TEM Journal. Volume 9, Issue 2, Pages 613-618, ISSN 2217-8309, DOI: 10.18421/TEM92-25, May 2020.

Adaptive Workplace E-Learning Model

George Pashev, Liliyana Rusenova, George Totkov, Silvia Gaftandzhieva

University of Plovdiv "Paisii Hilendarski", 24 Tzar Assen Str., Plovdiv, Bulgaria

Abstract **– The paper presents the main concepts of an approach which can be used in a corporate environment for Adaptive education of employees. A formal model is proposed to facilitate adaptivity in terms of pursuing personal and corporate goals. Software prototypes, which prove the applicability of the model, are presented and discussed.**

Keywords **– Adaptive E-learning, Workplace, SORM Model, Learning System.**

1. Introduction

Workplace training is a dynamic process that can take place in practice as a face-to-face mentor training or as an e-learning self-study.

The technological development and a large number of e-learning environments, as well as their advantages (flexibility, delivery of content at anytime and anywhere, on any type of electronic devices, reusability, etc.), give rise to computer-aided learning, which is inherently self-study. Technology-assisted learning has many components, but the most important ones that directly affect the end result are:

- Participants in the learning process: trainee; trainer; manager;
- Learning objectives: individual goals of the learner; corporate goals (defined by the manager); overlapping of individual and corporate goals;
- The context of the workplace in which the training takes place;

DOI: 10.18421/TEM92-25
https://doi.org/10.18421/TEM92-25

Corresponding author: George Pashev,
University of Plovdiv "Paisii Hilendarski".
Email: georgepashev@uni-plovdiv.bg

Received: 30 March 2020.
Revised: 25 April 2020.
Accepted: 06 May 2020.
Published: 27 May 2020.

- Opportunities for adaptability, flexibility and adaptation of the system in the context of workplace (a type of workplace, employee's stereotype/profile, company goals).

All activities of the company can be represented through business processes, and the processes themselves can be divided into smaller components - sub-processes. In the business process building methodology, this is called "business process decomposition". The purpose of decomposition is very simple - if a large process is difficult to manage, it has to be split into parts. Designing business processes allows us to "disassemble" and "assemble" process components by resizing them.

Adaptive E-learning systems are usually designed in school or university environments. However, learning in the workplace is somewhat different from school or university learning in terms of the learner, learning content, and learning context.

Some practical questions arise:

Q1) What do learners need in a workplace context?
Q2) Can we use the usual pre-task or on-task assessment in the workplace domain?
Q3) What is to be considered learning content in a workplace context?

Some authors [1], [2] rely on the hypothesis that performance measurement results are the employees' indicator for their personalized learning needs. Slizyte and Bakanauskiene [3] have summarized it as a systematic procedure to improve performance by setting performance objectives, assessing performance, collecting and analysing performance data, and utilizing performance results to drive performance development. Flexible performance measurement systems such as EFQM, KPI are proposed. *However, this approach lacks the opportunity of taking into account non-performance based indicators.*

The paper proposes a model which requires the fifth goals mentioned below (G1-G5).

It is our goal (**G1**) to include non-performance based indicators, which can also take part in the e-learning adaptation process.

DeRouin, Fritzsche and Salas [4] review the pros and cons of providing adults some control in the learning content they receive. They provide theoretical aspects for a deeper understanding of learner control in adult Workplace adaptive e-learning. Tyre and von Hippel [5] describes the nature of adaptive learning processes in organizations. They found that adaptation is a situated process, thus, in that different organizational settings contain different kinds of clues about the underlying issues, expose learners to different ideas, and offer different resources for generating and analyzing information. We believe that a more sophisticated Adaptive Workplace E-learning Model should also include the opportunity of setting personal goals (**G2**). Moreover, the adaptation should allow dynamic change of personal goals.

Setting personal learning goals implies the need of a dynamic analysis of how personal goals correspond with the company management goals for the corresponding job positions. The employee can be advised by a virtual assistant how to alter personal goals in order to become more suitable for one or another companywide job position. Moreover, more suitable job positions can be proposed for this learner (dynamic company reorganisation (**G3**)).

Using adaptive approaches in Workplace Learning and Company Management, can bring more security for employees' future in a turbulent economic environment. Adaptation can assist managers in their decisions on how to do optimisations in the company structures by the generation of advice for managers (**G4**).

The model should be flexible enough to allow inclusion of interns, students, young and inexperienced workers. The manager can be advised what type of interns to seek for specific job positions that currently experience employee starvation. On the other hand, interns can be advised how to set or alter their personal goals in order to achieve better chances to be hired in the company after their internship ends. This process of interns' adaptation to company needs will be referred to as **G5**.

The general type of software system that meets the requirements (**G1-G5**) has to provide functionality for:

1. Designing and conceptual presentation of the work processes, company goals, personal goals and the external factors involved in the processes, key indicators that are important for the achievement of the set goals.
2. Digitizing/digitizing documents involved in the organization's document flow, processing and retrieval of metadata, capabilities for semantic indexing of documents, modelling of key success indicators.
3. Design of business process models, modelling of system adaptability.
4. Opportunities to analyse and evaluate the progress made towards achieving the goals, feedback and recommendations of the system.

To maximize the requirements for the success of the training and the achievement of the end goals, we will formulate a conceptual framework for an e-learning system in the workplace (SORM).

2. Our Solution

Following the analysis of good practices in the field of e-learning in the workplace, we propose a SORM model, which is made up of the following layers:

1) Repository;
2) Organization Model;
3) Business Processes;
4) Transport Layer;
5) Interface.

The **Interface layer** is built by the modules, such as *Interface Generator* module defines the conceptual framework of various types of graphical, user interfaces that implement the functionality (defining, editing, deleting data and metadata) serving the activities of:

- defining the metadata of the participating documents in the BP (business process),
- defining the logical structure of an employee's portfolio,
- defining the organizational structure,
- defining the role hierarchy,
- definition of key evaluation indicators.

The listed *activities are performed by employees with different managerial levels in the organizational structure.*

The *Interface Generator* serves for the participation in the training process for three types of employees - managers/executives, experts in a given professional field and trainees.

Manager defines the strategy for the development of the organization and defines the rules and business processes in it. Experts are employees (but not necessarily) with *extensive professional experience and skills* who prepare training materials by following company goals. *Learner trainees* are employees who are new to or need additional qualification or retraining.

The **Transport Layer** contains many services that provide access to the individual functionality of the Business Processes, Organization, and Storage layers. In implementing this layer, we adhere to the good practices proposed by Francisco and Pastor [6], namely a methodology for creating REST-based web services with well-known web engineering methods is described.

The **Business Process Layer** architecture is made up of the following modules:

- ***Business Process Descriptor*** enables a user to define the steps in the business process, their metadata, type (start-up, intermediate, and endpoint), the predicates for the implementation of the functionality built into each step, the relationships between the steps that determine their sequence.
- A ***Goal Descriptor*** allows a user to define the goals that are to be achieved as a result of completing a business process.
- ***Business Process (BP) Generator*** generates the sequence of steps in the business process and the relationships between them.
- ***BP Execution*** implements the BP sequence of steps and controls the constraints imposed.

Data Warehouse Layer consists of Data stores. A data store may be an external storage medium, a local database, or an array of information stored and processed by an external system to which the SORM has access. It is presented with a model made up of two components, distinguishing the stored data according to their purpose - *training of employees* and *servicing the business process*.

Data which support learning activities are: educational sites, units and courses, materials for diagnostics and assessment of the progress of the training, tests, exam tasks, etc.

Data used to execute a BP or returned as a result of it are:

- documents involved in the business process and their metadata;
- actions involved in the business process;
- model of the organizational structure;
- definition of the hierarchy of roles involved in BP;
- instances of user profiles;
- instances of business processes;
- a nomenclature of "elementary actions" participating in the PD;
- role-playing links - a set of elementary actions;
- a nomenclature of "key competences";
- a nomenclature with "key indicators" and their weights for assessing progress towards business goals;
- learning outcomes;
- generated recommendations.

The SORM architecture has the following generalized diagram:

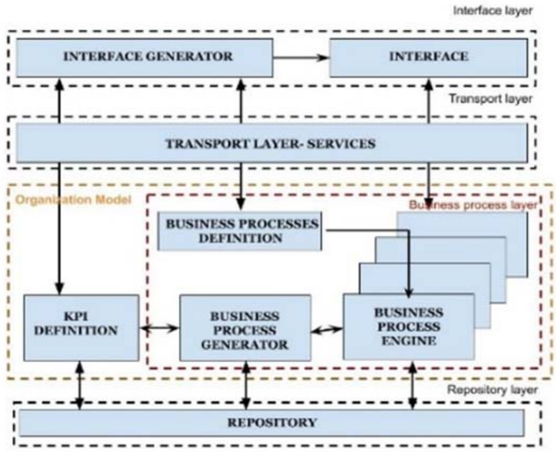

Figure 1. An overview of an Adaptive Workplace E-Learning System Architecture

The **workflow generator** provides the following system functionality:

- workflow description;
- configuring workflow according to specific metadata;
- selection and arrangement of units in a sequence that meets the individual goals of the learner in the context of a work assignment;
- assessing the progress of training (achievement of training objectives);
- updating the trainee's portfolio against the results achieved;
- generating recommendations for new learning purposes.

The workflow generator includes the following modules:

Module 1: Description of workflows.
Module 2: Adaptive generation of workflows within a workflow instance execution.
Module 3: Modelling and 'pursuing' adaptive goals in the context of user role as well as personal goals in the context of individual users/workplace.
Module 4: Modelling and triggering "observational" processes that can extract information about how workflows are performed in the context of the workplace.
Module 5: Modelling and deploying "virtual users" such as "virtual employee", "virtual manager".
Module 6: Adaptive generation of recommendations for further learning purposes.
Module 7: Generation of test units.

Process step model

For the process description, we will define **three types of steps** - initial (start) step, step (current) and final step. More than one start and end step may be involved in the process model, which is dictated by the different starting conditions that can be defined for starting it. The final steps may be more than one depending on the goals we have set and the results achieved in the process.

The StartStep is the first step that contains a predicate that checks the initial conditions (if any) for starting the process. If the value of the ancestor is true, it goes to the next possible step in the process. If the value of the ancestor is false, one of the final steps is performed.

The current Step is a step that defines a part of the business process that involves documents, metadata, and functions.

The EndStep is an end step that finalizes a process that may or may not have a function that returns, as parameters, various endpoints that result from the execution of the process.

We denote by [t] any t, which may also be the empty set Φ.

We call this step model the named N-Fork S = <Docs, Context, Roles, Activities, StepAtrbts, Priorities, [ParentStep], StartDate, EndDate, Spectatior> where Docs is the set of documents involved in the step. Each document is represented by an N-string of metadata, which are <attribute, value> pairs, where the attribute is a descriptive characteristic of the content of the document and value is a specific value of the corresponding characteristic.

Let Doc1 be a document involved in the business process. Then Docs = {Doc_1, ... Doc_n}, Doc_i ∈ Docs; i∈ N; Doc_i = {<$attributes_1$, $value_1$>, ... <$attribute_n$, $value_n$>}, <$attributes_i$, $value_i$> ∈ Doci; i∈ N;. It is possible to attribute = file, a value = ”file_path” containing the path to the specific data being processed as a single unit in the form of a file.

The context of a particular instance of the process model Roles - multiple of the roles involved in the step.

Let there be a predicate with the general type Role (r_i, value), where r_ (i) is an element of the Roles set involved in the process, value is a specific value, the name of an employee who performs the role r_i∈Roles; i∈ N. The Role () predicate will be used to filter the steps that need to be rendered and played by a specific role.

Activities-elementary actions that participating entities can perform. Each Roles has a number of Activities that are eligible for it according to its level in the hierarchy.

Let A: Roles↦ Activities_j x Activities_ (j + 1) x… .x Activities_k such that Activities_j: for each (j∈ [1, k]) Activities_j ∈ Activities; And we will call the role matching function and the eligible Activities.

StepAtrbts = <Name, Type, Value> such that Name is an attribute name, Type is an attribute type, Value is an attribute value valid for the corresponding Type.

Priorities - A set of priority elements: priority∈Priorities that determine the priority of executing steps in a process.

[ParentStep] - A parental step in a process that allows another process to be called, such as a subprocess of the current instance of the process

StartDate - date of the process instance creation

EndDate - the end date of the process that determines the priority in the sequence of steps. The closer the values of StartDate and EndDate are, the higher the precautionary step in the process. In a process where there are several possible next steps for execution with equal Priorities values, which one can be selected, taking into account the value of (EndDate - StartDate).

A spectator-observer who can be a virtual agent or an expert role, overseeing the learning process of roles lower in the hierarchy.

Process instance

The instance model of a process can be represented by a finite state machine FSM = <G, CurrS, State, Marker>, where G-graph model; CurrS-set of possible current steps; State - state of the instance of the process; Marker- is a pair of <attribute, value> that can return a result after executing the step, which can be used to determine the next step to execute in the process.

Model of an edge in a graph

The model of an edge in column G will be represented by a pair Ri = <Si, Sj> where Si, Sj are steps in the process model. Sj is selected as a result of the value of the predicate involved in Si.

Professional Domain Model (PO)

Workplace training refers to a specific professional area that can be viewed as a multidimensional space of concepts to deal with.

Let M be a multidimensional space with dimensions D_1, D_2, ... D_n, n∈N, n≥1 be a Cartesian product D_1 × D_2 × ... × D_n.

It is important to define the links between concepts in the professional field. We will use a hyper graph to represent the model of the professional field.

The M_d model of the PO is an oriented hypergraph H_d = (V_d, E_ed) in which the nodes V_d represent the concepts in the PO, E_d are directed edges defining the precursor-type connections between the concepts, 〚E〛 _hd are the undirected hyper-edges that define belonging to concepts from V_d. (by analogy with [7]).

Model of employee

We present the model of the employee with the N-tuple: M_E = <Employee, Context, Portfolio> where:

- Employee is an ordered pair of Emploee = <name, value> that will accept the value of the name of a particular employee;
- Context = <Department, Position> where Department is name of department of the organization's structure, and Position is position from the hierarchy with corporate roles;
- Portfolio = <PGoals, CGoals, KPIs, MatchGoals> where PGoals are personal training goals;

We will view personal learning goals as a set of KPIs - a set of elements identified as key indicators of success:

- PGoals = {kpi1, kpi2, …… ..kpiN}, i = 1, N;
- CGoals - corporate training goals set for the employee by a senior manager in the hierarchy.

We will consider corporate training goals as a set of elements: CGoals = {kpi1, kpi2, …… ..kpiM}, i = 1, M.

MatchGoals (PGoals, CGoal) is a function that matches personal training goals with corporate ones. It returns as a result indicating the cross-section between two sets of goals, which can be:

- Set PGoals to be a subset of CGoals;
- The set of CGoals is a subset of PGoals;
- The section of PGoals and CGoals is an empty set;
- Set PGoals to match CGoals;
- The sets PGoals and CGoals have a section that is a subset of the elements of the two sets.

The function returns an enum τ of the following:

- No matches (empty set);
- PGoals is a subset of CGoals;
- CGoals is a subset of PGoals;
- CGoals matches PGoals;
- CGoals PGoals have a cross section, a subset of both.

Recommend() - The function takes the value returned by MatchGoals and the common predicates (the intersection of PGoals and CGoals) and CGoals and a boolean parameter, which is used to indicate whether the function is going to generate recommendations for company manager, or for employee/intern/student for specific employee position. It returns a set of recommendations for personal training purposes to the employee or recommendations for company manager.

Parts of *Recommend()* algorithm are shown in the following pseudo-code:

```
Function Recommend(τ, PGoals, CGoals, isForEmployee,
employee, CurrentPosition){
      :employee
        Foreach(CGoals not in PGoals as curGoal){
        generateRecommendationForLackingGoal();
      }
      Foreach(PGoals in CGoals as (personalGoal_I,
companyGoal_I)){
        If(value(personalGoal_I)<value(companyGoal_I)
generateRecommendationForMoreEffortNeeded(personalGoa
l_I);
      }
      If(!isForEmployee){
        //is for manager
        Var
positionGoals=findAllPositionGoalsSortedByDistanceTo(
PGoals);
      If(positionGoals[0].position!=CurrentPosition){

      generateRecommendationForManagerToChangePosition
OfEmployee(employee, positionGoals[0].position);
      }
//…. Some code omitted
      }
}
```

Figure 2. The Recommend algorithm pseudo-code

As it is obvious by Fig. 2 implementations of the *Recommend* function can be used for development of intelligent personal advisors both for company managers and employees/interns/students.

Model of abstract representation of hierarchy of employee roles

We will define the model representing the hierarchy of roles of company employees as a tree (hierarchy) H.

The set H is a root tree (hereinafter referred to as a tree only) if it is an empty set, or if the following conditions are simultaneously fulfilled:

It contains a single element h called the root of the tree, which we denote by root (H).

The remaining elements (except the root) are divided into m (m≥0) non-intersecting empty sets H_1, H_2,…, H_m, each of which is a tree.

A path in a tree is a sequence of vertices h_1, h_2,…., H_n without repetition, and for every two consecutive vertices h_ (i-1), h_i is exactly one of the two: h_i is the heir to h_ (i-1) or h_ (i-1) is the heir of h_i.

The paths (H) will mark the set of all possible paths in tree H leading to the leaf.

The root of the tree represents the highest level in the managerial hierarchy of roles in the company. Each of his heirs, other than a letter, makes management decisions that can overturn the decisions of all his successors. The path in the tree represents administrative decision-making, defining their priority over the role hierarchy.

Workplace model

We will present the workplace model with the N-tuple: WPLC = <M_E, Context, H r, {FSM}> where:

- M_E- employee model;
- Context - organizational context;
- H r-hierarchy of roles in the organization;
- {FSM} - multiple instances of processes in which the employee participates.

Key Competences Framework (KPI-key performance indicators)

The KPI framework defines the ontology of key competencies. This includes the specifications for all positions/positions in the organization that are assigned to a specific role; the indicators for evaluating each position; the set of knowledge required for each indicator (KPI) as well as the relationship / hierarchy between competencies and any possible combinations of competences. The KPI framework defines the relationships between all indicators, including prerequisite, composition, relevancy and inhibitor, with a predicate defined for each opportunity.

The set of predicates consists of <Relationship Name, Predicate, Definition> triplets like:

- *Composition Comp (a, b) Ability a is part of ability b.*
- *Pre-requisite PreCon (a, b) Ability a is a prerequisite for ability b.*
- *Relationship Rel (a, b) Ability a is related to ability b.*
- *Inh inhibitor (a, b) If ability a is utilized, then utilize b.*

The framework is flexible enough to be able to incorporate various well-known taxonomies in addition like Bloom's Taxonomy [8]. In [9] there is a defined algorithm (eLP_GA) and an approach for generating a curriculum/learning paths with given learning goals and current status of the trainee. A similar approach is used for SORM.

3. Conclusion and Future Work

An adaptive workflow e-learning model in the Workplace education domain is proposed. To a significant extent the proposed model for the SORM meets our requirements (**G1-G5**): personal goals are supported, personal advisors for goals setting or for managerial company structural reorganization can also be created, based on the discussed functions for goals comparisons (for example *MatchGoals*() and *Recommend*() functions). Some of the future work includes the development of such intelligent advisors. The model also supports flexible inclusion of interns, unexperienced workers and students. Moreover, students can be advised by such advisors how to alter their personal goals and personal education in order to gain better chance to be hired in a specific company. Such specific intelligent advisors will be presented in a future research. Some partial software experiments have been conducted, based on previous research in the field of adaptive workflow and e-learning paths generation. The proposed algorithm in a previous research (eLP_GA) is flexible enough to decide which goals are easier to be achieved within the current learner's context, a reachability radius and given learning resources, along with the goals that cannot be achieved at all. Better versions of this algorithm in terms of complexity and performance will be proposed in future works.

References

[1]. Jia, H., Wang, M., Ran, W., Yang, S. J., Liao, J., & Chiu, D. K. (2011). Design of a performance-oriented workplace e-learning system using ontology. *Expert Systems with Applications*, *38*(4), 3372-3382.

[2]. Stephanie, S. (2005). Signs of the times. *Training Magazine*.

[3]. Slizyte, A., & Bakanauskiene, I. (2007). Designing performance measurement system in organization. *Organizacij Vadyba: Sisteminiai Tyrimai*, *43*, 135-148.

[4]. DeRouin, R. E., Fritzsche, B. A., & Salas, E. (2005). Learner control and workplace e-learning: Design, person, and organizational issues. *Research in personnel and human resources management*, *24*, 181-214.

[5]. Tyre, M. J., & Von Hippel, E. (1997). The situated nature of adaptive learning in organizations. *Organization science*, *8*(1), 71-83.

[6]. Valverde, F., & Pastor, O. (2009). Dealing with REST services in model-driven web engineering methods. *V Jornadas Científico-Técnicas en Servicios Web y SOA, JSWEB*, 243-250.

[7]. Pashev, G. (2016). Dynamic generation and optimal management of flows of activities and resources for e-learning, PhD Thesis, *University of Plovdiv*, Plovdiv, Bulgaria.

[8]. Krathwohl, D. R., & Anderson, L. W. (2009). *A taxonomy for learning, teaching, and assessing: A revision of Bloom's taxonomy of educational objectives*. Longman.

[9]. Pashev, G., Totkov, G., Kostadinova, H., & Indzhov, H. (2016, June). Personalized Educational Paths through Self-Modifying Learning Objects. In *Proceedings of the 17th International Conference on Computer Systems and Technologies 2016* (pp. 437-444).