

Simulation2Instruction: Using Simulation in All Phases of Instruction

Robert Wray, Ph.D.
Soar Technology, Inc.
Ann Arbor, MI 48015
wray@soartech.com

Allen Munro, Ph.D.
University of Southern California
Los Angeles, CA
munro@usc.edu

ABSTRACT

Today, significant use of simulation technologies is seen for all services within the US military, at almost all echelons, and for most kinds of military roles and missions. Simulation's primary role today is to provide practice experiences that approach real world fidelity. Simulation enables more frequent and sustained practice than the real world often allows and contributes to the transfer of training to operational environments.

Using simulation for realistic practice should remain central to simulation's role in training. However, simulation can be applied throughout all phases of the training cycle. Historically, the primary bottleneck for additional uses has been the time and resource cost of content development. We show how existing simulation technologies can be readily exploited and extended to enable the use of simulated experience across a much broader span of the cycle of instruction. We outline interworking technologies (Simulation2Instruction) that, in combination, remove the primary bottleneck of content development by automatically capturing all student (and instructor) activity in a simulation. These simulation recordings then become "content" for demonstrations, assessments, and stand-alone instruction. Content development activity shifts to search rather than construction. Search is both faster and simpler, which extends the audience of content developers because less technological skill is required.

We illustrate Simulation2Instruction technologies with examples of demonstrations, assessments, and stand-alone instruction (used for remediation) in a simulation environment in use at a US Navy schoolhouse. We also offer recommendations for modest requirements for existing and future simulation platforms that will result in enabling similar uses and benefits to others wishing to exploit simulation for all phases of instruction.

ABOUT THE AUTHORS

Robert E. Wray is a Senior Scientist at Soar Technology. He received a Ph.D. in computer science and engineering from the University of Michigan. At Soar Technology, he leads or has led R&D projects for the Air Force and Navy research offices, and the Defense Advanced Research Projects Agency. Dr. Wray's research encompasses many areas of artificial intelligence research including agent-based systems and agent architectures, machine learning, cognitive science, intelligent tutoring systems, and knowledge representation and ontology.

Allen Munro is a professor of research at the University of Southern California's Rossier School of Education and is co-director of USC's Center for Cognitive Technology (CCT). He conducts research on learning in the context of simulations, games, and interactive graphical visualizations. Under the sponsorship of the Office of Naval Research (ONR), he and his colleagues created RIDES, VIVIDS, and iRides—software systems for authoring and delivering simulation-centered training. He is interested in automated assessment and adaptation in the context of games and simulations.

Simulation2Instruction: Using Simulation in All Phases of Instruction

Robert Wray, Ph.D.
Soar Technology, Inc.
Ann Arbor, MI 48015
wray@soartech.com

Allen Munro, Ph.D.
University of Southern California
Los Angeles, CA
munro@usc.edu

INTRODUCTION

Simulation-based training is becoming a cornerstone of military training. Simulation technologies are deployed throughout all the branches of the US military, at almost all echelons, and for most kinds of military roles and missions. Well-designed simulation enables more frequent and sustained practice, due to much lower costs of training than the real world. High-fidelity experiences also contribute to the transfer of training to operational environments (Micheli, 1972).

Investment in simulation technology has historically been directed toward producing realistic (high fidelity) environments comparable to the fidelity of live practice environments. However, effective instruction requires more than just providing a realistic practice environment. Merrill (2002) suggests that while the problem/task focus enabled by simulation is important for learning, practice (“application”) is only one of four phases that are needed to structure task-centered learning effectively. Problem-centered activation (relating new concepts and skills to prior experiences), demonstration, and integration (enabling students to demonstrate learning) are also important.

Our long-term goal is to develop general technology that enables the use of simulation technology for these additional elements of instruction. We refer to these technologies collectively as *Simulation2Instruction* or S2I. S2I makes it easier for schoolhouse instructors, training content developers, and students to exploit simulation systems for uses other than task practice. Examples of user activities enabled by S2I include:

- Direct reference to prior experiences (activation) via simulation recordings;
- Controlled presentation of concepts and demonstrations situated in a problem-centered context (demonstration);
- Assessment of student knowledge and skills within a performance context (practice and integration);
- Reflection on ones’ own work and work of others, including the ability to present one’s work to others (integration).

Today’s simulation systems can provide these capabilities if content developer(s) have sufficient

technical skills and resources. The distinguishing factor of S2I is that users are able to create content for these purposes without requiring specialized technical skills or expertise in instructional design.

S2I meets these user requirements by drawing content directly from the simulation and scenarios already in use for existing practice. This approach then leverages the simulation training pipeline to semi-automatically produce these additional instructional products. The notion of using simulation to produce instructional products is not new; in fact, it has been a core element of RIDES (Coller et al., 1991; Munro & Towne, 1992; Towne & Munro, 1992) and iRides (Munro, 2003) for more than two decades. RIDES anticipates some aspects of S2I by defining explicit demonstration and test modes that are direct ancestors of some of the S2I capabilities presented herein. Other simulations have sometimes also supported demonstration modes.

S2I is technically feasible and scalable today however, because automatically storing the results of simulation experiences now has negligible cost. Recorded simulation traces (which typically are intended to support session replay for after action review) then become “content” for demonstrations, assessments, and stand-alone instruction. Content development activity shifts to search (finding a good example from the data store) rather than construction. Search is faster and simpler. Simplicity extends the audience of potential content developers because little technical skill is required.

This paper presents current progress toward the S2I vision. We first describe functional capabilities within S2I and some general requirements for simulation systems and the content creation pipeline to support S2I. We then describe some implemented examples, drawn from a simulation/practice environment being used at a US Navy schoolhouse. These implementations illustrate ways the S2I capabilities can be made manifest in today’s training environments. The paper concludes with recommendations for future simulation systems that will make it easier to integrate and to streamline the Simulation2Instruction pipeline for next-generation simulation environments.

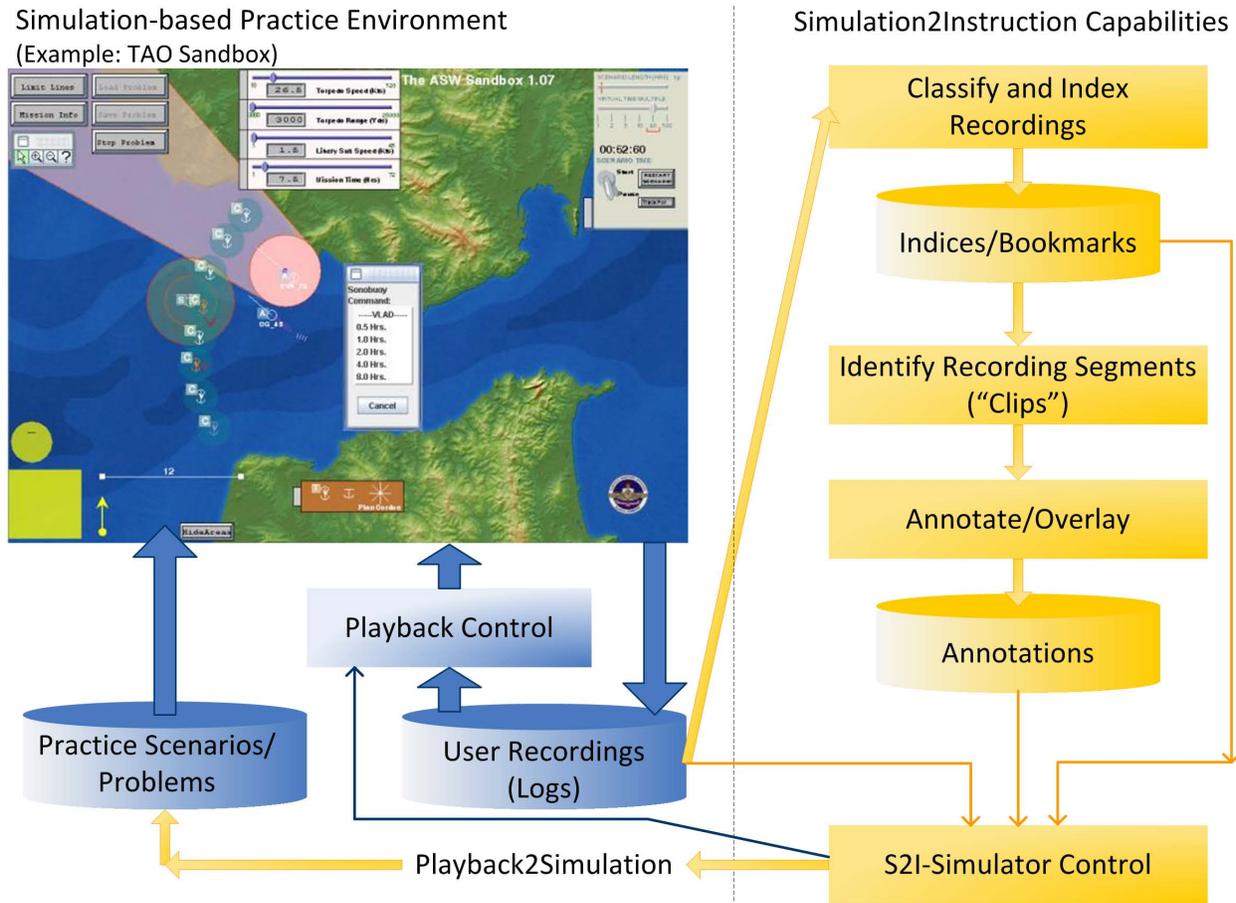


Figure 1: Overview of the Functional Capabilities of Simulation2Instruction.

SIMULATION2INSTRUCTION (S2I)

Figure 1 illustrates the core functional components and data transformations entailed by the S2I concept. The left side of the diagram depicts a typical simulation-based practice environment. Students load practice scenarios or problems into the simulation environment and then complete the practice exercise. As the simulation advances, a recording of the student's actions and the resulting, evolving situation is created. After having completed the exercise, a user (the student and/or instructor) can playback the recording for after-action review, interactive discussion and assessment, etc. (As we discuss further in the next section, although metaphorically it is simple to think about these user recordings as video recordings, it is important for Simulation2Instruction that events be recorded as logs rather than captured as video.)

The right side of the figure outlines the functional capabilities that comprise Simulation2Instruction. The first step in the process is to classify and index recordings. Classification should take advantage of any

meta-data associated with the problem/scenario (name of the scenario, learning objectives, etc.) and any overall assessments or evaluations that were made during student practice (i.e., as the recording is being created). S2I is designed to support both pre-defined classifications ("tags") as well as user-created classifications, enabling an extensible and instructor-customizable classification method.

Indexing is the process of identifying specific segments of interest within a recording. The index is similar to a bookmark (to use a web analogy) or a "key frame" from the world of film editing. The indexing capability identifies key frames to enable quick cueing to portions of interest within a recording. For example, if a recording was classified with the tag "error of omission," key frames could be inserted to mark the beginning and end of the portion of the recording where that error occurred.

Once recordings are classified and indexed, instructors must be able to find relevant recordings, and, as importantly, specific segments or clips within

recordings. Classifications and indices become search terms to identify specific recordings. The search function is used in tandem with the S2I-Simulation Control component to bring up and review candidate recordings. We are currently investigating techniques to categorize logs and automatically create indices (see Recommendations, below). However, instructors are presently willing to create indices in recordings manually (some simple meta-data is captured automatically) in order to benefit from S2I capability.

Once a recording (or set of recordings) has been identified for some instructional purpose (e.g., a demonstration or a test), an instructor or content developer can author annotations or overlays for the simulated context. Annotations are used within the rendered simulation experience to add instructional information. Examples include a simple text string (e.g., the directive “Learn about the situation and your mission” in Figure 3, as discussed further in the use case section), graphical annotations, such as an arrow or a polygon outline (e.g., a semi-transparent rectangle centered on some object to which student attention is addressed), and a “voice over” that accompanies playback of the segment.

An overlay is a distinct screen area like a window (again see the example in Figure 3). It is not meant to be integrated within a simulated scene (i.e., it appears “over” a rendered scene). Overlays can be used for supplementary text, graphics, and videos that would be technically challenging to integrate within a rendered scene. In current implementations of S2I, annotations and overlays are authored by instructors, content developers, and students using familiar authoring concepts (e.g., PowerPoint abstractions). Long-term, it may be possible to semi-automate annotation of recordings, but such capability is not a near-term goal.

The annotated recordings are now available to be presented as instruction. An S2I-Simulator Control component pulls together the specified content in a form that can then be provided to simulation Playback Control. A few simulations today readily support annotations, overlays, and streaming multimedia sources during playback, although most do not. As a consequence, the S2I Simulator Control component must be specialized for individual simulation environments, filling the gap between the simulation’s playback capabilities and media requirements.

Finally, S2I offers the ability to transition from a particular point in a playback recording to active practice starting at that point (Playback2Simulation). Playback2Simulation may also create, as a by-product of instantiating the recording state as a scenario state, a

new scenario file, representing the situation when the user initiated Playback2Simulation.

SIMULATOR REQUIREMENTS

Long-term, we envision S2I as a set of reusable algorithms and software capabilities that can be used across multiple simulation environments. As suggested by the vertical line separating the simulation and the S2I capabilities in Figure 1, the organization of S2I is designed to impose minimal special requirements on the simulation environment itself. This decoupling is important from multiple points of view:

- Content developer perspective: Programming skill is not required to use simulation content for instruction;
- Program development perspective: Instructors and content developers can continually capture and use content as the overall set of simulation capabilities develops over time; and
- Policy/acquisition perspective: Wholly novel requirements do not need to be introduced to support S2I when a simulation is acquired or upgraded.

In initial investigations and implementations of S2I, we have observed the following general simulation-system requirements that are needed to make Simulation2Instruction technically feasible as a reusable software capability that can be effectively employed by non-technical users:

1. Logging of simulation activity
2. Playback of simulation logs with annotations
3. External scenario definition to enable scenario construction for Playback2Simulation

Logging Simulation Activity

S2I’s feasibility depends on a near-complete capture of simulation events, including user actions, simulation events, and indirectly observable simulation decisions (e.g., combat results; goals/actions of virtual entities). Today’s computer games demonstrate the feasibility of fully logging user action in a multiplayer dynamic environment for playback. For example, every version of Epic Games’ Unreal Engine includes a **demorec** capability which creates a logged recording for playback by the Unreal Engine.¹ Unreal Tournament servers can record the actions of all players in a multiplayer environment and provide playback from alternative points of view. Training simulations are increasingly including this capability as well, primarily as a requirement to support after-action review.

¹ www.unrealengine.com

Table 1: Benefits of Video- vs. Log-based Capture of Simulation Traces**Benefits of screen/user capture recordings**

- What You See Is What You Get
- Exact repeatability
- 3rd-party instrumentation
- Low-cost, COTS tools for video editing and manipulation

Benefits of log-based capture

- Ubiquitous recording (due to significantly reduced storage)
- Easier to classify and index via semi-automated and automated processes
- Required for Playback2Simulation
- Easier to coordinate interoperation with other simulators or data sources (voice recordings)
- Easy to replay from alternative perspectives

Most logging processes abstract details that the simulator itself can readily reproduce. For example, launching a missile at a particular point in a scenario may be the only event that is logged and the flight of the missile during playback could be computed by the simulation. This kind of logging process makes the log more compact and effectively utilizes simulator capability. However, it implies that the simulator must record every variable that is needed for accurate playback of the experience. As a simple example, if the launched missile had some probability of kill in the original simulation experience, the playback needs to reflect whatever happened during the simulation and not re-compute Probability(kill) during playback.

The challenges of logging for accurate playback might recommend screen capture technology as an alternative option for S2I. Table 1 summarizes some of the benefits of the two alternatives. Most notably, because logs require orders of magnitude less storage (and storage is inexpensive), it is feasible for every student's experience in every scenario to be captured. For example, in the simulation system we discuss below, an hour of simulation time can be captured in a log on the order of 10 kb in size. On the other hand, the storage requirements for automatic screen capture are large enough that pervasive capture of student experience is not feasible. For example, FRAPS, a common tool used for screen capture by players of commercial video games, can require as much as 50-75Gb of storage for every hour of playing time in immersive, 3D simulation environments.

Logged recordings will also better facilitate automatic indexing by providing a comparatively semantically rich data source in the absence of categorization of user

behavior captured during run-time. Finally, logs appear to be necessary for Playback2Simulation because logs can record events and states that are not observable, whereas screen capture only reflects observable states.

Playback with Annotation

A second requirement for simulation systems is that minimally they be able to support rendering annotations along with playback of the recorded scene. Many simulations and computer games have graphical and text objects that can be repurposed for use as annotations. In essence, these objects define a language of annotation for that simulation environment. The richer the underlying language, the better able the S2I capabilities will be able to use annotations to support instruction within that simulation.

To illustrate, in a dynamic, multi-object environment, it would be preferable for annotations to bind to objects as they move in space and time. For example, in Figure 3, it might be useful to have text annotations appear next to and move with the ship that the student is commanding in the simulation. If a simulator supports such bindings, then it is trivial to map an annotation object to a simulation object in a recording via the authoring tool. If the simulator does not support such bindings, then the obvious approach to manage dynamically moving objects is to animate their movement within the annotation authoring tool, although such authoring is tedious. S2I annotation tools currently support manual animation because spatial/temporal object-bindings are not usually an exposed capability within a typical game's Playback Controller (see Recommendations, below).

Moving Seamlessly between Instruction & Practice

In today's simulated environments, there is usually a strong separation between the non-practice phases of instruction and practice. For example, an instructor might present a briefing of some concept, a student might view an introductory video, and then begin practice in the simulation, sometimes physically moving to a new environment to practice.

One of the hypothesized benefits of S2I is that it breaks down the expedient but arbitrary separation of practice from other elements of instruction. In addition to presenting simulation content annotated for demonstration, assessment, and reflective practice, S2I also enables direct transition from a recorded experience (whether the student's or someone else's) to immediate practice.

In Figure 2, imagine a student in the process of playing back a previously recorded scenario. The recording might be a previous experience of the student, another student, or a scenario recorded by the instructor. The recording playback progresses through a series of world states that are rendered to the student. When the recording reaches the situation represented in the figure as state 4, the student wishes to begin a new practice at this point. This practice will enable the student to act within and to visualize the resulting situation that unfolds from that point by having taken a different decision. Also, at this point, a new scenario file can be created, representing the situation at state 4. The student begins practice from this point, with a new recording being automatically created as the student interacts in the simulation.

Playback2Simulation is the S2I capability for enabling such immediate transition from playback to the simulation. As we outline below, we have implemented this capability in two separate simulation environments. Anecdotally, instructors for both simulations environments have related that they see Playback2Simulation as a highly compelling capability for use within their programs of instruction.

A possible by-product of the Playback2Simulation capability is the creation of a new scenario file, instantiated with the current situation at the time in the recording when Playback2Simulation was invoked. Because annotations and indices are separate from the recorded log, they can be readily excised from the new scenario file. This approach also makes it feasible to “splice” the original recording and the recording generated from the Playback2Simulation experience together to create a new recording. In Figure 2, for example, it would be possible to construct a new recording (1,2,3,4,E,F,G,H), enabling a student or instructor to piece together a demonstration from a number of individual practice exercises.

SIMULATION2INSTRUCTION USE CASES

Having outlined the Simulation2Instruction concept and requirements for simulation technology to make

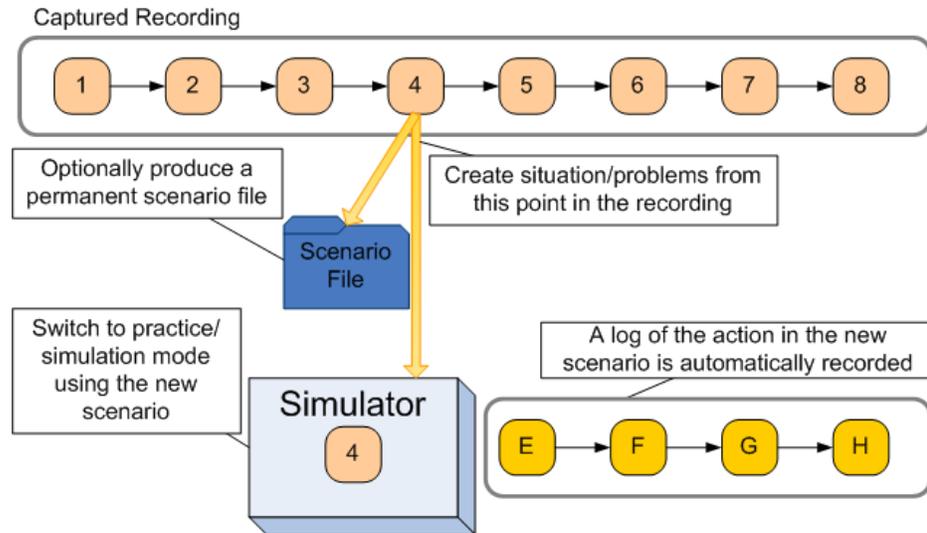


Figure 2: The Playback2Simulation Concept.

the technology feasible, we now outline several implemented examples intended to convey more concretely both the potential benefits of S2I and the specific progress we have made toward realizing the vision. Examples include 1) Demonstration, 2) Assessment, and 3) Reflective practice.

To illustrate, we draw from an-going simulation training effort that embeds S2I concepts while presenting realistic problems and varied practice. *TAO Sandbox* provides a problem-centered environment for demonstration, practice, and assessment in the domain of tactical action planning for surface warfare (Munro, Pizzini, & Bewley, 2009). The TAO Sandbox enables both instructor-mediated assessment as well as a rapid skill assessment capability (*DARTS*), both of which are outlined below.

The TAO Sandbox is currently in use at the US Navy Surface Warfare Officers School (SWOS). In terms of the S2I concept, TAO Sandbox simulation automatically makes simulation recordings or logs that are captured during the course of a scenario. These logs are then used by additional components (some of which are part of the TAO Sandbox executable, some of which are wholly separate) to support and realize the use cases below.

Using Simulation for Demonstration

Demonstrations in the context of simulations can be used for several purposes, including:

- Instructors demonstrate in the classroom.
- Students present their individual solutions to problems by stepping through recordings.
- Students study pre-authored demonstrations (for self-study, remediation, etc.).



Figure 3: Recorded Session embedded with Guiding Annotations.

Instructor demonstrations in the classroom. At SWOS, instructors use the TAO Sandbox for demonstration in two different ways. The first approach is to use *live demonstration*. An instructor authors a scenario that can be used to illustrate a point about surface warfare tactics in advance. In class, the instructor shows how the problem can be addressed, commenting on actions and events. The second approach is to use a *recorded demonstration*. In this case, the instructor pre-records a session with a problem, and then provides commentary while playing the session back to students. One of the advantages of the playback approach is that the instructor has full control of playback and can stop, start, initiate discussions, etc. A basic simulator recording capability is sufficient for these use cases.

However, demonstrations may be more meaningful when they are accompanied by annotations that highlight options and decision rationales. The TAO Sandbox includes recording editing features (Annotate function) that can be used to annotate recorded sessions with brief text messages and attention directing graphics, such as arrows. The Annotate function thus supports a third approach to in-class demonstration that uses recordings that include these annotations as part of the instructor's presentation of the demonstration.

Students present solutions in class. At SWOS, instructors sometimes assign groups of students to attempt different tactics problems in the TAO Sandbox. A group might have enough time to record several attempts to solve a problem. They select the best one to present, and members of the group describe their decisions as the recording is played. They can use the

Annotate function to highlight key decisions or untaken alternatives in these After Action Reviews.

Study pre-authored solutions. Instructors can build a library of commented solutions to tactics problems with a variety of challenges, using the 'recorded session with comments' feature of the TAO Sandbox (see Figure 3)². There is some evidence that having a student review problems solved by others can be as useful as hands on practice

(Sweller & Cooper, 1985). Students can study solved problems on their own. The TAO Sandbox supports stopping a recorded session playback and resuming live play at any point, so students can combine viewing demonstrations with active practice.

Instructors can record audio narratives to accompany the playback of pre-recorded simulation sessions. Pause and play buttons on the session Playback Control interface will also pause and play the audio file for the session recording. We are using these pre-authored solutions in conjunction with the assessment approach outlined below to deliver individualized remediation lessons for students based on assessment results.

Using Simulation for Assessment

Recorded examples of simulation can provide a large library of content for creating situated questions that can be used for different kinds of assessments. In our work to-date, we have explored/are exploring using captured and annotated recordings to support:

- Readiness assessment
- Instructor-mediated assessment
- Formal and informal assessments of progress
- Fine-tuning of practice-based assessments

A separate software component, DARTS (Diagnostic, Automated Rapid Testing of Skills) has been developed as an adjunct to the TAO Sandbox. DARTS is designed to provide assessments in the context of recorded simulations sessions. In authoring mode, instructors create individual test items by choosing

² All examples and screenshots have been constructed by the authors solely for illustration purposes. They are not drawn from curriculum materials used at SWOS.

Table 2: Summary of Assessment and Reflective Practice Use Cases enabled by S2I for DARTS/TAO Sandbox.

	Capabilities	Potential Uses
Readiness Assessment	<ul style="list-style-type: none"> • Enable use of captured content to situate test items in the decision making context • Construct tests for specific topics/learning objectives • Automatically deliver and grade rapid skill tests (timed responses) • Recommend remediation lessons based on test outcomes 	<ul style="list-style-type: none"> • Readiness assessments for incoming students • Assess readiness for specific units within classroom component of the course ("ASW readiness") • End-of-unit testing/validation (complement to written tests)
General Testing Platform	<ul style="list-style-type: none"> • Support authoring of questions oriented toward conceptual knowledge as well as decision making • Alternative question types (multiple-choice, select some/all, drag and drop, etc.) • Remove time-pressure for responses 	<ul style="list-style-type: none"> • In-class pop quizzes • "Readiness checks" for role players/console operators • Computer-based tests to complement conceptual knowledge aspects of (written) end-of-unit tests
Self-study Mode	<ul style="list-style-type: none"> • Remove time constraints on answering questions • Deliver immediate feedback (right/wrong response) • Offer pointers to source/instruction materials (e.g., "see NWP X-Y, Ch. 6") • Enable immediate review of remediation lessons (when available) 	<ul style="list-style-type: none"> • Review results of a assessment/test • Homework assignments • More open-ended path to delivery of remediation lessons
Practice Validation	<ul style="list-style-type: none"> • Use recordings with questions to clarify student knowledge and skill with respect to actions taken during practice • Use in conjunction with adaptive training curriculum to select scenarios 	<ul style="list-style-type: none"> • Provide data for choosing "best next scenario" for an individual student • Construct tests based on scenario results that increase measurement confidence
Progressive Test	<ul style="list-style-type: none"> • Present a series of successive decision steps within an overall scenario • Enable branching of recordings based on student responses 	<ul style="list-style-type: none"> • Applicable to both "assessment" and self-study use cases • Foundation for reflective practice

segments of previously recorded simulations sessions to use as the context for questions. At runtime, students are presented with the selected portion of the session together with questions and possible answers. For example, a TAO Sandbox recorded session or series of recorded sessions that present a challenge related to avoiding a Torpedo Danger Area (TDA) could be used to assess student knowledge in this area (Figure 4).

Because DARTS software depends only on captured recordings, it is flexible enough to support many complementary assessment needs. Table 2 summarizes use cases to which DARTS S2I assessment capability is being applied. In addition to assessing individual students for readiness for a course/unit within a course and a general testing platform, tests can be used with immediate feedback to provide a self-study mode. This enables the use of the same assessment content both for testing and for a crude individualized remediation.

Another potential use of DARTS is to supplement the results of practice-based assessment (*practice validation* in the table). For example, if an error is made during practice, is that an error due to conceptual knowledge gaps, inadequacy in skill, or just an interface/data entry mistake? Short DARTS tests after the practice can be used to attempt to confirm/disconfirm observations made during practice and to "drill down" to make more fine-grained assessments of knowledge and skill state.

The value of this approach is that it complements the in-situ, problem-based assessment. By its nature,

problem-based methods offer ecological validity in assessment of skills, but, in a learning environment where time-on-task is limited, typically cannot be used to span the full-range of knowledge, skills, and decision-making within a domain. Currently, these kinds of tests depend on selection of specific questions from a pre-authored repository but, as capabilities are developed to automatically assess student action and index recordings with those assessments (see next section), we envision it being feasible to construct individualized tests using a recording of a student's just-completed practice as source content for test items.

Using Simulation to Reflect on the Work of Others

As mentioned above, seeing a worked solution can be comparable in value to active practice. Similarly, it can

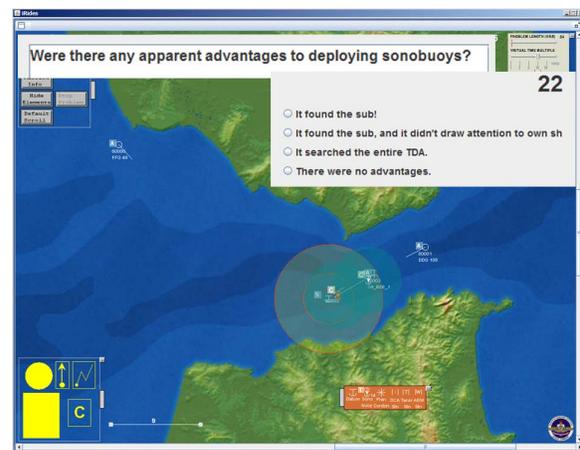


Figure 4: Example test item constructed from S2I elements integrated with DARTS/TAO Sandbox.

also be useful pedagogically to critique a solution (or partial solution) worked out by others, which supports integration (Merrill, 2002).

The automatic capture of simulation traces provides a significant first step for realizing such a capability. Uses for simulation traces might include:

- Learning by observation (vicarious learning)
- Critiquing/judging solutions
- Situational judgment tests
- A stimulation for group discussions and group evaluation

A simulation recording and playback capability is sufficient for all of these cases when the activity is led by the instructor. Similarly, the Annotation function can be used to add context and highlight salient features of a situation in the recording to support reflective practice.

We have developed another example of more student-directed reflection and critiquing via DARTS. Progressive tests (the last row in Table 2) can be constructed in DARTS that present a sequence of decisions in a specific scenario. At each decision point, students might be asked what they would do in that situation (i.e., a typical test). However, they can also be asked to judge/evaluate the appropriateness of some decision (a judgment test), or to offer rationales or critiques for some decision in the recording (a reflective prompt). Using the Playback2Simulation capability, the student can also play out alternatives and see the consequences of those different decisions.

Generalizing S2I to Additional Simulations

This paper highlights the application of S2I in a single domain. However, the S2I concept is not specific to the TAO Sandbox simulator. For example, we have also embedded S2I capabilities in an “instructional game” focused on an Army tactical decision-making domain (Wray, Woods, & Priest, 2012).

Both the TAO Sandbox and the instructional game provide two primary modes: 1) an S2I instructor interface that enables instructors to develop and/or leverage simulation content for demonstration, practice, and assessment; and 2) a problem or simulation mode that provides opportunity to practice in defined scenarios. Research and development of the instructional game has focused more on developing authoring tools that enable instructors and training developers (who will often have little instructional design background) to author good instruction paired with practice. TAO Sandbox has focused more on defining patterns of use, such as enabling instructors to create demonstrations with a scenario simulation or to

capture recordings to use for assessment and remediation.

The instructional game application includes comparable S2I recording, playback, and annotation capabilities. Because briefing simulation outcomes is one of the requirements for the Army instruction, the ability to capture and playback the student’s practice has been seen as particularly useful for classroom use. Students can use the Annotation function to add information that describes rationales for actions, responses to decisions points, and alternative outcomes to include in their briefings to classmates. This approach has the potential to be especially effective because it allows the students to present their decisions and rationales in a dynamic visualization of the battle, rather than via static screenshots and text summaries.

S2I FOR FUTURE SIMULATION SYSTEMS

The prototypes discussed in this paper illustrate that Simulation2Instruction capabilities are feasible and useful with today’s simulation technologies. Additionally, as discussed previously, these S2I capabilities can be realized without introducing significant, new requirements on training simulations.

In order to better exploit the promise of S2I and efficiently and effectively extend the reach of simulation into all phases of instruction, we recommend the following modest steps for next-generation simulation systems. Fulfilling these requirements will often have benefits in addition to supporting S2I, but following these suggestions will make it easier to employ and reuse S2I capabilities across multiple simulation environments.

Design for Student Data

Minimally, new simulations need to log what users are doing and be able to reproduce those experiences accurately. However, simulation systems should also log other potentially useful user data that is available, such as interactive audio, dynamic assessments, and any after-action assessments or evaluation.

For example, automated assessment of student action (produced as practice is undertaken) will facilitate more automatic indexing of recordings. Dynamic assessments can be provided by an instructor, by other observers and students, or by an automated assessment process. For example, the TAO Sandbox has been instrumented to detect pedagogically relevant actions and events during learner practice (Munro et al., in press). We are currently exploring how we can use these assessments to insert indices in a recording to support automatic synthesis of test questions.

Beyond logging data, organizations installing and using simulations need to plan for a lot of student data. This will be true whether S2I capabilities are part of the vision or not. Student data needs to be logged automatically, but it also must be scrubbed of personally identifying information for long-term use. There is tension between enabling uses of logged data for individualized after action review, assessment, and remediation while also supporting more general purpose uses, which requires scrubbing. This suggests a multi-stage capture process, where all logged data is available for a short period of time, and then a consolidation process would remove personally identifying information for long-term storage.

Open Architectures

Simulations typically expose little underlying functionality to content developers. This limitation is especially acute for proprietary simulations and game environments, restricting their scope and use (and ultimately, value) to the DoD. For example, the currently implemented S2I Annotation function, as described previously, assumes content developers will animate annotations manually because annotations often cannot be bound to simulation objects in proprietary simulations. However, these games and simulations usually have an object-to-object binding relation available within the simulation environment itself. Having more of this functionality exposed will provide the means to improve automation of categorization and indexing and annotation for S2I, thus requiring even less authoring.

Integration of annotations within a rendered scene also may need to be varied or integrated. For example, when is it important for annotations to be rendered within an immersive environment rather than as a flat layer overlaying 3D playback? Munro and colleagues (2002) previously evaluated alternative approaches to the presentation of instructional information in 3D learning environments. For example, objects in a 3D environment can offer access to relevant information by having in-scene menu objects appear. Without an open architecture that allows direct access to object information and object states, these recommended approaches to presenting and integrating annotations within a virtual scene may be impossible to satisfy.

Design for Interleaved Simulation and Playback

The Playback2Simulation concept offers potential as a next-generation “killer feature.” The S2I prototypes have demonstrated the technical feasibility of such interleaved simulation and playback. However, to support this capability more fully and generally in

future simulation systems, we recommend the following:

1. Scenarios (or problems) should be *writable by external processes*. This requirement is more programmatic than technical. For an external process to create a scenario, the scenario format or language must be exposed to 3rd-party developers and content developers. Proprietary scenario formats are not able to accommodate this feature. Although not strictly a requirement, scenario formats designed with embedded meta-data and well-defined semantics (e.g., RDF, or OWL) will generally be easier for S2I tools to access and use.
2. All relevant *state information must be included in the scenario file*. Consider the prior example of the simulator that is flying a missile during playback. The existence of the missile and its current state must be captured in the new scenario file.
3. *SAFs should generally be re-entrant* for all states. Re-entrant design is a feasible requirement for SAF technology (Wray et al., 2005). Even without fully re-entrant SAFs, most SAFs will be able to reconstruct a situation assessment (but not a history) from a combination of simulation state and internal state that will be acceptable for playback. However, when the history is relevant to SAF action, simulation following playback may introduce some quirks due to the missing history.

CONCLUSIONS

Simulation has typically focused on realistic practice. However, effective instruction incorporates more than practice; S2I focuses especially on demonstration and opportunities for reflection and integration. In this paper, we have outlined a collection of technical capabilities (Simulation2Instruction) that together extend the reach of simulation to these other phases of instruction, while making minimal new demands on existing simulation technologies.

S2I exploits the emerging pervasiveness of simulation recordings. It uses these recordings to partially automate the creation of simulation content for instructional purposes. Instead of designing a demonstration of a procedure, the instructor can instead search to find a few candidate recordings, find the portions of the recording relevant to the demonstration, and create graphical and audio annotations that are rendered in synchronization with playback to highlight concepts, ask questions, or prompt reflection.

We illustrated each of these use case, showing S2I capabilities implemented within or integrated with the

TAO Sandbox, a practice environment currently in use at the US Navy's Surface Warfare Officers School. These implementations illustrate both the technical feasibility of the S2I concept and suggest their general utility across the span of instruction. We are also currently planning training effectiveness evaluations of specific S2I capabilities, focused primarily on the DARTS assessment capability and its use within an adaptive training program. To suggest the generality of the S2I concept, we outlined an application in which we are using comparable recording, playback and annotation capabilities in a different simulator for a different training community.

Simulation2Instruction will become more pervasive and more valuable as it relies less on manual indexing and authoring of captured recordings and more on automated categorization, indexing, and annotation. We outlined preliminary steps toward the goal of greater automation of categorization and indexing. Satisfying the proposed recommendations for next-generation simulators will make employing S2I capabilities easier and more effective and, consequently, deliver the benefit of simulation throughout all phases of instruction.

ACKNOWLEDGEMENTS

This work is supported in part by the Office of Naval Research. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Department of Defense or Office of Naval Research. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon. The S2I foundation and the DARTS assessment capability were supported by ONR contracts N00014-10-C-0526 and N00014-11-M-0504. Development of TAO Sandbox was supported by the Office of Naval Research: N00014-08-1-0126, N00014-08-C-0563, N00014-09-C-0813, N00014-10-C-0265, N00014-10-C-052670, We would like to thank Laura Hamel, Angela Woods, Quentin Pizzini, and David Surmon for their contributions to the development of the S2I prototypes described herein.

REFERENCES

- Coller, L. D., Pizzini, Q. A., Wogulis, J., Munro, A., & Towne, D. M. (1991). *Direct manipulation authoring of instruction in a model-based graphical environment*. Paper presented at the international conference on the learning sciences: Proceedings of the 1991 conference.
- Merrill, M. D. (2002). First principles of instruction. *Educational Technology Research and Development*, 50(3), 43-59.
- Micheli, G. S. (1972). *Analysis of the transfer of training, substitution, and fidelity of simulation of transfer equipment*. Orlando: Naval Training Equipment Center.
- Munro, A. (2003). Authoring simulation-centered learning environments with Rides and Vivids. In T. Murray, S. Blessing & S. Ainsworth (Eds.), *Authoring Tools for Advanced Technology Learning Environments*. Dordrecht, The Netherlands: Kluwer Academic Publishers.
- Munro, A., Breaux, R., Patrey, J., & Sheldon, B. (2002). Cognitive Aspects of Virtual Environments Design. In K. Stanney (Ed.), *Handbook of Virtual Environments* (pp. 415-434). Mahwah, NJ: Lawrence Erlbaum Associates.
- Munro, A., Pizzini, Q., & Bewley, W. L. (2009). *Learning Anti-Submarine Warfare with a Game-Like Tactical Planner*. Paper presented at the IITSEC 2009.
- Munro, A., Surmon, D., Koenig, A., Iseli, M., Lee, J., & Bewley, W. (in press). Detailed modeling of student knowledge in a simulation context. In V. Duffy (Ed.), *Proceedings of the 2012 applied human factors and ergonomics conference* (Vol. 3, Advances in applied modeling and simulation). Boca Raton, FL: CRC Press, Taylor and Francis.
- Munro, A., & Towne, D. M. (1992). Productivity tools for simulation centered training development. *Educational Technology Research and Development*, 40, 65-80.
- Sweller, J., & Cooper, G. A. (1985). The use of worked examples as a substitute for problem solving in learning algebra. *Cognition and Instruction*, 2(1), 59-89.
- Towne, D. M., & Munro, A. (1992). Two approaches to simulation composition for training. In M. Farr & J. Psotka (Eds.), *Intelligent instruction by computer: Theory and practice*. London: Taylor and Francis.
- Wray, R. E., van Lent, M., Beard, J. T., & Brobst, P. (2005). *The Design Space of Control Options for AIs in Computer Games*. Paper presented at the IJCAI 2005 Workshop on Reasoning, Representation, and Learning in Computer Games, Edinburgh, Scotland.
- Wray, R. E., Woods, A., & Priest, H. (2012). *Applying Gaming Principles to Support Evidence-based Instructional Design*. Paper to be presented at the 2012 Interservice/Industry Training, Simulation, and Education Conference.