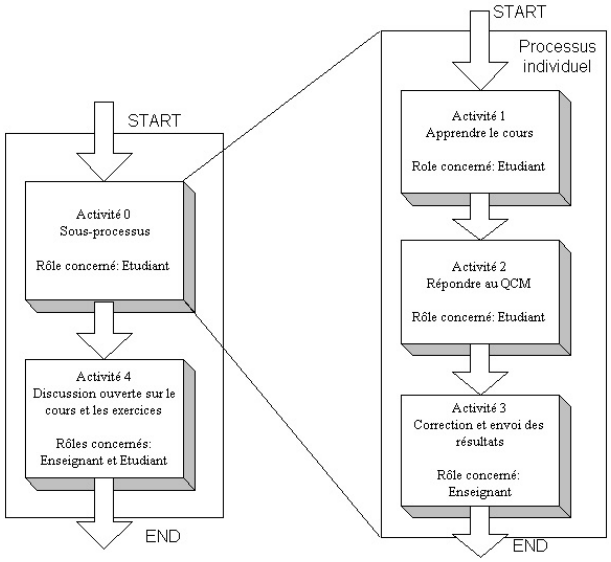

Figure 3: Modelization of pedagogical module

4.1.1 Description of the sample scenario

In the sequel, we will take a course in physics as an exemple of the modelization of a module. This module is broken into four steps described hereafter :

- *course learning* activity associated to the role *learner*;

- *exercices* activity associated to the role *learner*;
- *exercices correction* activity associated to the role *Tutor* ;
- *discussion about the module* activity associated to the roles *learner* and *Tutor*.

4.1.2 Course model

Since some parts of the module can be realised at his own rythm by each student, one has to take in into account so as to enable flexibility in the schedule of the activities. There are two ways to manage the schedule of the activities for a group of students :

- In the first mode, an activity is terminated only when all the students have terminated it. In such a way, the activities of a whole group are synchronised. Even though it is very close to traditional face to face learning, it does not take benefit of distance learning mode;
- The second mode identifies the parts of a module that can be realized autonomously. This way each student can progress at his own rythm inside a group with some activities giving a synchronization point to the group.

These two modes are supported in COW by the mean of sub-processes. In our scenario, the tutor decides that the three first activities can be realised individually by each student. These activities are then modelised into a single process. The process corresponding to the whole module is then composed of two sequential activities. The first one being in fact a reference to the individual work sub-process and the second one corresponding to a synchronous discussion beetwen the members of the group.

4.1.3 Time constraints

Management of time constraints is an important aspect of learning activities. This is particularly true for group based learning where there must not be too much lag between the learners. COW supports the notion of *deadlines* which correspond to the time at which an activity must be started or terminated. It also supports a notion of *limit* which defines the minimal or maximal duration of an activity. When a stop deadline or a maximum limit is reached, the workflow suspends the activity. It can then use different policies to handle the case. For example, the workflow can terminate the activity authoritarily or notify the tutor how will take a decision about it.

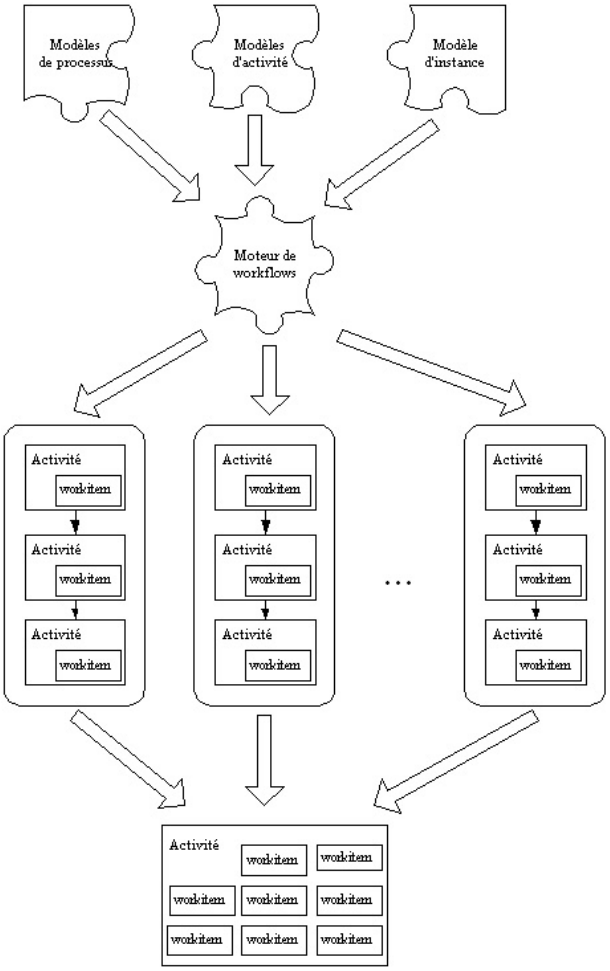

Figure 4: Instanciation des modèles

4.1.4 Course instance

The creation of a process instance requires an instance model which describes the mapping of roles to actual users and of resources to tools. The mappings can be global to the model or defined on an activity basis. This separation between process model and instance data allows for a better reuse of models. Figure 4 illustrates the workflow operation. Taking the process and activities models ans instance data, the workflow engine will create a subprocess for each learner. Theses subprocesses contains the three activities that can be performed individually and each activity contains only one workitem. The third activity is performed by the tutor role who will have three workitems to do from three different processes. When all subprocesses are terminated, the engine will create a collaborative activity with one workitem for each learner and one for the tutor.

4.2 Parcours de formation dynamique

Dans le cadre de la formation continue, la difficulté principale réside dans l'impossibilité de connaître le

processus global que suivra un apprenant. Nous nous sommes inspirés des travaux de la plate-forme Flex-eL.

Pour résoudre ce problème, nous disposons d'un modèle de processus comprenant les étapes initiales de tout apprenant (Inscription, paiement, ...) suivi d'une activité de choix dans laquelle l'apprenant décidera des modules d'enseignement qu'il souhaite suivre.

Ainsi, l'étudiant se crée un parcours personnalisé. Nous stockons les parcours dans l'objectif d'analyser le comportement et le cheminement de l'étudiant afin de comprendre ses choix, et ainsi cristalliser l'expérience pour proposer un parcours semblable à des utilisateurs ayant le même profil. Cette partie est en cours de réalisation.

5 Conclusions et perspectives

Dans cet article, nous avons vu que les systèmes de workflows, grâce à leur flexibilité, peuvent être utilisés pour la coordination dans un système d'éducation à distance. Nous avons présenté les méthodes que nous utilisons pour représenter les modèles de formations et leurs instanciations dans COW, notre moteur de workflows.

Actuellement, nous intégrons notre moteur dans la plate-forme de campus virtuel de la société Archimed. Cela nous permettra de réaliser des expérimentations grandeur nature en utilisant les ressources gérées par le Campus Virtuel.

La modélisation des processus d'enseignement reste un problème non réellement résolu. Nous avons montré qu'il était possible de modéliser un cours, mais cela demande certaines compétences en informatique et notre modèle n'exprime pas de manière suffisamment compréhensible les différents éléments d'un enseignement. Pour répondre à ce problème, nous débutons un travail portant sur le langage EML (Educational Modelling Language) [16] qui a pour objectif la réalisation d'un modeleur graphique d'enseignement générant des modèles au format EML et une traduction des concepts vers notre langage de description de workflows.

Un autre type de flexibilité que nous voulons traiter au sein du laboratoire porte sur la possibilité d'accéder à une plate-forme d'enseignement de manière personnalisée via différents types de périphériques. Pour cela, nous avons initié un travail avec les spécialistes IHM de notre laboratoire pour fournir des interfaces utilisateurs personnalisables et adaptables au matériel (PC, PDA, GSM, ...). L'architecture que nous avons mise en place est décrite dans [17].

Remerciements

Une partie du travail présenté dans cet article est financée par la société Archimed (`http://www.archimed.fr`).

Bibliographie

1. Claude Viéville – Learning Activities in a Virtual Campus, Chapitre 15 dans *The Digital University - Building a learning Community*, Reza Hazemi et Stephen Hailes (Eds), Springer, pages 215-227, 2002,
2. Object Management Group (OMG) – Workflow Management Facility Specification, Version 1.2, 2000. `http://www.omg.org/technology/documents/formal/workflow_management.htm`.
3. Vlada Matena et Mark Hapner – Enterprise JavaBeans^{TM} Specification Version 1.1. Sun Microsystems, mai 1999.
4. Workflow Management Coalition. `http://www.wfmc.org`.
5. Workflow Management Coalition – Workflow Process Definition Interface - XML Process Definition Language, 22 may 2001. `http://www.wfmc.org/standards/docs/xpdl_010522..pdf`.
6. Workflow Management Coalition – Terminology & Glossary. WfMC-TC-1011, Version 3.0, février 1999 .`http://www.wfmc.org/standards/docs/TC-1011_term_glossary_v3.pdf`.
7. Workflow Management Coalition – The Workflow Reference Model. WfMC-TC-1003, Version 1.1, 19 janvier 1995.
8. Gregor Kiczales, Jim des Rivières et Daniel G. Bobrow – The Art of the Metaobject Protocol. MIT Press, septembre 1991. ISBN : 0-26-261074-4.
9. Johann Eder et Walter Liebhart – The workflow activity model WAMO. In *Proceedings of the third international conference on cooperative information systems (Coopis'95)*, Wien, Autriche, mai 1995.
10. Olivera Marjanovic et Maria E. Orlowska – Making Flexible Learning More Flexible. IEEE International Workshop on Advanced Learning Technologies IWALT'2000, décembre 2000. `http://flex-el.com`.
11. Distributed Systems Technology Centre `http://www.dstc.edu.au`.
12. Archimed SA. `http://www.Archimed.fr`.
13. Spécification XML Protocol. `http://www.w3.org/2000/xp/`.
14. Web Services Description Language (WSDL) 1.1. `http://www.w3.org/TR/wsdl`.
15. Andrew Tanenbaum – Systèmes d'exploitation: Systèmes centralisés, Systèmes distribués. 1994, InterEditions.
16. Rob Koper – Modeling units of study from a pedagogical perspective, the pedagogical meta-model behind EML. `http://eml.ou.nl/introduction/docs/ped-metamodel.pdf`
17. Thomas Vantroys et José Rouillard – Workflow and Mobile Devices in Open Distance Learning. IEEE International Conference on Advanced Learning Technologies ICALT 2002, Kazan, Tatarstan, 9-12 septembre 2002. `http://lttf.ieee.org/icalt2002/`.