Knauf, Rainer; Böck, Ronald; Sakurai, Yoshitaka; Dohi, Shinichi; Tsuruta, Setsuo :

Knowledge mining for supporting learning processes

Zuerst erschienen in:
IEEE International Conference on Systems, Man and Cybernetics (SMC) ; (Singapore) : 2008.10.12-15. - Piscataway, NJ : IEEE, 2008. - ISBN 978-1-4244-2383-5. - S. 2615-2621
DOI: 10.1109/ICSMC.2008.4811690

Knowledge Mining for Supporting Learning Processes

Rainer Knauf & Ronald Böck
Faculty of Comuter Science and Automation
University of Ilmenau
Ilmenau, Germany
rainer.knauf@tu-ilmenau.de

Yoshitaka Sakurai, Shinichi Dohi & Setsuo Tsuruta
School of Information Environment
Tokyo Denki University
Chiba New Town, Japan
{ysakurai,tsuruta}@sie.dendai.acjp

***Abstract*— AI technologies for knowledge mining are commonly used in technical environments. Their application for social processes like learning processes, for example, is a quite a new challenge, which is characterized by having "humans in the loop". Humans' desires, preferences and decisions may be unpredictable and thus, not appropriate for modeling - at a first glance. However, in learning processes didactic variants can be anticipated and can become a subject of AI technologies. A semi-formal modeling approach called storyboarding, is outlined here. A storyboard represents various opportunities for composing a learning process according to individual circumstances, such as topical prerequisites (educational history), mental prerequisites (preferred learning styles, etc.), performance prerequisites (a requested success level in former learning activities, etc.), and personal aspects (needs, wishes, talents, aims). By storyboarding, various didactic variants can be validated by considering the average learning success associated with the different paths through a storyboard in a case study. Based on validation results, success chances can be derived for the different paths. Here, a concept and an implementation to pre-estimate success chances of intended (future) learning paths through a storyboard are introduced. They are based on a Data Mining technology, and construct a decision tree by analyzing former learners' paths and their degrees of success. Furthermore, this technology generates a supplement to a submitted path, which is optimal according to the success chances. This technology has been tested at a Japanese university, in which students had to compose their individual plan (subject sequences) in advance, and the technology helped them by predicting success chances and suggesting alternatives.**

I. Introduction

The design of learning activities in collegiate instruction is a very interdisciplinary process. Besides deep, topical knowledge in the subject being taught, an instructor needs knowledge and skills in many other subjects. This includes IT-related skills to use today's presentation equipment, didactic skills to effectively present the topical content, plus skills in fields like social sciences, psychology and ergonomics.

In particular, university instruction often suffers from a lack of didactic design. Since universities are also research institutions, their professors are usually hired based on their topical skills. Didactic skills are often underestimated in the recruiting process.

Our approach to facing problems like these is a modeling concept for didactic knowledge called Storyboarding. A storyboard provides a roadmap for a course, including alternative paths and possible detours if certain concepts to be learned need reinforcement. Using modern media technology, a storyboard also plays the role of a server that provides the appropriate content material when deemed required. Our suggestion to ensure a wide dissemination of this concept is to use a standard tool to develop and process this model, which is Microsoft Visio. Additionally, we developed a platform independent web based storyboard development environment [13], which allows the design of storyboards while guaranteeing their logical soundness.

The paper is organized as follows. Section 2 is an introduction to the storyboard concept. In includes the present state of the current development. Section 3 gives an overview on Knowledge Engineering Technologies, which have been developed and implemented for storyboards. In section 4, we summarize the research undertaken so far and outline current work as well as research horizons.

II. The Storyboarding Concept

Storyboarding as a means to model information and learning processes has been introduced around 1998/1999 by Feyer and others [8], [9]. One of the first Storyboarding languages called SiteLang [6] was introduced in 2001. However, this language was dedicated to model Web Information Systems (WIS) and has its related but limited expressivity and it suffers from non usability by non IT experts.

The latter, however, was not a drawback for the application in [6]. There were attempts to apply SiteLang for learning processes [1]. However, this application was limited to e-Learning only. An application to use that language for the reverse purpose, i.e. not to model WIS itself, but to model stories or data contents in WIS was presented in [11].

This work inspired the authors of the present paper to investigate successful didactic patterns with the more general storyboarding approach. However, the limitation to WIS and e-Learning is also revealed by the necessity to formally distinguish media types [7] and to integrate context [12]. A first approach in distinguishing various abstraction layers in storyboards was introduced in [11]. Here, a limited number of different layers were modeled by different means. Again, this suffers from a lack of generality and from an intended application (e.g. WIS different from learning, which is limited to electronic material or the data in internet sites of WIS).

1-4244-2384-2/08/$20.00 © 2008 IEEE

Our storyboard concept is built upon standard concepts which enjoy

- clarity by providing a high-level modeling approach,
- simplicity, which enables everybody to become a storyboard author, and
- visual appearance as graphs.

With respect to a better formal composition, processing, verification, validation and refinement the concept as introduced so far [4] [10] has been further developed. We adopt these modifications. Here, we define a storyboard as follows:

A storyboard is a nested hierarchy of directed graphs with annotated nodes and annotated edges. Nodes are scenes or episodes. Scenes denote leaves of the nesting hierarchy. Episodes denote a sub-graph. There is exactly one Start- and End- node to each (sub) graph. Edges specify transitions between nodes. They may be single-color or bi-color. Nodes and edges have (pre-defined) key attributes and may have free attributes.

The interpretations of these terms follow after presenting a small example.

The representation as a graph (instead of a linear sequence) reflects the fact that different readers trace the paper differently according to their particular interests, prerequisites, a current situation (like being under time pressure), and other circumstances. The story-board is the authors' design document representing expectations of human behavior. For exemplification, Figure 1 shows a top level storyboard on one of the author's conference papers. Alternative paths may be driven by the reader's role:

- The Ilmenau research group may skip the Introduction, the section on storyboarding and the summary, acknowledgements and references, because they are familiar with it. Since the example application on higher level storyboards is new to them, they will study this example.
- The Tokyo research group may skip the introduction and the section on their so-called Dynamic Learning Need Reflection System (DLNRS) [3], [4]for similar reasons. In fact, the DLNRS is their original invention.
  However, they should read the section on storyboarding, because it contains some supplements that are new to them.
  Like the Ilmenau group, they are interested in studying the section on higher level storyboards and skip the rest of the paper.
- Referees (hopefully) want to read all. After the summary, they can read Acknowledgements and References independently in any sequence. For their duty they have to check the References at least.

A storyboard can be traversed in different manners according to

- users' interests, objectives, and desires,
- didactic preferences[1],

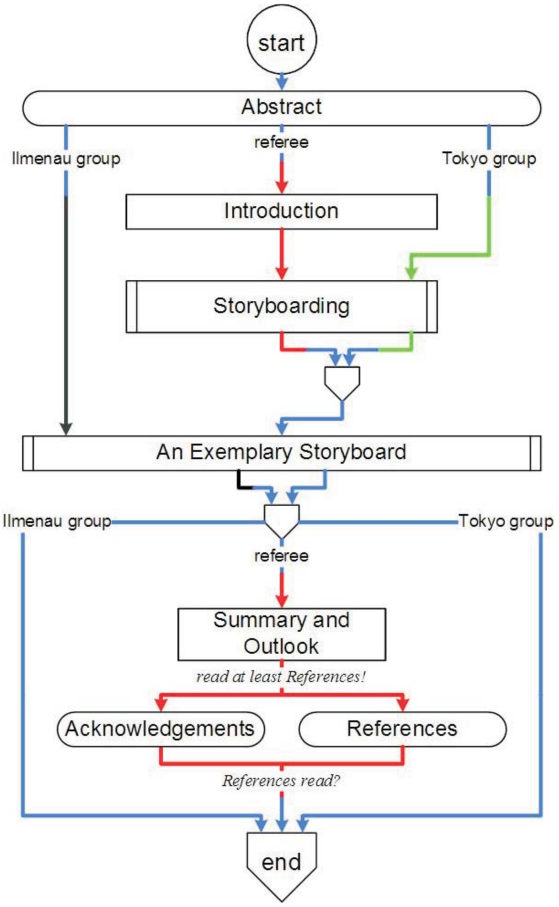


Fig. 1. A Storyboard on a Conference Paper

- the sequence of nodes (and other storyboards) visited before (i.e. according to the educational history),
- available resources (like time, money, equipment to present material, and so on) and
- other application driven circumstances.

In fact, people may read the present paper in ways that are different from our assumptions modeled in Figure 1. However, for the ways we anticipate, we can ensure that they are coherent. A storyboard may be seen as a model of an anticipated reception process that is interpreted as follows:

- *Scenes* denote a non-decomposable learning activity that can be implemented in any way. It can be the presentation of a (media) document, opening a tool that supports learning (URL or e-learning system) or an informal activity description.
- *Episodes* are defined by their sub-graph.
- *Graphs are* interpreted by the paths, on which they can be *traversed*.
- A *Start Node* of a (sub-) graph defines the starting point

[1] In the authors' experience, some students understand better by presenting illustrations, others by providing a small example and others by providing formal descriptions.

of a legal graph traversing.

- An *End Node* of a (sub-) graph defines the final target point of a legal graph traversing.
- *Edges* denote transitions between nodes. There are rules to leave a node by an outgoing edge:
  1) The outgoing edge must have the same color as the in-coming edge by which the node was reached.
  2) If there is a condition specified as the edge's key attribute, this condition has to be met for leaving the node by this edge.
- *Key attributes of nodes* specify application driven information, which is necessary for all nodes of the same type, e.g. actors and locations.
- *Key attributes of edges* specify conditions, which have to be true for traversing on this edge.
- *Free attributes* specify whatever the storyboard author wants the user to know: didactic intentions, useful methods, necessary equipment, e.g.

The types of nodes and edges in our storyboard implementations are shown in tables I and II.

What are peculiarities of the concept? At a first view, this purpose is similar to the purpose of traditional storyboards that are produced for shows, plays, theater games or movies, i.e. visual arts. The materials and tools of the storyboarded learning activities (e.g., text books, scripts, slides, hard- and software models, e-learning systems and others) are something comparable to the requisites of a show. Basic differences of our storyboards to those used to "specify" a show are:

- the primary purpose (learning vs. entertainment),
- the degree of formalization, and, as a consequence of being semi-formal,
- the obligation of everything above the level of scenes, which does (and should) not apply to storyboards in arts, in which the intendant has some freedom of individual interpretation and
- (thanks to formalization) the opportunity to formally represent, process, evaluate, and refine our storyboards, which does not apply at all to story-boards in visual arts.

Also, Storyboards have somewhat in common with classic AI knowledge representations like *Semantic Networks* and *Frames* as well as with process modeling languages like *State Diagrams* and *Petri Nets* (see e.g. [1] for use in learning processes), *Workflow Diagrams* (see e.g. [14] for use in learning processes) and *Flow Charts* (see e.g. [16] for use in learning processes). Items that make this concept more expressive for didactic knowledge than representations as mentioned above are

- the potentially unlimited nesting of graphs,
- the opportunity to express "conditioned" edges by using the colors (bi-colored edges, e.g.) or respective key annotations to edges,
- the opportunity to use (two kinds of) fork-edges,
- the potential of nodes to carry many different teaching materials and tools as hyperlinks[2], and, most important, and
- the fact that a scene can be implemented in any way, i.e. is not restricted to something electronically available or even formally structured (like any knowledge representations and any material included in process models).

III. Knowledge Mining over Storyboard Paths

A basic objective of this storyboard application is to use Knowledge Engineering technologies on the (semi-) formal process models.

The objective of this research is inductively "learning" successful storyboard patterns and recommendable paths. This is some sort of meta-learning, i.e. the learning of learning knowledge. This is performed by an analysis of the paths where former students went through the storyboard and it is based on their success that is associated with these particular paths [2].

To exemplary show the feasibility and benefit of high level storyboarding for its qualified assistance of students suffering from the "jungle of opportunities and constraints" in university education, a simple prototype was recently developed to evaluate curricula created or modified by the students in advance of their study [2].

For this purpose, we introduced a concept to estimate success chances of curricula, which are composed by students at the School of Information Environment of the Tokyo Denki University in their curriculum planning class in the first semester. Since the storyboard representation enjoys a certain degree of formality, there is an opportunity to apply data mining techniques on storyboarding paths that have been used by students. Furthermore, these paths can be associated with the student's related success, i.e. his/her final result of the study. Based on these examples, the success chance of intended paths can be estimated as follows. The concept is described in detail in [2]. Furthermore, [2] contains a prototypical implementation in Prolog, which shows its applicability.

A. Construction of a decision tree

The construction of the decision tree is based on the paths of former students through the storyboards which model the "space of opportunities", in which the students took a particular one, which is a path through the storyboard.

Each of those paths can be associated with the degree of success, which has been achieved by the student. In case a set of students went the same path, the degree of success can be estimated by a weighted average degree of them.

More concretely, this path begins at the Start Node of the top level storyboard and terminates at its End Node. Each episode on this path is replaced by its sub-graph. This replacement continues throughout the entire hierarchy of nested graphs. Figuratively speaking, the decision tree is constructed

[2] The author developed a storyboard for an AI course at an US university and included material of his own AI course in Germany. Now, this storyboard serves both universities and is also a common platform for internationally sharing teaching materials.

TABLE I
NODE TYPES

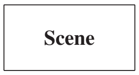
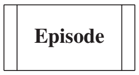
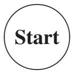
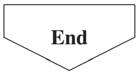
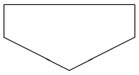
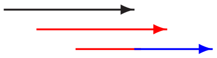
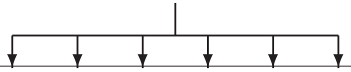
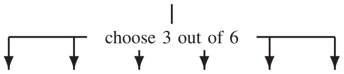
	Symbol	Behavior when	
		following a hyperlink	double clicked
Scene		• opening a document (*.doc, *.pdf, *.wav, *.vsd, *.ppt, *.xls , . . .) with the appropriate tool • nothing, if just verbally described scene	• opening a document (*.doc, *.pdf, *.wav, *.vsd, *.ppt, *.xls , . . .) with the appropriate tool • visiting a website with the standard browser in case it is an URL • opening the standard mail tool in case it is an e-mail address
Episode		opening a subgraph that specifies the episode	• opening a document (*.doc, *.pdf, *.wav, *.vsd, *.ppt, *.xls , . . .) with the appropriate tool • visiting a website with the standard browser in case it is an URL • opening the standard mail tool in case it is an e-mail address
Start Node		jumping to the start node of the related super-graph	not meaningful
End Node		jumping to the *Reference Node* that successes it's associated *Episode* Node in the related super-graph	not meaningful
Reference Node		jumping to the *End Node* of the sub-graph that is associated to the preceded *Episode* Node	not meaningful

TABLE II
EDGE TYPES

	Symbol	Interpretation
Simple Edge		defines a unique successor node
Fork		defines several successor nodes, which have to be traversed independently from each other, i.e. in any sequence or parallel
Fork with conditions		defines several successor nodes, which have to be traversed independently from each other, i.e. in any sequence or parallel, according to the specified condition, e.g. take n out of m paths

on the basis of a "flatten" storyboard. Flatten, in this context, means the graph hierarchy "flattened down" to just one level with no sub-graph. As a result, such a path comes to atomic scenes only [2].

Since it may happen that nodes of different sub-storyboards carry identical names (like *Exercise # 1*, *Example*, e.g.), the nodes in the storyboard hierarchy must receive unique names. This has been performed by adding the upper episode names as a prefix and separating the prefixes by a dot, e.g. *e2.e5.s1* could by the new name of Scene *s1* in Episode *e5*, if *e5* is in the sub-storyboard of episode *e2*.

The decision tree is based on the concept of bundling common starting sequences [2] of the various paths to a node of the tree. In [2], these starting sequences are called "least common denominator". Of course, all paths traversed by the students begin with the Start Node that forms the root of the decision tree.

Different subsequent following (next) nodes of the paths will result in different sub-trees right below the actual root on the last node of the common starting sequence. This continues for each lower level sub-tree accordingly. If there are different paths with a common starting sequence from the root to the actual root different in the next (subsequent) nodes, related sub-trees will be established.

Each node in this tree that represents a final node of a path, is followed by a label-node. Label-nodes contain a list of marks that students received after going through this path. Each mark is along with the number of occurrences (the number of students getting the mark).

Additionally, the weighted arithmetic average (WAA) value of these marks is also attached with this label. The value of WAA serves as an estimate of success chances for future students that plan to go through the same path.

The WAA along with the distribution of received marks for the related path is represented as a fact for the related leaf node in the decision tree. For exemplification, in the fact

- *assess([1,2,2,1],marks(1.45,[mark(1,14),mark(2,5),mark(5,1)])).*

the list $[1, 2, 2, 1]$ represents the position of the node within the tree, 1.45 is the WAA, and the subsequent list states, that mark 1 has been received by 14 students, mark 2 by 5 students, and mark 5 by 1 student.[3]

Since subjects of a semester are usually visited concurrently (which is represented by the fork edges, see Table II, for the planning of them on the storyboard level considers them as a single node. We represented the decision tree as a set of facts

- *tree(Depth,Position_in_tree, List_of_concurrent_subjects).*

Here, the position is represented like in the above *assess* - Facts by a position identifier, and *List_of_concurrent_subjects* specifies the subjects to study in concurrence.

A new path is added to the tree by simultaneously traversing the path's subject sequence and the decision tree down from the root until

1) the path is finished or
2) there is a "next node" in the path that is different from all "next sub-tree roots".

In the first case, the related new assess - fact at its leaf position is updated accordingly or added to the knowledge base, if there was no one so far. In the latter case, a new sub-tree is made out of the remaining path and hooked into the tree.

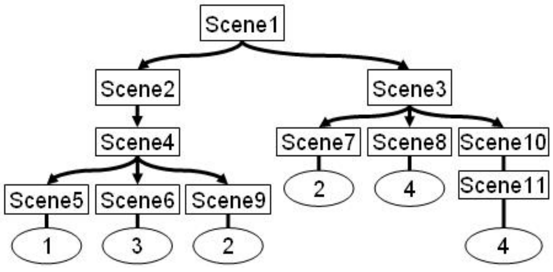


Fig. 2. A Simplified Decision Tree

Figure 2 shows a small example of a storyboard-tree. For simplicity, the labels (elliptic nodes) are reduced to the value of WAA. According to the form of the scenes in a storyboard the atomic attributes of the tree are placed in rectangles. The path through the storyboard-tree is defined by its directed edges. Only the connection to the label is non-directed, because it refers to the complete path.

B. Utilizing the decision tree for path estimation and completion

The utilization or application of this decision tree that contains knowledge gained by Data Mining of storyboard paths is as follows. If a submitted path is already represented in the decision tree (as a path from its root to a node that is succeeded by a label node, i.e. with an *assess* - fact), the prediction or estimation is very easily done through presenting the content of this label.

In the other case, i.e. if a student submits a curriculum plan that is not represented in the decision tree, the most similar sub-path in the decision tree will be identified.

Here, similarity refers to the number of same subjects in the common part of paths that are represented in the tree. In other words, those paths in the decision tree that have the longest leading (starting and its succeeding) part in common with the path representing the submitted curriculum plan will be identified.

The similarity s ("least common denominator" in [2])is in the range $0 \leq s \leq 1$. In the worst case, there no node in common with any path in the tree (similarity = 0) and in the best case, the submitted path is completely represented in the tree (similarity = 1).

Like in the tree construction procedure, this is performed by simultaneously traversing the path's subject sequence and the decision tree down from the root until

1) the path is finished or
2) there is a "next node" in the path that is different from all "next sub-tree roots".

In the first case, the related *assess* - fact at its leaf position provides the desired success estimation. In the latter case, the *assess* - fact of the current tree position is provides the desired information.

Of course, in such a case, one may be interested in suggestions to modify the submitted path in a way that the chances for success reach an optimum or become the highest in their value. Thus, for example, it is suggested to exchange the submitted remaining path by the most successful alternative remaining path with the best WAA value among the remaining paths in the decision tree whose common leading part is the longest.

For this purpose, we supplement the estimated success chances by the most successful rest - path starting at the last node of the tree traversing and provide it along with this optimal achievable success, if this optimum is better than the success estimation of the submitted path.

Additionally, the user is informed about the degree of similarity of his submitted path and the one found in the decision tree. We call this similarity *significance* of the success estimation and compute it as the number of nodes in sequence that are common in the submitted path and the decision tree, related to the entire length of the decision tree.

The application procedure for the decision tree in Prolog is sketched as follows:

```
evaluate([K|R],Result,Sign,RestPath,RecRestPath,AchResult) :-
  tree(1,Pos,Root),
  equivalent(K,Root),!,              /* there is a tree with this root */
  eval(Pos,[K|R],Result,RestPath,Depth,TargetPos),
  length(R,1,Length),
  Sign = Depth / Length,             /* calculation of significance */
```

[3] For simplification, the (simpler) German university mark system is used to explain this. Here, 1 is the very best mark, mark 2 means "good", mark 3 means "satisfactory" with mark 4 ("sufficient") one can still pass and mark 5 means failing.

```
   bestrest(TargetPosition,RRP,AR),      /* find optimal supplement */
  check(Targetpos,RRP,AR,RecRestPath,Achesult).
                                         /* check, if it is really better */

evaluate(L,marks(0,[]),0,L,RecommendedRestPath,Assess) :-
                                      /* no path with this start in tree */
  tree(1,Pos,_),
  bestrest(Pos,RecommendedRestPath,Assess),
                 /* provide complete (optimal) path as "supplement"*/
  no_better(Assess).

check(TP,_,marks(Avg,_),[ ],marks(0,[ ])) :-
   assess(TP,marks(Avg1,_)),
  Avg1 < Avg, !.   /* not better, recommend "empty supplement" */

check(_,RRP,AR,RRP,AR).     /* recommend computed supplement */

eval(Pos,[ ],marks(0,[ ]),[ ],Depth,Pos) :- !,
  length(Pos,0,Depth).                   /*empty path, nothing to do */

eval(Pos,[_|R],Result,R,Depth,Pos) :-
   assess(Pos,Result),!,            /* assessment available, provide it */
   length(Pos,0,Depth).         /* length of common node sequence */

eval(Pos,[_ ],marks(0,[ ]),[ ],Depth,Pos) :- !,
  length(Pos,0,Depth).                      /* 1 element (left), ready! */

eval(Pos,[_|[K|R]],Result,RestPath,Depth,TargetPos) :-
  tree(_,DownPos,E),
  one_below(Pos,Downpos),
  equiv(K,E),             /* next node is root of one of the sub-trees */
  eval(DownPos,[K|R],Result,RestPath,Depth,TargetPos).

no_better(marks(V,_)) :- assess(_,marks(W,_)), W < V, !, fail.

no_better(_).

/* best rest path behind and its estimated success estimation: */
bestrest(TargetPosition,RecommendedRestPath,AchievableResult) :-
  findall(AssessedPositions,
                       assess_below(TargetPosition,AssessedPositions),L),
  best_position(L,Best),
  assess(Best,AchievableResult),!,
  compose_path(TargetPosition,Best,RecommendedRestPath).

bestrest(_,[ ],marks(0, [ ])).
```

Based on this modification suggestion for the submitted path,which comes along with the achievable estimated success estimation when taking it, the user can make a decision on whether or not holding on to the submitted curriculum or modifying in accordance with an optimization result through considering the attached success chance.

IV. Summary and Outlook

Storyboarding is a way of managing didactic knowledge for organizing learning processes. The general idea and objective (and vision) of this research can be outlined as follows:

- *Let's make explicit what we talk about!*
  The idea is (semi-) formally represent didactics.
- *Let's apply such representations in (our university) practice!*
  This way, also non-experts in didactics will become able to process a model of didactics. In particular, for university teaching this seems to be very useful, because university teachers (professors and tutors), who are usually excellent experts in their subject, but do not necessarily have the didactic skills to teach their subject.
- *Let's explore conditions that can be (formally) checked!*
  This includes the verification of logical anomalies such as consistency conditions, but also has the potential to check topical teaching knowledge like invariants, didactic principles, and so on.
- *Let's check the result of applying certain didactics in a case study!*
  By validating applied didactics based on the degree of success, we will be able to identify successful paths through storyboards and distinguish them from the less successful ones.
- *Let's learn from the validation results!*
  Based on the results of validation, we will be able to refine didactic knowledge towards incremental improvement by its (re-) validation. The cycle of ongoing validation and refinements bags the chance to incrementally (evolutionary) improve the didactics of teaching.
- *Let's derive successful didactic patterns!*
  This is a vision: Deriving didactic patterns inductively from successful and failing examples. The idea is to discover some general patterns that
  - good examples (storyboard paths that usually end up with a high level of success) have in common and
  - do not appear in bad examples (storyboard paths that end up with failure or bad results).
- *Let's utilize these patterns!*
  Of course, such patterns are the right components, when designing new storyboards. Thus, some day we may be able to support didactics by a design tool that makes use of a pattern library.

Three storyboards (one at at a US-, a Japanese -, and a German University) are prototype examples so far. These examples indicate that the concept is very general and "many purpose".

Because of clarity and simplicity, everybody can become a storyboard author. No Software technological Knowledge is needed, no specialized (expensive) tool is needed.

By storyboarding, didactic design became explicit and subject to AI inference techniques, evaluation and quality assurance:

1) Structure tests for verification are developed [15] [5] as a method to discover logical anomalies in storyboards.
2) An inheritance concept has been developed as a means of logical (deductive) inference over this knowledge representation.
3) Based on a set of operations that ensures logical correctness of storyboards, we developed a web-based storyboard development environment for our storyboards.
4) An inheritance concept of node attributes within the hierarchy of nested graphs has been developed [17] and implemented.
5) As a first step towards knowledge mining over this knowledge representation, we developed a method to estimate success chances of intended storyboard paths by applying data mining methods to paths that have been traversed formerly and their related degree of success.

Additionally, this approach also suggests a supplement to a given curriculum that leads to an optimum with respect to the success chances.

The latter approach has been developed for practical use at a Japanese university (Tokyo Denki University), where students are requested in their first semester to compose an individual study plan, which meets all formal regulations as well as their individual needs, desires, talents, opportunities, and carrier goals.

Since this approach is just prototypically implemented in Prolog, a next step towards it's utilization is designing an interface between the SQL-database behind the storyboard development environment in [13] and the Prolog implementation of the approach of [2].

In fact, the above mentioned list of objectives and visions starts with items that are well done so far, but ends up with items that are hard to achieve and subject of much research left. In particular, the last two items are not touched at all so far, but they are our dream and ultimate goal.

Our upcoming work to further develop the storyboard concept is directed towards solving the following issues.

- A definition and representation of (formally to check) criteria that allows the specification of individual goal-driven storyboards.

  In fact, this is very different depending on cultures, countries, and universities. Therefore, we plan to do that prototypically for SIE at TDU.
- Storyboards have a high performance with respect to didactic issues of planning education processes. However, there is still no capability to manage these processes according to their resources (e.g. to concretely plan weekly timetables based on requests and available capacities like rooms, teachers, equipment and so on).

  Therefore, a desirable synergy effect is expected when incorporating the planning capabilities of the Dynamic Syllabus tool of the DLNRS into the storyboards.
- Including meta-knowledge is another focus to infer learning needs, learning desires, preferences and talents. This meta-knowledge is useful for maintaining the university's educational resources according to the needs through having some prediction about upcoming students' learning needs.

  This is desirable not only for a need-oriented and effective planning at universities, but also for suggesting the contents such as class schedules represented by storyboards according to the students' desires.
- As well, individual learning plans should not only be based on individual quantitative capability (like GPA) or the success of former students who went similar ways.

  Also individual properties, talents and preferences should be considered. For example, some students are more talented for analytical challenges, some are more successful for creative or composing tasks, and others may have an extraordinary talent to memorize a lot of factual knowledge.

  Consequently, at some point we need to include some sort of user profile to avoid lavishing the students with suggestions that don't match their individual preferences and talents.

REFERENCES

[1] Balaji, V., Arora, A., Jain, A., Goyal, G., Dubey, G., and Singh, S., SCORM Learning Sequence Modeling with Petri Nets in Cooperative Learning, *Learning Technology*, 7(1), Special Issue on SCORM 2004 Sequencing & Navigation, ISSN 1438-0625, 2005.

[2] Boeck, R., *Ein Data Mining Verfahren zur Pfadbewertung in Storyboards* (German), Diploma Thesis, University of Ilmenau, Faculty of Computer Science and Automation, Chair of Artificial Intelligence, 2007.

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

[4] Dohi, S., Sakurai, Y., Tsuruta, S., and Knauf, R., Managing academic education through dynamic storyboarding, *Proc. of the World Conference on e-Learning in Corporate, Government, Healthcare,and Higher Education 2006 (E-Learn 2006)*, Honolulu, Hawaii, USA, Chesapeake, VA: Association for the Advancement of Computing in Education (AACE), ISBN: 1-880094-60-6, pp. 1611-1619, 2006.

[5] Duesel, H., *Konzeption und Realisierung von Methoden der formalen Verifikation von Storyboards* (German), Diploma Thesis, University of Ilmenau, Faculty of Economic Sciences, 2007.

[6] Duesterhoeft, A. and Thalheim, B., SiteLang: Conceptual modeling of Internet sites, *Conceptual Modeling - ER 2001*, vol. 2224 of *Lecture Notes in Computer Science (LNCS)*, Berlin: Springer, pp. 179-192, 2001.

[7] Feyer, T., Kao, O., Schewe, K.-D., and Thalheim, B., Design of data-intensive web-based information services, Li, Q., Ozsuyoglu, Z.M., Wagner, R., Kambayashi, Y., and Zhang, Y. (eds.): *Proc. of the 1st Internat. Conference on Web Information Systems Engineering (WISE 2000)*, IEEE Computer Society, pp. 462-467, 2000.

[8] Feyer, T, Schewe, K.-D., and Thalheim, B., Conceptual modeling and development of information services, Ling, T. and Ram, S. (eds.): *Conceptual Modeling - ER'98*, vol. 1507 of *Lecture Notes in Computer Science (LNCS)*, Berlin: Springer, pp. 7-20, 1998.

[9] Feyer, T. and Thalheim, B., E/R based scenario modelling for rapid prototyping of web information services, Chen, P.-S. (ed.): *Advances in Conceptual Modeling*, vol. 1727 of *Lecture Notes in Computer Science (LNCS)*, Berlin: Springer, 1999, pp. 253-263.

[10] Jantke, K.P. and Knauf, R., Didactic design though storyboarding: Standard concepts for standard tools, *Proc. of the 4th Internat. Symposium on Information and Communication Technologies, Workshop on Dissemination of e-Learning Technologies and Applications*, Cape Town, South Africa, 2005, pp. 20-25, New York: ACM Press, ISBN 0-9544145-6-X, 2005.

[11] Kaschek, R., Matthews, C., Schewe, K.-D., and C. Wallace, C., Analyzing web information systems with the Abstraction Layer Model and SiteLang, *Proc. of the Australian Conference on Information Systems (ACIS 20039)*, 2003.

[12] Kaschek, R., Schewe, K.-D., Thalheim, B., and Zhang, L., Integrating context in conceptual modeling for web information systems, *Proc. of Web Information Services 2003 (WES 2003)*, 2003.

[13] Kasperski, S., *Entwicklung eine unabhaengigen Storyboardrepraesentation* (German), Studienarbeit, University of Ilmenau, Faculty of Computer Science and Automation, Chair of Artificial Intelligence, 2007.

[14] Lin, J., Ho, C., Sadiq, W., Orlowska, W.E., Using Workflow Technology to Manage Flexible e-Learning Services, *Educational Technology and Society*, 5(4), 2002.

[15] Sauerstein, G., *Formale Modellierung und daduktive Interpretation von Storyboards. Modellierung eines komplexen Prozessmodells* (German), Studienarbeit, University of Ilmenau, Faculty of Computer Science and Automation, Chair of Artificial Intelligence, 2006.

[16] Sykes, E.R. and Franek, F., A prototype for an intelligent tutoring system for students learning to program in Java, *Proc. of the IASTED International Conference on Computers and Advanced Technology in Education*, Rhodes, Greece, pp. 78-83, 2003.

[17] Xu, G., *Ein Vererbungskonzept fuer Storyboards* (German), Studienarbeit, University of Ilmenau, Faculty of Computer Science and Automation, Chair of Artificial Intelligence, 2006.