*Knauf, Rainer; Böck, Ronald; Sakurai, Yoshitaka; Tsuruta, Setsuo*

***Knowledge engineering with didactic knowledge - first steps towards an ultimate goal***

*Publikation entstand im Rahmen der Veranstaltung:*
Proceedings of 22th International Florida Artificial Intelligence Research Society Conference 2009 (FLAIRS-2009), 19 - 21 May 2009, Sanibel Island, FL, USA

# Knowledge Engineering with Didactic Knowledge
# First Steps towards an Ultimate Goal

**Rainer Knauf**
Ilmenau University of Technology
rainer.knauf@tu-ilmenau.de

**Ronald Boeck**
University of Magdeburg, Germany
ronald.boeck@ovgu.de

**Yoshitaka Sakurai, Setsuo Tsuruta**
Tokyo Denki Unversity, Japan
{ysakurai,tsuruta}@sie.dendai.ac.jp

### Abstract

Generally, learning systems suffer from a lack of an explicit and adaptable didactic design. A previously introduced modeling approach called storyboarding is setting the stage to apply Knowledge Engineering Technologies to verify and validate the didactics behind a learning process. Moreover, didactics can be refined according to revealed weaknesses and proven excellence. Successful didactic patterns can be explored by applying mining techniques to the various ways students went through the storyboard and their associated level of success.

## Introduction

University instruction often suffers from a lack of didactic design. So far, the ad hoc application of didactic skills in teaching situations is not formally modeled for use by less experienced instructors. Moreover, much of such skills are not represented at all, but just “implemented” in the heads of experienced teachers.

To make didactic design explicit, a modeling approach called storyboarding (see (Knauf et al. 2008) for a recent state of this approach) is used here. This (semi-) formal model is setting the stage to apply Knowledge Engineering (KE) Technologies to verify, validate, and refine the didactics behind a learning process.

Here, we focus a verification and a validation approach.

## Verification of Storyboards

Our concept of storyboarding is a semi-formal one. The graph hierarchy is completely formal and below the level of scenes it is completely informal. Thus, the scenes form the interface between the formal and the informal levels.

The formal levels are the key feature to support the logical reliability such as consistency, completeness, non-redundancy, and so on. To ensure consistency and completeness of our storyboards, we developed and implemented several verification procedures:

1. A *Hierarchy Completeness* Test focuses questions such as whether every episode has exactly one related graph and vise versa.
2. Also, a *Path Completeness* Test the reachability of each node (in particular, of the *End Node*) from the *Start Node* is checked.
3. Furthermore, the *Node Soundness* of outgoing edges, i.e. the completeness and consistence of alternative outgoing edges (with the same beginning color), is checked.
4. Edge colors, which express the *Interdependence of Incoming/Outgoing edges*, are also a subject of formal verification by checking, whether (1) there is a unique (beginning) color of the Start node’s outgoing edges and (2) at least one outgoing edge with the same (beginning) color for each incoming edge’s (finishing) colors.



## Validation of Storyboard Paths

The objective of this research is both (1) an a posteriori validation of storyboard paths by considering the success levels achieved by students and (2) an a priori validation of intended (future) storyboard paths by utilizing the results of former students.

For the reader’s convenience, we refrain from formally describing the technology, but provide a small example of a decision tree construction and utilization, which is derived from our application setting at a Japanese university.

For simplicity, we (1) refer to the subject compositions as *episodes* $e$, (2) refer to the particular courses as *scenes* $s$, (3) generalize from concrete episode- and scene names to abstract ones such as $e1, e2, s1, s2, \dots$ and (4) convert the letter-based students’ performance evaluation scale to a numerical (Grade Point) scale ranging from $4$ down to $0$.

**Pre-processing path information** First, each given path is decomposed by recursively replacing episodes by their related sub-graph path until the paths consists of scenes only. Concurrent scenes, i.e. subjects that run in parallel, i.e. in the same semester, are united to a scene set and form <u>one</u> element of the student’s path. As a result, each path is a linear sequence of elements. Attached to this sequence, there is the associated success label composed of the Grade Point Average (GPA) of the student, who went this path. Figure 1 shows a concrete storyboard path along with the result of the flattening procedure.

**Validity estimation** In the above example, the student finalized his study with a Grade Point Average of $3.0$. For

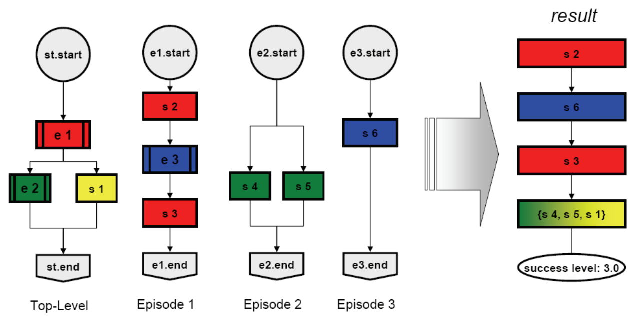


*Figure 1:* A student's path through the (nested) storyboard

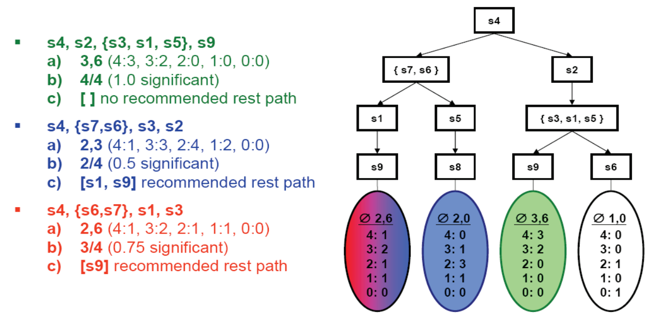


*Figure 3:* Examples of (a) success estimation, (b) its significance, and (c) recommended rest paths

each scene and each episode in the storyboard, the average final Grade Point Result of students, who went through it, can be considered the node's validity.

**Composing a decision tree of paths** Next, a decision tree is constructed. Figure 2 shows the result of the decision tree construction in our application. As illustrated in the figure's left hand side, 17 students went through the storyboard on four different paths (indicated by different colors).

| [ s4, {s6, s7}, s1, s9 ] | | [ s4, {s6, s7}, s5, s8 ] | |
|---|---|---|---|
| 4 | 1 student | 4 | 0 students |
| 3 | 2 students | 3 | 1 student |
| 2 | 1 student | 2 | 3 students |
| 1 | 1 student | 1 | 1 student |
| 0 | 0 students | 0 | 0 students |
| **[s4, s2, {s3, s1, s5}, s9]** | | **[s4, s2, {s3, s1, s5}, s6]** | |
| 4 | 3 students | 4 | 0 students |
| 3 | 2 students | 3 | 0 students |
| 2 | 0 students | 2 | 1 student |
| 1 | 0 students | 1 | 0 students |
| 0 | 0 students | 0 | 1 student |

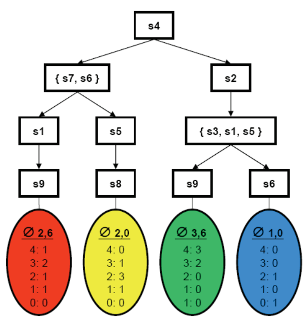


*Figure 2:* Storyboard paths and a derived decision tree

In the derived decision tree each of these four paths form a path in the tree from the root towards a leaf. Attached to each leaf, there is a label node, which holds the success information of the students, who went this path.

**Utilizing the decision tree** Finally, the decision tree is utilized. Figure 3 shows the usage of the decision tree for three submitted paths, one which is represented completely in the decision tree (green) and two, which are not represented completely in the decision tree (blue, red).

The success estimation of the green path is simply performed by providing the related success label of the related path in the tree.

For the blue path, there is no identical path in the tree. Here, the estimation procedure looks for a path within the tree, which has the longest sequence in common with the submitted path. This is $[s4, \{s7, s6\}]$. Since this path has two nodes in common with the submitted one (of four nodes), the significance of the success estimation is calculated by $2/4$, i.e. 0.5.

Behind the node $\{s7, s6\}$, there are two different subtrees, which led to different success degrees by former students, $[s1, s9]$ and $[s5, s8]$. Since the latter is the better one, it is recommended as a rest path to optimize success chances.

For the red path, the usage of the decision tree is performed accordingly.

By practicing this way to utilize a decision tree, we realized that we rarely found a path in the tree, which is completely equivalent to a submitted path. However, if an element of a node that contains a scene set in the tree is not in the related node of the submitted path, it still could be a subject that the student already passed successfully in a previous semester. Therefore, the containment in the decision tree was extended with respect to the educational history of a student. A previously taken course may always be considered as an element of a subsequent node:

Let $P = [P_1, P_2, \ldots, P_n]$ be a path submitted by a student with $P_i$ being a set of courses taken in the $i$-th semester. Let $T = [T_1, T_2, \ldots, T_m]$ be a path that is represented in the decision tree.

$P$ is represented by $T$ ($P \sqsubseteq T$), if all courses of all $P_i$ are in any $T_j$ with $j \leq i$: $\forall i \forall j \; P_i \subseteq \bigcup_{j=1}^{i} T_j$.

## Outlook

Our upcoming work focuses (1) the integration of our system into an existing class schedule system in our application setting (Dohi et al. 2006a) and (2) the integration of a user (student) profile to provide success estimations and refinement suggestions due to a student's individual learning needs, learning desires, preferences and talents.

## References

S. Dohi, S. Nakamura, Y. Sakurai, S. Tsuruta, R. Knauf: Dynamic Learning Need Reflection System for academic education and its applicability to Intelligent Agents. *The 6th IEEE International Conference on Advanced Learning Technologies (ICALT 2006)*, Kerkrade (Netherlands), Los Alamitos, CA: IEEE Computer Society, ISBN 0-7695-2632-2, pp. 459-463, 2006.

R. Knauf, R. Boeck, Y. Sakurai, S. Dohi, S. Tsuruta: Knowledge Mining for Supporting Learning Processes. *The 2008 IEEE International Conference on Systems, Man, and Cybernetics (SMC 2008)*, Singapore, ISBN 978-1-4244-2384-2, Library of Congress: 2008903109, IEEE: Piscataway, NJ, USA, pp. 2615-2621, 2008.