Capitalize and Share Observation and Analysis Knowledge to Assist Trainers in Professional Training with Simulation
Case of Training and Skills Maintain of Nuclear Power Plant Control Room Staff

Olivier Champalle[1,2], Karim Sehaba[1,3] and Alain Mille[1,2]

[1]*Universite de Lyon, CNRS, Lyon, France*
[2]*Universite de Lyon 1, LIRIS, UMR5205, F-69622, Villeurbanne, France*
[3]*Universite de Lyon 2, LIRIS, UMR5205, F-69676, Bron, France*

Abstract: The observation and analysis of the activity of learners in computerized environments training is a major issue, particularly in the context of professional training on nuclear power plant full-scope simulator. In such a context, the role of the trainers is critic and require constant alertness throughout the simulation especially for the young trainers. The objective of our work is to propose an approach to facilitate the observation and analysis of the trainees' activities. This approach is based on interaction traces. It consists in representing the operators' actions and the simulation data in the form of modelled traces. These modelled traces are transformed in order to extract higher informations levels on the behaviour of trainees. Trainers can visualize the different levels to analyse the reasons, of successes or failure of trainees. This approach has been implemented in a prototype, called D3KODE, allowing the representation, processing and visualization of traces. D3KODE was evaluate according to a comparative protocol conducted with a team of trainers from EDF Group.

1 INTRODUCTION

This work addresses the general question of observation and analysis of learners' activities in a situation of professional training. More specifically, we addresse the case of full scale simulators (FSS) designed to maintain and enhance the knowledge and skills of Nuclear Power Plant (NPP) control room staff (Champalle et al., 2011) (Blanc et al., 2010). In this context, the observation, analysis and debriefing of individual and collective interactions of trainees is a dense activity that require attention and constant alertness of the trainers throughout the simulation.

Indeed, during each simulation session, the instructor runs the simulation scenario, observes behaviour of trainees, drives the simulator based on the actions performed by trainees and fills an observation balance sheet to prepare and conduct the debriefing. Observation balance sheet contains a set of expected operations that trainees should be able to satisfy (see for example the Annexe G of (Agency, 2004)). These operations correspond to the knowledge and skills that are necessary to insure safety of the plant.

To assist trainers in their tasks, the NPP FSS pos-

sesses several tools to follow, record and replay the parameters of the simulator and the actions of the trainees. These tools have however their limitations:

- The data stored in the logs are very low level. In such a context, understanding and following the activity require a strong expertise that all trainers don't have;

- The amount of data collected during a simulation is so big that it is very difficult to analyse them manually and extract high level information reflecting the behaviour of trainees;

- The synchronization of these different types of data (video, sound, logs,...), is difficult and expensive. Indeed, the data are stored in different places and files and do not share the same time line.

With the aim of going beyond these difficulties our proposals are the following:

- assist trainers in observation and analysis of activity by:
  - providing a visual synthesis of the activity with the expected observations of the trainees (realized or not realized);

– allowing exploration of the abstraction levels to facilitate the analysis of the activity;

– allowing trainers to add their own observations corresponding to their expertise to save time during the phases of observation and analysis.

● strengthen the debriefing by:

– providing trainees a visual synthesis of their activities in a self-reflexive way;

– providing trainers factual information to exchange with the trainees.

● favour sharing of observation and analysis knowledge between trainers by:

– allowing trainers to exchange on their practices;

– allowing new trainers to reuse the knowledge of the confirmed trainers.

To improve the actual tools for trainers and concretize our proposals we base our reflections on a better exploitation of simulation data.

This article is organized as follow: section 2 presents a state of the art on related works. Section 3 presents our approach, models and tools for observation and analysis activities. Section 4 present the evaluation of our approach and the results. The last section is devoted to a conclusion and perspectives.

2 RELATED WORKS

Exploitation of the trace of trainees' activities is a spread practice in the computerized dedicated environments for training. With the term Digital Traces we mean all numerical data produced by an activity or set of activities. These activities result from human system interactions and/or between a system on another system.

In order to find the most relevant approaches for our proposals, we have conducted a state of the art in two steps. At first we have compared approaches close to our context and research proposals. In a second step, we focused our study in the field of knowledge engineering to determine the approaches best suited to capitalize knowledge from observation and analysis of the activity, whatever it is.

Approaches on Similar Research Context

In the domain of Nuclear Power Plant Full-Scope Simulator, the SEPIA system (Dunand et al., 1989), a computer training system by artificial intelligence, was designed for operators training of EDF Group

on pre-defined scenarios. SEPIA is based on an expert system constructed from the knowledge of driving procedures and expert trainers. Once the simulation over, SEPIA gives a feedback by analysing and correcting the operators' actions. During the debriefing, SEPIA allows the operators to obtain explanations concerning any registered parameters.

In Aircraft simulation (IIPDSS), (Bass, 1998) introduces an intelligent trainer pilot decision support system that use trace of trainee-pilot' activities to help trainer during the simulation, debriefing, and performance evaluation. During the practice, the system displays in real time to the trainer a list of message to help him understand and guide the trainee. For the evaluation, the system collects and displays the evaluation criteria applicable to a particular simulated mission and instances when the trainee failed to meet the criteria. At the end of the simulation, the system provides a complete trace of the divergent actions of the trainee, associated with the corresponding data of the aircraft simulation environment. The system provides also a summary of the skills with which the trainee has difficulty.

The PPTS project (Pedagogical Platoon Training System) (Joab et al., 2002) assists trainers to observe and analyse the tactical behaviour of a platoon network of four LECLERC tank simulators. The PPTS integrates an ITS which reproduces the expertise of trainers in order to exploit and analyse the numerical traces of the simulation to highlight the three skills levels expected from each crews: technical, tactical and strategic. Each level being constructed on the basis of lower level. At the end of the simulation, the PPTS generates a summary document and comment on the skills of the crews to assist trainers in the debriefing phase.

Works cited above use the digital traces of activity to diagnose and analyse behaviour of trainees but miss important properties. Indeed, the tools proposed in these approaches are based on closed systems. Their implementation is generally heavy (ITS, expert system) and requires a long and close collaboration with experts. Another limitation lies in the "static" knowledge used by these systems. Indeed, once these systems are built, it is not possible for the trainers to create and share their own observation and analysis knowledge. It is also not possible to define the levels of expected skills.

To meet this specific need of creation and sharing of observation and analysis knowledge based on digital traces, we studied other approaches from knowledge engineering.

Knowledge Engineering

(Dyke, 2009) proposes the analysis tool Tatiana (Trace Analysis Tool for Interaction Analysts) which implements the concept of *replayable* in order to assist the analysis of heterogeneous data (videos, logs, ..). A replayable is a generic analytic artefact that models and capitalizes an analysis methodology built from user trace. A replayable can be visualised, replayed, enriched, transformed to produce a new replayable and synchronised with other like artefacts.

The system ABSTRACT (Analysis of Behaviour and Situation for menTal Representation Assessment and Cognitive acTivitiy modeling)(Georgeon et al., 2011) is used to analyse human activity from successive transformations of low level trace. Transformations are based on SPARQL rules and can be reused in different contexts. Abstract provided the possibility to visualize traces and their transformations.

The TBS-IM (Trace Based System Indicators Moodle) (Djouad et al., 2010) allows the creation of individual and collective indicators for educational activities on the learning platform Moodle. TBS-IM supports the user in the elaboration of indicators masking successive abstraction levels necessary for their construction. These indicators calculations can be capitalized for the purpose of reuse.

These approaches are similar to our research objective. Their advantage lies in the common use of the concept of *Modelled Trace* (see Section 3) and the principle of reuse of transformation for abstracting a low level trace in order to highlight higher levels of knowledge. There are however differences and limits. In (Dyke, 2009), there is nothing to assist the trainers in the creation of the transformations rules, TBS-IM (Djouad et al., 2010) can not view the activity on different levels of abstraction and ABSTRACT (Georgeon et al., 2011) does not allow transformation with several rules.

If these approaches do not fit all the searched properties, the approach based on modelled trace and transformation rule, enhanced in ABSTRACT and TBS-IM, seems to be more flexible, facilitating the analysis and knowledge sharing.

3 CONTRIBUTIONS

Our research is based on the concept of modelled trace developed by the SILEX[1] team (Settouti et al., 2009). A Modelled Trace, noted *M-Trace*, is a set of observed elements associated to the trace model it-

self. We call observed element, noted *obsel* any structured information generated from the observation of an activity (Georgeon et al., 2011). In our research context, activity observed is a training simulation on a NPP FSS of EDF. The obsels collected are the result of users interactions (trainees and trainers) with the FSS, and the simulation of the NPP process itself. Formally, each observed element has a type, a label and a time stamp in the M-Trace. According to his type an obsel can have a set of attributes/values, which characterizes it. An obsel can potentially be in relation with other obsel of the same M-Trace through *relation type* defined in the Trace Model. The Trace Model defines the types of observed elements (i.e the attributes that characterize them) and the types of relationships they can have between them. A Modelled Trace is then a structure of data (obsels and relations) explicitly associated with its trace model. Modelled Traces are managed with a Trace-Based Management System (TBMS) (Settouti et al., 2009). The TBMS is responsible for managing the storage of traces (rights management, database...) and their transformations. A Transformation process performs transformations on M-Traces like applying filters, rewriting and aggregating elements, etc. so as to interpret and abstract M-Trace.

Visual Synthesis of the Activity

For reasons we have explained above, the data collected by the simulators are difficult to analyse. It's why we distinguish three levels of m-traces:

- Primary M-Trace whose obsels arise from data collected by the sources of the simulator;

- M-Trace of Pedagogical Objectives, which represents the first level of expected obsels that trainees have to realize and trainers to check;

- M-Trace of Pedagogical Objectives Family. This higher M-Trace shows obsels that describe "pedagogical objectives family" (realized or not) as expressions of the expected trainees' capacities.

These different levels of M-traces are obtained by applying rule-based transformations. These rules makes explicit the observation knowledge of expert. As shown in figure 1, each obsel belonging to a trace of level *n* is in relation with its origin obsel(s) from the trace of level $n-1$. The obsels of the Primary M-Trace are in relation with the data collected by the simulator. Such structure allows trainers to explore, analyse the trainees' activities with a top-down investigation to understand the reasons, be they individual or collective, of failure. Consequently, trainers would also be able to better prepare and conduct the
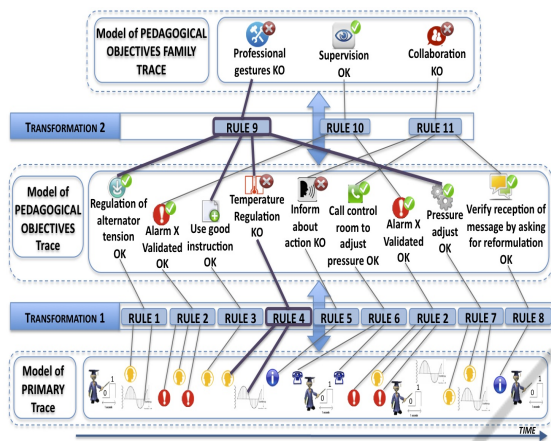
---

[1] http://liris.cnrs.fr/silex

Figure 1: Principle of analysis by transformation and visualization of trace.

session's debriefing with trainees. On the other hand it would be possible to help new trainers to improve their skills by helping them in observing Trainees. For example, if the trainers want to understand the reasons why the obsel "Professional Gestures" of the M-Trace of Pedagogical Objectives Family is KO[2], they have just to navigate through their different origin obsels of the M-Trace of Pedagogical Objectives. According to the Rule 9, the obsel "Professional Gestures" is OK only if all of these four origin obsels are OK.

Generic Trace Model

Whatever the level of M-Trace, its model and the simulator used, we believe that a simulation M-Trace must "pick-up" its own identity to be locatable and usable over time. This "identity card" of the M-Trace would be particularly useful for large-scale statistical research and/or analysis on a set of M-Trace corpus, and particularly to feed experience feedback, described in the Trace-Based framework presented in (Champalle et al., 2011). Through experience feedback, trainers try to understand good or bad practices on several simulations in order to improve contents of the future training courses.

So, as described in the class diagram of Figure 2, all the M-Traces of our model have an ID, a beginning and end date, the level of M-Trace (Primary, Pedagogical Objectives,...), type of simulator, type of training (Initial or Retraining), category (Summative, Formative), with the scenario of simulation and the training program (Op Reactor, Op Turbine,...). In our context, the obsels are the direct result of users' interactions (trainees and trainers) with the FSS, and the message of the NPP simulation process (Primary M-Trace) or result from execution of transformations

(M-Trace of Pedagogical Objectives and M-Trace of Family Pedagogical Objectives). Therefore each obsel type has common attributes: an ID, a begin and a end date, a label, the ID of the Generative subject (Person, Group, Simulator), a SubjectNature (evaluated or not), the RoleSubject (Op Reactor, Op Turbine, Supervisor, etc.) and a realization attribute (OK or KO). This model can be specify for each M-Trace model and obsel as needed.
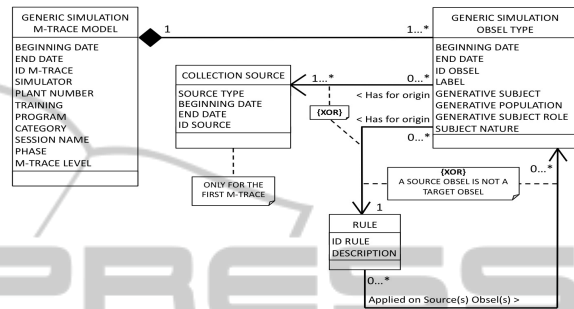

Figure 2: Generic Trace Model.

Transformation Model

Transformation allows generating a target trace of level n from a source trace of level n-1. Each Transformation is composed of a set of rules which have a part *Condition* and a part *Construction* (Figure 3). The condition part expresses constraints on the elements (obsel type, relationship type, values of attribute, etc.) of the M-Trace(s) source(s).

The construction part allows defining the obsel and the relationships of the new target M-Trace if all the constraints of the condition part are satisfied. For each part of a rule (condition and construction), it's possible to use specific operators as arithmetic, boolean and/or comparison in order to express constraints on the attribute values of sources obsels (condition) or make calculations to initialize the values of the attributes of the obsel target (construction).

D3KODE

D3KODE as "Define, Discover, and Disseminate Knowledge from Observation to Develop Expertise" is a Web application, which stores and transforms traces according to the organization and the models presented in previous sections. D3KODE also allows the user to interactively view the various trace levels. So the trainers can explore the different abstraction levels in purposes of investigation and/or education to target gaps and difficulties of each trainee.

The storage of M-Trace and transformation models and rules is based on the kTBS (kernel for Trace
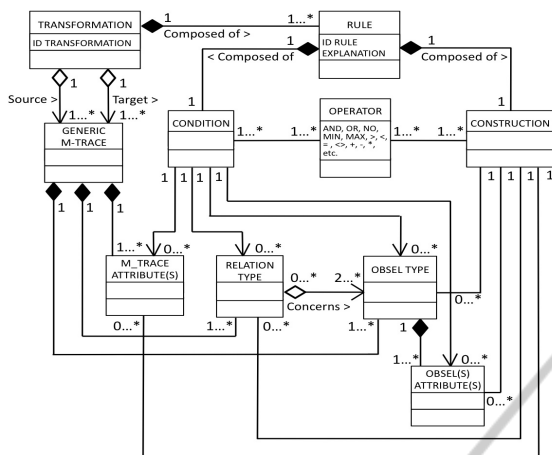
---

[2]The operator did not validate this objective

Figure 3: Transformation model.

Base System). The kTBS[3] is a Trace-Based Management System architecture (Settouti et al., 2009) developed by the SILEX team. Data of the kTBS are encoded in RDF[4] (Resource Description Framework). The transformation rules, are written in SPARQL1.1[5]. D3KODE is multi-user and multi-language.

4 EVALUATION AND RESULTS

In order to evaluate the properties of our approach we have defined an evaluation protocol based on a qualitative comparative method (Per and Martin, 2009). It consists in comparing the observation, analysis and debriefing of the trainees' activities with and without D3KODE. This evaluation was driven on a summative scenario on a FSS of an NPP of EDF Group. It mobilized 8 people: 2 confirmed trainers to drive the simulation and observe trainees, 4 trainees to drive the simulator (2 for each simulation) and 2 experts to observe the simulations and to assess the support provided by D3KODE. The trace models and transformation rules of the scenario were modelled by an expert trainer who did not attend the evaluation. In this way, we ensure the correct evaluation of the sharing of knowledge between trainers.

Simulation with D3KODE

The first simulation lasted 1 hour and 14mn and generated 3591 obsels which were collected and injected into D3KODE. The trainers have observed 2 No Realizations (NR) tracked in the simulation logs and viewable by D3KODE.

---

[3]http://liris.cnrs.fr/sbt-dev/ktbs/

[4]http://www.w3.org/RDF/

[5]http://www.w3.org/TR/sparql11-query/

During the analysis, the obsels displayed by D3KODE were faithful with the simulation. Nevertheless, their visualization, did not bring more information to trainers. They have however considered valuable and useful the quick access to information such as begin and end date, obsel's label, rule's description, etc. This possibility has facilitated the analysis and exchanges between trainers. The presence, in D3KODE, of the 2 NR was a significant contribution for the trainers. No Realization are indeed what is traditionally harder to be confirmed.

During the debriefing, D3KODE was used by trainers as a visual aid to re-trace the chronology of observation and exchange with the trainees. During this phase D3KODE has also helped to highlight a NR which was "forgotten" by the trainers.

Traditional Simulation

The second simulation lasted 1 hour and 13mn. According with the evaluation protocol, D3KODE wasn't used for analysis and debriefing. During this simulation, The trainer have observed 3 No Realizations (NR) tracked in the simulation logs. The analysis and debriefing phase of the operators took place in much the same way that the simulation 1.

Global Analysis of the Results

The analysis of the results allowed us to collect several remarks on the contribution of D3KODE.

First of all, the trainers, showed themselves particularly interested in the visual synthesis of the activity on several levels. It would allow them, in the phase of analysis, to compare and verify the observations noted during the session, in order to be sure that nothing has been forgotten and so to raise all ambiguities. In this way our proposal concerning the revealing of observable KO was very relevant.

This evaluation also confirmed that the approach of D3KODE, based on rules and transformations in order to share knowledge of observation and of analysis, was understood and validated by all trainers. They have also validated the interest of our approach for the creation of additional specific observation (outside of the balance sheet of the evaluation) to attend their analysis and decision on the conduct of trainees. D3KODE would be used for post-analysis and calculations that trainers can not do in real time and/or analysis.

For the phase of debriefing, the essential contribution of D3KODE lies in the factual data presented through the visual synthesis of the activity. This functionality is indeed considered by the trainers as a good mean to encourage the reflexive self-analysis of the

trainees and highlight the axes of improvement.

Globally, the evaluation of D3KODE in real condition demonstrated that the trainers considered the contribution of D3KODE as relevant to enrich their activity. In particular D3KODE could strengthen the current tools of the follow-up and the analysis of the trainees' activities mainly a posteriori.

To wider scale, D3KODE would be even of a particular interest to analyse traces of simulation in quantity to highlight the recurring errors of the trainees and integrate them to the EXperience Feedback and build new more adapted trainings.

It should be added that if the trainers perceive D3KODE as potentially beneficial, they have pointed out that the use of D3KODE would introduce a change in their practices.

5 CONCLUSIONS

This article addresses the problem of observing and analysing trainees' behaviour on Nuclear Power Plant Full-Scope Simulator. This work, conducted in partnership with the UFPI of EDF Group, is applied in the context of training and maintaining the knowledge and skills of NPP control room staff. The objective of our work is to propose models and tools to help trainers capitalize and share their observation and analysis knowledge in order to improve observation, analysis and debriefing of trainees' activities during formative/summative assessment.

The approach we proposed is to transform the raw traces, based on data collected from the simulator, in order to extract high level information on the activities of trainees. For this we have proposed a dedicated trace model and transformation. In order to guarantee the exploration between various levels of M-Trace, each obsel possesses a link on its origin.

We have also developed a prototype, called, D3KODE which favour share of trainers' observation knowledge, and which stores, process and visualize the traces. This prototype implements the various models we have created. So as to validate our approach, we have conducted an evaluation based on a comparative method. This experiment was conducted with a team of trainers from UFPI of EDF Group in a real context. The result of the evaluation demonstrated that our approach was favourably welcomed by the trainers and could be really relevant to enrich their activity.

Our future work will aim to address the second objective of the project: exploitation of traces for the experience feedback to refine the needs and optimize training programs for years to come.

REFERENCES

Agency, I. A. E. (2004). Use of control room simulators for training of nuclear power plant personnel. Report 36017984, International Atomic Energy Agency, Vienna (Austria).

Bass, E. J. (1998). Architecture for an intelligent instructor pilot decision support system. In *IEEE International Conference on Systems, Man, and Cybernetics*, volume 1, pages 891 –896, San Diego, CA.

Blanc, K. L. L., Boring, R. L., and Gertman, D. I. (2010). Review of methods related to assessing human performance in nuclear power plant control room simulations. In *Conference: American Nuclear Society International Topical Meeting on Nuclear Plant Instrumentation, Control and,Las Vegas, NV,11/07/2010,11/11/2010*, pages 411 – 422. Idaho National Laboratory (INL).

Champalle, O., Sehaba, K., Mille, A., and Prie, Y. (2011). A framework for observation and analysis of learners' behavior in a full-scope simulator of a nuclear power plant - approach based on modelled traces. In *ICALT 2011 - The 11th IEEE International Conference on Advanced Learning Technologies*, pages 30–31. Communication affichée.

Djouad, T., Benmohammed, M., and Mille, A. (2010). Indicators computation from modeled traces in the context of computer human learning environment. In *International Symposium on Modelling and Implementation of Complex systems*, pages 1–7.

Dunand, R., Dravers, F., and Guillermard, J. (1989). Sepia : un système d'enseignement par intelligence artificielle pour les opérateurs d'edf. *Revue générale nucléaire*, 3:254 – 257.

Dyke, G. (2009). *Un modèle pour la gestion et la capitalisation d'analyses de traces d'activités en interaction collaborative*. PhD thesis, Ecole Nationale Supérieure des Mines de Saint-Etienne, St Etienne.

Georgeon, O., Mille, A., Bellet, T., Mathern, B., and Ritter, F. (2011). Supporting activity modelling from activity traces. *Expert Systems*.

Joab, M., Auzende, O., Futtersack, M., Bonnet, B., and Leydour, P. L. (2002). Computer aided evaluation of trainee skills on a simulator network. *ITS 2002 : Proceedings of the 6th International Conference on Intelligent Tutoring Systems*, pages 521–530.

Per, R. and Martin, H. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering*, 14:131–164.

Settouti, L. S., Prie, Y., Marty, J.-C., and Mille, A. (2009). A trace-based system for technology-enhanced learning systems personalisation. In *The 9th IEEE International Conference on Advanced Learning Technologies*.