Tales of Authoring Tools with Lessons

Sowmya Ramachandran
Eric Domeshek
Stottler Henke Associates Inc.

sowmya@stottlerhenke.com
domeshek@stottlerhenke.com

Intelligent Tutoring System Authoring Tools: Harvesting the Current Crop and Planting Seeds for the Future

ITS Authoring Challenges

- The space of variations in tutoring systems large:
  - Domains: Reading, Algebra, Military Tactics, Emergency Response
  - Domain-Specific Problem-Solving Environments: Simulations, Games, Algebra Workbooks
  - Design approaches: Cognitive Tutors, Constrain-Based Tutors, Pseudotutors, Dialog-Based Tutors

You can't be everything to everybody

Making Knowledge Cheap

- ITS are knowledge-intensive systems
- Why should we expect that machines are easier to teach than humans? We do not, but perhaps
  - Knowledge reuse and transfer could make acquisition cheaper

All wish to possess knowledge, but few, comparatively, are willing to pay the price.

Paths to Solutions

- Be something to a few people
  - Focus on classes of similar domains
  - Focus on specific classes of assessment and pedagogy
  - Build tools to author specific classes of knowledge
- Facilitate reuse and transfer
- Teach, don't tell
  - Use machine learning for knowledge acquisition

Agenda

- Brief overview of our authoring tools
  - Two video demos
- Lessons Learned

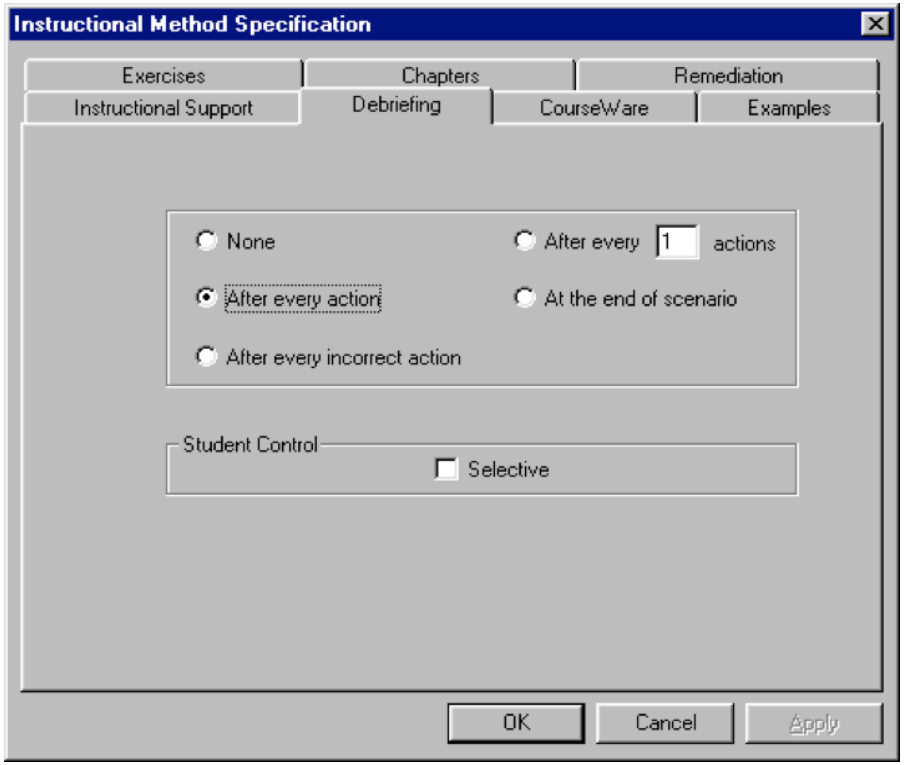
Stottler Henke

- 50 person company
- Mission is to conduct R&D and technology transfer. Not in the business of basic research.
- ITS project profile
  - Typically military domains
  - Typically simulation-based
  - Short timelines: 6mos – 2 years
  - Funding range: $500K-$3M

SH Authoring Tools

IITSAT

- IITSAT
  - Authoring for pedagogy
  - Target: Instructional Designer

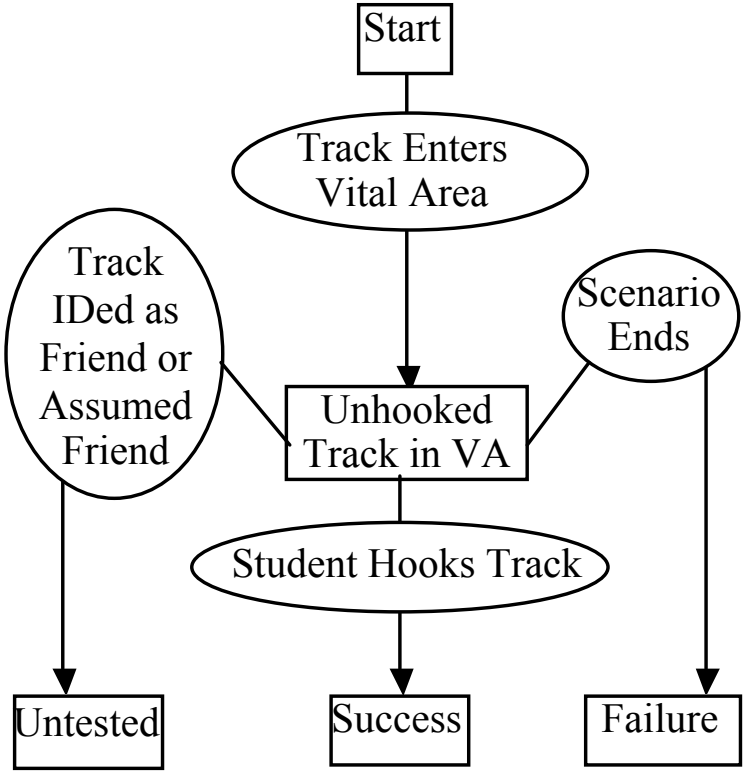


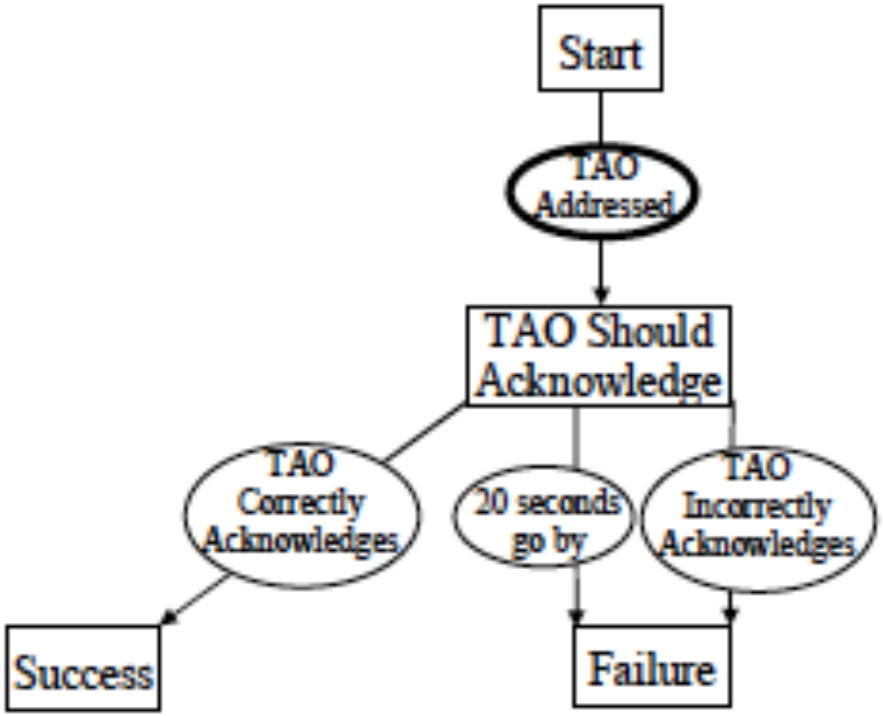
SimBionic

- Authoring system for augmented finite state machines
- Used extensively in-house for performance assessment
- Target: Developer

SimBionic Demo

SimBionic Example




Start
TAO Addressed
TAO Should Acknowledge
TAO Correctly Acknowledges
20 seconds go by
TAO Incorrectly Acknowledges
Success
Failure

SimCore

- Add intelligent tutoring capabilities to existing simulations
- Also includes a 2D, low-fi, authorable simulation tool
- Automated assessment and coaching for procedural tasks
  - E.g. following a protocol or a standard operating procedure
- Uses procedure templates for performance assessment
- Target: Subject matter experts/instructional designers with some modeling skills
- Has been used for:
  - Emergency first response training
  - Veterinarian care protocols

SimCore Demo

FlexiTrainer

- Flexitrainer Authoring Tool to enable rapid specification of the knowledge needed for intelligent, adaptive tutoring
  - Behavior Transition Networks (BTNs) enable detailed performance assessment and tutoring
  - The tutor is a collection of agents using event-based communication
  - Architecture makes it easy to integrate FlexiTrainer with different simulators
  - Agent-based architecture makes it extensible in regard to tutoring behaviors and student modeling

Instructional Planning Example

- Have implemented a simple scheme for goal-directed instructional planning
  - The Instructional Planner (IP) posts goals based on the student model
    - E.g. increase_mastery(skill)
  - Different teaching agents respond to different goals with information about how they will achieve that goal
  - IP brokers among the agents using a set of rules

FlexiTrainer Authoring Tool: Editors

- Form-based Curriculum Editor enables
  - specification of network of domain principles
  - association of exercises with principles with coaching messages, and assessment behaviors
- Visual Behavior Editor enables
  - the specification of tutoring and assessment behaviors

FlexiTrainer - Edit Instance - [stack_three_blocks_easy]

File Edit View Tools Build Help

Insert Type | Insert Instance

TSP | Exercises | SM | EF | SP | Behaviors

- Exercises Catalog
  - exercise
    - stack_three_blocks_easy

Property Values:

Property	Value
Name : string	stack_three_blocks_easy
Description : string	
tasks taught : tasks collection	stack_blocks
skills taught : skills collection	maintain_a_straight_line
principles taught : principles collection	coordination
number_of_blocks : integer	3
size_of_block : string	large
help_parameter : help_level	(none)

Build | Find

Lessons Learned

Goal: Ecosystem of tools

- ITS requirements are too diverse to be handled by a generalist authoring tool
- Building a community of specialists maybe easier and more tractable than building a generalist
  - An extensive toolkit to handle specific subsets of requirements (domain, pedagogical, problem-solving environments)
  - A shared operating environment for the tools to interoperate
  - E.g. FlexiTrainer, GIFT

Domain Model Bottleneck

- Establishing an architecture not sufficient for effective content creation by developers/experts/SMEs
- Domain modeling for scenario generation and assessment consume bulk of design and development resources
  - Domain model tends to be largely domain specific
  - Tools like SimCore can be useful for authoring models specific types of domain
- Tools to support such modeling must also be built as a part of the kit
  - Usability is a huge concern
  - Even with such tools, ITS development is likely to be resource-intensive

It Takes a Village…

- Amassing a good set of specialist tools with adequate coverage takes time and money
- Toolkits do not magically appear fully populated
- Until then, it is a challenge to persuade people to buy into an early stage ecosystem
  - Unproven path, cost savings not clear,
  - The need to conform to a common operating environment may not be worth the benefits
- There must be a plan for effective marketing of the system to users
- Continued funding must be obtained to upgrade the systems over time
- This is best done as a cross-organizational effort involving large participation from the research and development community
  - Open-source model

There are some low-hanging fruits

- Some tutoring component requirements less varied than others
  - Curriculum specification
  - Pedagogical knowledge
- One general or a small set of specialized solutions could be a sufficient set for many ITS efforts
  - Domain-independent solutions can be effectively designed

Power vs Usability

- Power and flexibility of knowledge representation are at odds with tool usability
- The use of machine learning could help
- A tiered approach to authoring is one way to address this trade-off. E.g.
  - First tier: Authors with specialized skills (programming/scripting) set up knowledge structures in the form of templates
  - Second tier: SMEs/Instructional designers select and populate template with content

Tiered authoring for Reuse

- In addition to helping usability, tiered authoring could also help with reuse. E.g.
  - A first layer serves to define generalized knowledge that is common to all situations
    - e.g. procedure templates
  - A second layer serves to specialize that knowledge to specific situations
    - E.g. extensions of templates with overrides and additions

Consider COTS tools

- Before building a custom authoring tool, it is useful to consider the use of COTS tools
  - We have used Excel for effectively authoring content for dialog-based tutors
  - We used MS-Word as a content authoring tool for a reading comprehension ITS. The authors preferred to use Word over a custom authoring tool for much of the authoring workflow.

- Benefits: Available, familiar, robust