

# Tutorial Planning: Adapting Course Generation to Today's Needs

Carsten ULLRICH

*German Research Center for Artificial Intelligence (DFKI GmbH),  
Stuhlsatzenhausweg 3, 66123 Saarbrücken, Germany*

Most of today's course generation (the automatic assembly of sequences of learning objects, e.g., [2]) does not allow the declarative representation of pedagogical knowledge as possible with, e.g., [3]. In my work, I develop a framework that adapts these approaches to today's needs. Within this framework several of today's problems are solved, such as the integration of distributed content and e-learning services within a course, dynamic adaptivity of a generated course, new forms of interaction, and offering course generation as a service.

A declarative and generic representation of pedagogical knowledge as advocated in [3] distinguishes between the tutorial tasks to be achieved (e.g., `teachConcept` or `provideAdequateExercise`) and different methods of achieving these tasks (e.g., using a problem-based educational strategy, and depending on the learner's characteristics). In my approach the pedagogical knowledge is executed by a hierarchical task network planner [1]. The result of the planning is a sequence of learning objects (a content structure). As tasks represent a vast range of pedagogical goals, the size of the sequence ranges from a single element to a complete curriculum.

The problem of how to provide dynamic adaptivity of a generated course can serve to exemplify my approach. Course generation faces the dilemma that early course generation cannot take into account how capabilities of the learner actually change. Still, generating a course as early as possible supports orientation and self-organisation of the learning process. A different solution as plan repair is lazy task execution. In this approach, planning may stop at the level of specially marked tasks (lazy tasks). These tasks are inserted in the content structure just like any other learning objects. When the learner first visits a page that contains a lazy task, the task in the content structure is passed to the tutorial planner. The resulting learning objects replace the task in the course structure for good (hence, when the page is revisited, the elements do not change, which avoids confusion of the learner). This means a course is partly static, partly dynamic. Lazy task execution offers new possibilities for authors, too. An author can define a course structure, where parts of her course are predefined, and others dynamically computed taking the learner model into account. In this way, an author can profit from the best of both worlds: she can compose parts of the course by hand and at the same time profit from the adaptive features of the tutorial planner.

## References

- [1] K. Erol, J. Hendler, and D. S. Nau. HTN planning: Complexity and expressivity. In *Proc. of the 12th National Conference on Artificial Intelligence (AAAI-94)*, volume 2, pages 1123–1128, Seattle, Washington, USA, 1994. AAAI Press/MIT Press.
- [2] P. Karampiperis and D. Sampson. Adaptive instructional planning using ontologies. In *Proc. of the 4th IEEE International Conference on Advanced Learning Technologies*, pages 126–130, 2004.
- [3] K. van Marcke. Instructional expertise. In C. Frasson, G. Gauthier, and G. McCalla, editors, *Proc. of the 2nd International Conference on Intelligent Tutoring Systems*, number 608 in Lecture Notes in Computer Science, pages 234–243. Springer, 1992.