

# An Intelligent Tutoring Architecture for Simulation-based Training

Dave Gomboc, Mark Core, H. Chad Lane, Ashish Karnavat, Milton Rosenberg

Institute for Creative Technologies, University of Southern California  
Marina del Rey, CA, U.S.A.  
{gomboc, core, lane, karnavat, rosenberg}@ict.usc.edu

## Abstract

Truly generic and reusable intelligent tutoring software architectures have remained elusive. As part of our effort to develop tutoring systems for simulations of ill-defined domains, a software framework has emerged with minimal dependencies on any domain-specific details, except for the data used to instantiate the framework for a particular application. Here, we describe this framework, its functionality, underlying representations, configurability, and use of natural language generation.

## Introduction

The Intelligent Guided Experiential Learning (IGEL) project focuses on supporting students' learning while practicing skills and problem solving in ill-defined domains using simulations. A primary goal of the IGEL framework is to minimize dependency upon domain-specific details. To accomplish this, we use *learning objectives (LOs)* as a means of abstraction. Examples in this paper are from ELECT BiLAT (Hill et al., 2006), which includes an instantiation of the described architecture for bilateral negotiation (Lane et al., 2007).

The lessons that IGEL attempts to impart to users are encapsulated by a hierarchical tree whose leaf nodes identify unique LOs. Direct siblings in this tree are more related than more distant nodes. So, a top-level LO in BiLAT represents socialization, which has a child "small-talk", which is then subdivided further.

Simulation runs are represented as a sequence of actions and the resulting changes in the state of the world. In BiLAT, states correspond to mental representations of the meeting partner and deals being made, while actions correspond to meeting actions (e.g. saying something, making an offer in a negotiation).

*Annotations* are the central data structure used by IGEL in deciding what guidance to provide. Annotations link simulation actions made by the student to LOs via positive and negative links that denote the student's understanding (or lack thereof) of the LO. This understanding is assessed by the expert model based on the action performed and the context in which it was performed. In BiLAT, the student action of small talk concerning history should be annotated as positive evidence that the student

understood the LO discussed previously on socializing by discussing history.

## IGEL Architecture

IGEL is comprised of a set of software components, each of which runs in its own process. As such, we have directly benefited from the transition from single-core to quad-core processors in desktop computers during our system's development. Component communication is handled via marshalling messaging objects to XML, transmitting them over TCP/IP, and reconstituting them upon receipt.

IGEL defines representation objects as a canonical way to represent such data within the database, as a Java object, and as XML. Everything from actions and states to tutoring tactics is represented via these objects, which may be stored/loaded from file and network streams, and databases. We have implemented generic code that perform transformations between these representations automatically for every applicable Java object.

**Messaging Bridge.** It is necessary to convert between the simulation's own and IGEL's (abstracted) representation of scenario data, states, actions, and software control messages. The bridge processes all communication traffic between the outside simulation and IGEL, making the appropriate adjustments to messages on the fly as it transfers messages from one network to the other.

**Expert Model.** The expert model performs domain-dependent reasoning on behalf of the tutoring system. Its two major roles are recommending actions to perform given a state, and assessing student actions at a state through the process of generating annotations for them. Our expert model for BiLAT is separate from the simulator. Integrating these could help ensure the fidelity of the expert model to the simulation and avoid communication overhead, but requires tight collaboration between the simulation and ITS builders. Organizational and geographical dispersity, differences in development funding and timing, and incentive disparity between developers may make such collaboration infeasible.

## Domain-independent Tutoring Framework

We distinguish between *coaching* (advice given during the problem solving itself) and *reflective tutoring* (an interactive after-action review (AAR) of the problem-solving process after its completion). The types of student interactions appropriate in each case are different. While

actively engaged with the simulation-based practice environment, the student typically lacks the additional cognitive resources required to answer questions or have the time to comprehend lengthy guidance. In contrast, the learner's complete attention can be focused upon the AAR: it is appropriate for a reflective tutor to engage the student in extended dialogue regarding their experience. IGEL includes distinct software components that implement coaching and reflective tutoring – these have different designs supporting these different requirements.

### Coaching

Because it is a design goal that coaching commentary be readily authorable by non-programmers, NLG templates for coaching commentary are specified within spreadsheets. Two separately controllable, domain-independent aspects of the coach determine when it will query the database to search for relevant commentary and how it selects between multiple commentary options.

**Timing selection.** In order to give hints before student actions and feedback after student actions, the coach needs to be aware of the progress of the simulation's main loop. IGEL supports multiple versions of the model-scaffold-fade algorithm inspired by cognitive apprenticeship (Collins et al., 1989). Note that timing strategies are always encoded as “attempt to give advice” because we cannot insist that all potentially relevant remarks be created by an author.

**Content selection.** At any advice-giving moment, IGEL can choose to either comment about a learning objective, or directly describe why a particular action was, was not, or would be beneficial.

**Wizard of Oz.** In order to allow us to observe how an expert coach would advise students, we also have a version of our coach that may be operated by a subject matter expert. The human operator of the Wizard of Oz GUI has full access to the simulation history as perceived by IGEL, the library of possible actions the user can perform, and what the automated coach would have said to the student were it being used.

**Simulation Event Processor and Logger.** IGEL's main loop during the external simulation's execution lies in the simulation event processor. The processor coordinates the activities of the logger, the expert model, and the coach with the simulation, ensuring that each of these components is signalled to provide its input at the appropriate time.

### Reflective Tutoring

At the conclusion of a distinct problem-solving episode in the simulation, our reflective tutor conducts an AAR with the student reviewing their performance. Our current system's domain-independent tutoring capabilities are implemented in a simple reactive planner written in Java. All domain-dependent information affecting the tutor, e.g. tutoring tactics, is entirely specified in XML data files.

These files are sufficiently simple that non-programmers may be trained to review and modify them directly, after which the tutor may be reinvoked without recompilation.

**Agenda Selection.** An AAR is driven by an agenda that is created from the assessments made during practice. IGEL can prioritize its discussion topics by different metrics, including error magnitude and action chronology. Annotations that share learning objectives are clustered so that closely related issues relating to different points of time during the simulation are juxtaposed in the AAR.

**NL Generator.** Authorability requirements mandate the use of a mechanism by which non-programmers may create coaching and tutoring feedback. We have used the pipe-and-filter architectural style to implement a template-based approach, whereby hard-coded text is mixed with XML tags that are replaced by appropriate text on-the-fly. Typically, domain-dependent substitutions precede domain-independent substitutions in our pipeline.

**Student Model.** IGEL does not have the benefit of a well-defined domain with well-understood and agreed-upon methods for successfully completing tasks. Our present, rudimentary student model tracks each student action and its accordance or discordance with the relevant LOs, as determined dynamically by the expert model.

### Summary

The BiLAT tutoring system, which employs the IGEL framework, is currently being evaluated by the U.S. Army. We are currently preparing for other simulations under development here at the Institute for Creative Technologies (ICT) to employ IGEL in combination with appropriate domain knowledge.

### Acknowledgements

This research was sponsored by the U.S. Army Research, Development, and Engineering Command (RDECOM). Statements and opinions expressed herein do not necessarily reflect positions or policies of the U.S. Government; no endorsement should be inferred.

### References

- Collins, A., Brown, J.S., and Newman, S.E. (1989). Cognitive apprenticeship: Teaching the crafts of reading, writing, and mathematics. In L.B. Resnick (Ed.), *Knowing, learning, and instruction: Essays in honor of Robert Glaser*, 453-493.
- Hill, R.W., Belanich, J., Lane, H.C., Core, M.G., Dixon, M., Forbell, E., Kim, J., and Hart, J., 2006. Pedagogically structured game-based training: development of the ELECT BiLAT simulation. In *Proceedings of the 25<sup>th</sup> Army Science Conference*, Orlando, FL, U.S.A.
- Lane, H. C., Core, M. G., Gomboc, D., Karnavat, A., Rosenberg, M., 2007. Intelligent Tutoring for Interpersonal and Intercultural Skills. In *Proceedings of the Interservice/Industry Training, Simulation, and Educational Conference*. Orlando, FL, U.S.A.