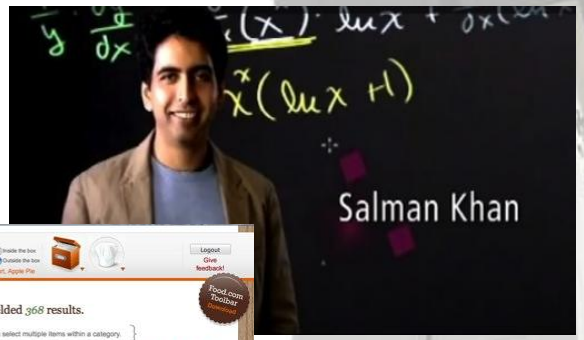
A Tutoring Page Markup Suite for Integrating Sharable Knowledge Objects (SKO) with HTML

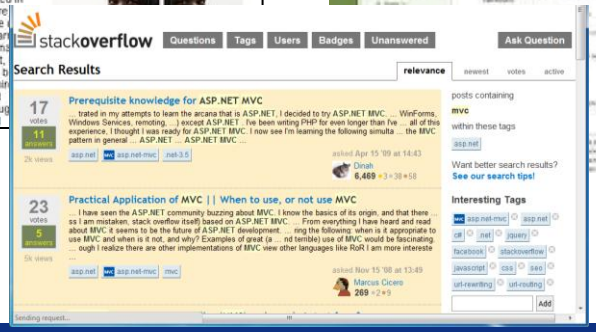
Benjamin D. Nye, Mohammad Faisal Rahman, Man Yang, Patrick Hays, Zhiqiang Cai, Arthur C. Graesser, and Xiangen Hu

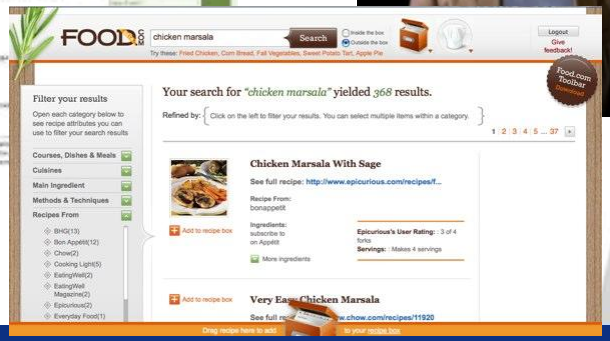
The University of Memphis

Goal: Add Tutoring to Web Content

- Current Situation
  - The Internet has **vast learning resources**
  - Almost all of these are **non-interactive**
  - Some of these are **high-quality**
  - Wide range of domains (STEM, medicine, how-to…)



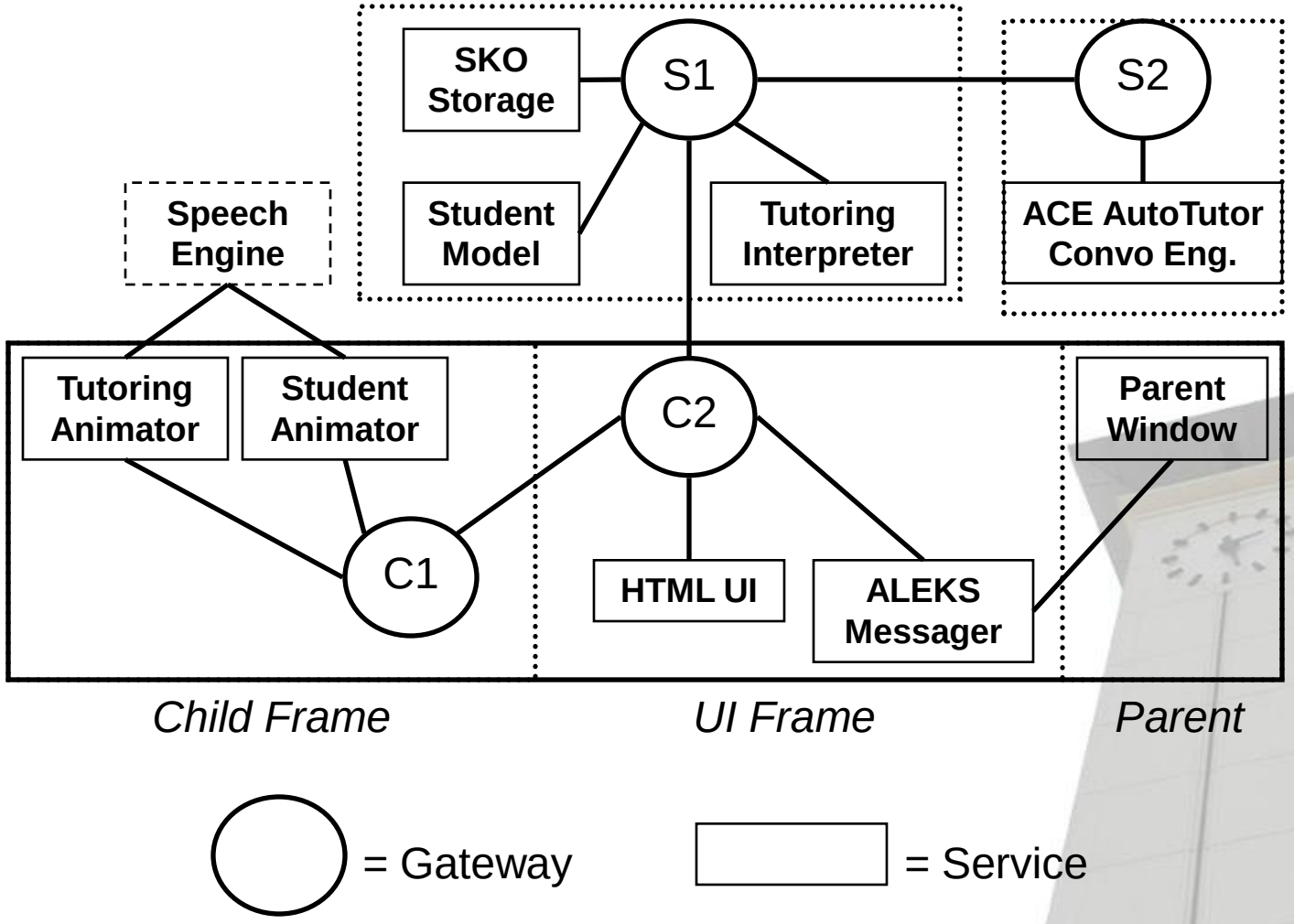




Goal: Add Tutoring to Web Content

- Approach: Tutoring as a Service
  - Mark up existing HTML with targeted ITS activities
  - Use natural language ITS to talk about content
  - Enhance and assess knowledge simultaneously
  - Simplify authoring for domain and pedagogy experts

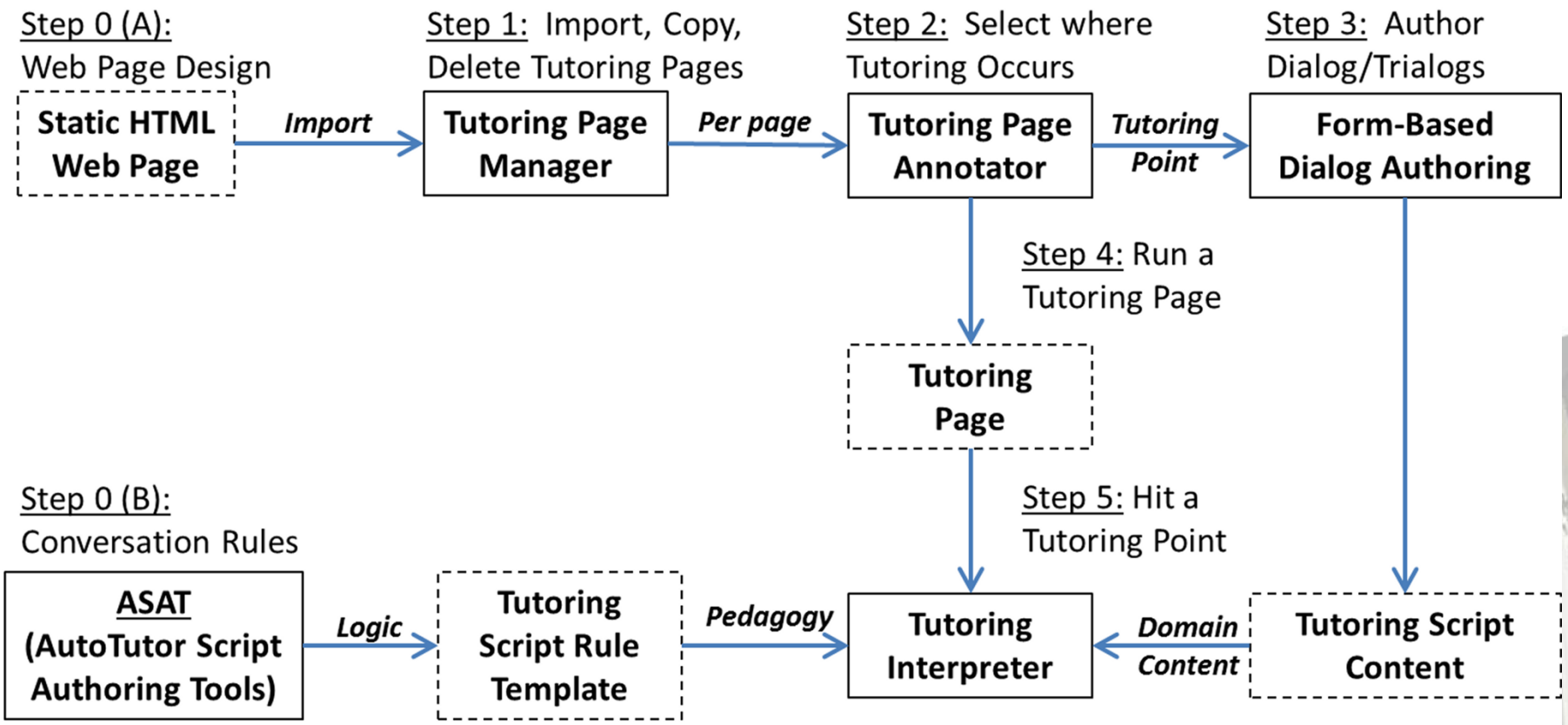
Services Under the Hood



Tool Suite

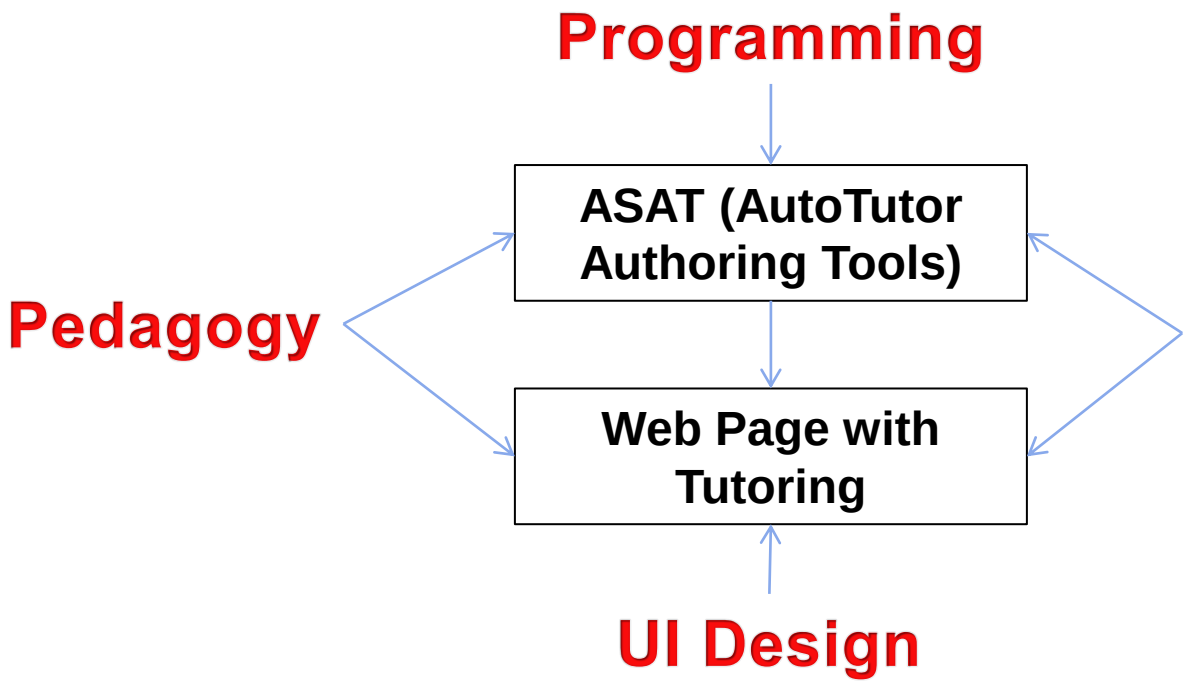
- Tutoring Page Manager: Create/Import/Delete Pages
- Tutoring Page Annotator: Tutoring Locations
- Form-Based Authoring Tool: Tutoring Content
- (ASAT): Tutoring Logic Template
- (Python Scripts): Content pre-processing for templates
- (HTML/CSS Editors): Display Formatting

Tool Suite


Step 0 (A):
Web Page Design
Static HTML Web Page
Import
Step 1: Import, Copy, Delete Tutoring Pages
Tutoring Page Manager
Per page
Step 2: Select where Tutoring Occurs
Tutoring Page Annotator
Tutoring Point
Step 3: Author Dialog/Trialogs
Form-Based Dialog Authoring
Step 4: Run a Tutoring Page
Tutoring Page
Step 0 (B):
Conversation Rules
ASAT (AutoTutor Script Authoring Tools)
Logic
Tutoring Script Rule Template
Pedagogy
Step 5: Hit a Tutoring Point
Tutoring Interpreter
Domain Content
Tutoring Script Content

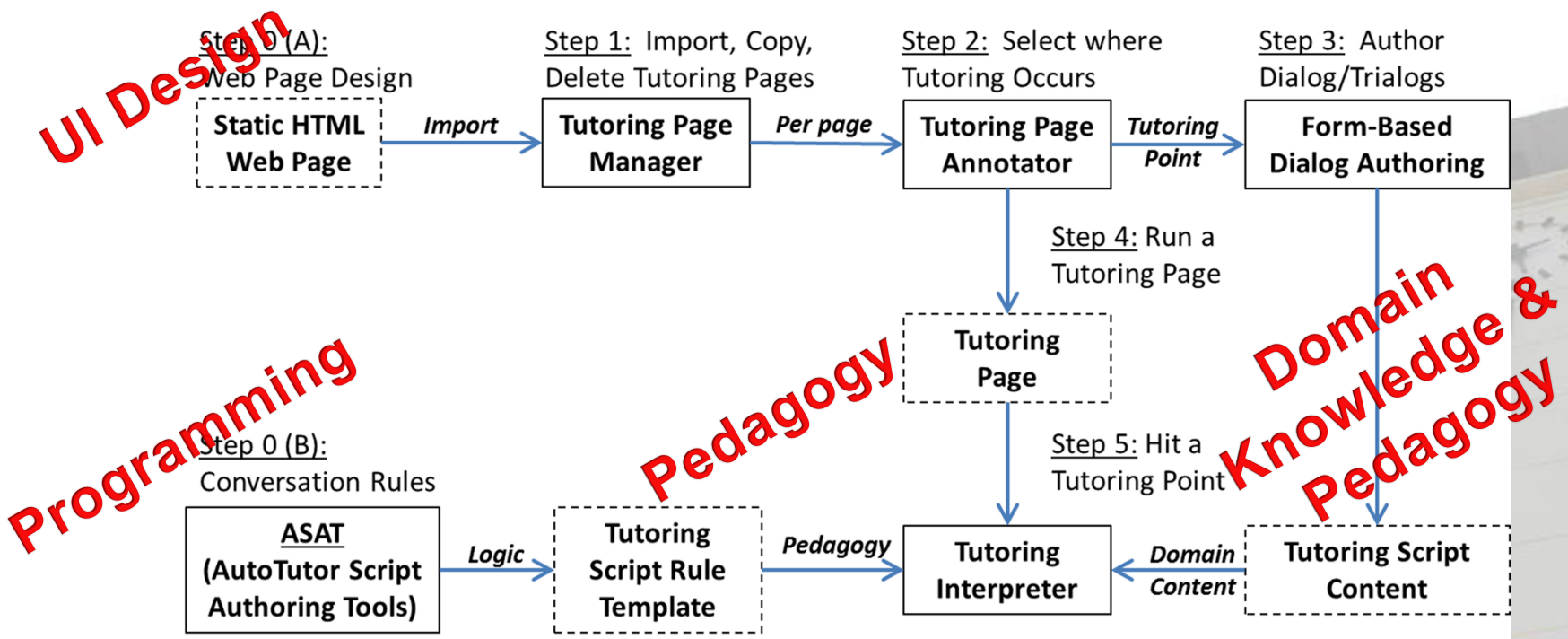
Theoretical Underpinnings

- Our Traditional Approach
  - Super-Experts: Specialized graduate-level researchers
  - Close Team Collaboration: Combine expertise

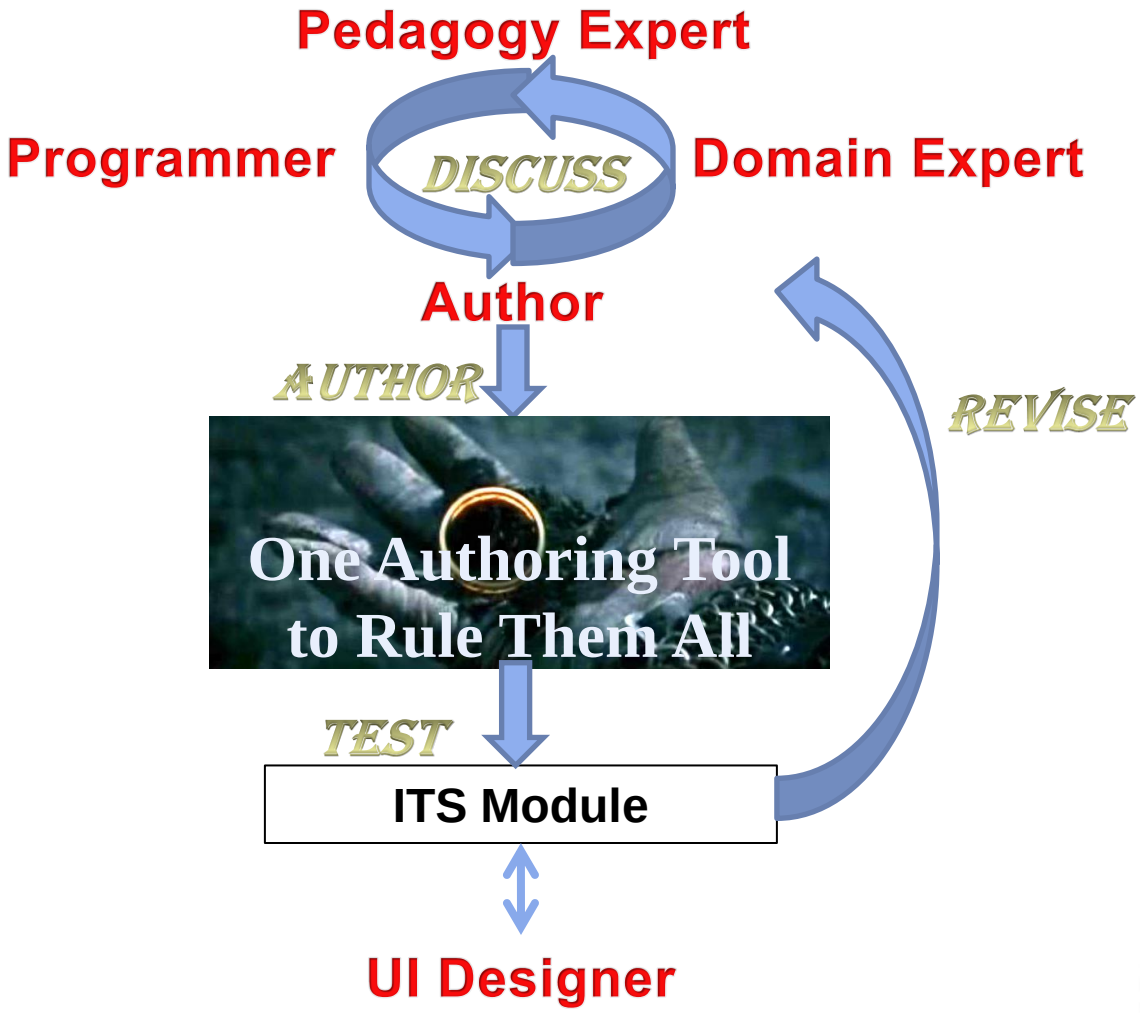


Theoretical Underpinnings

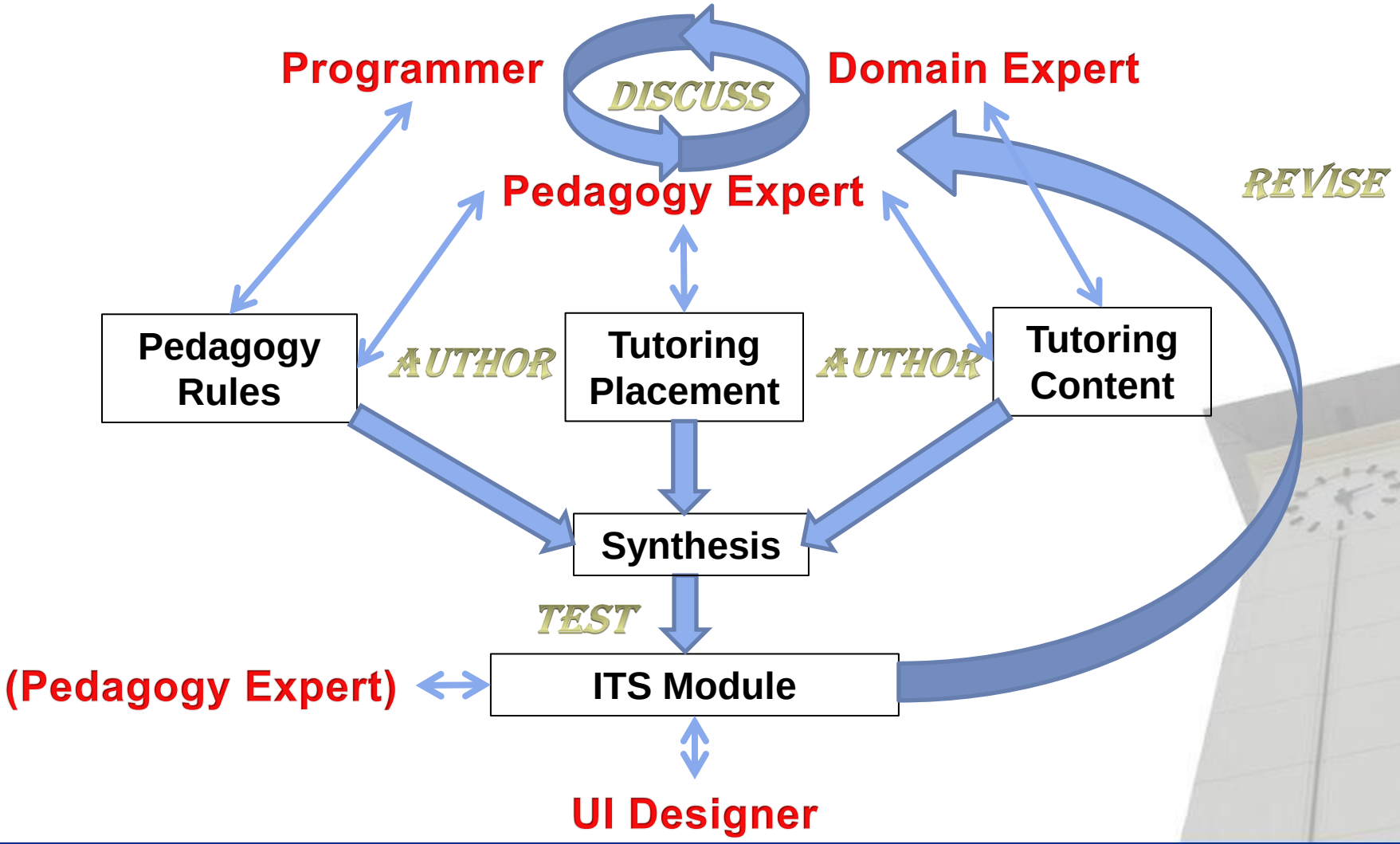
- Assembly-Line Division of Expertise
  - UI Design (Communication, Pedagogy)
  - Tutoring Content (Domain Knowledge, Pedagogy)
  - Tutoring Logic (Pedagogy)



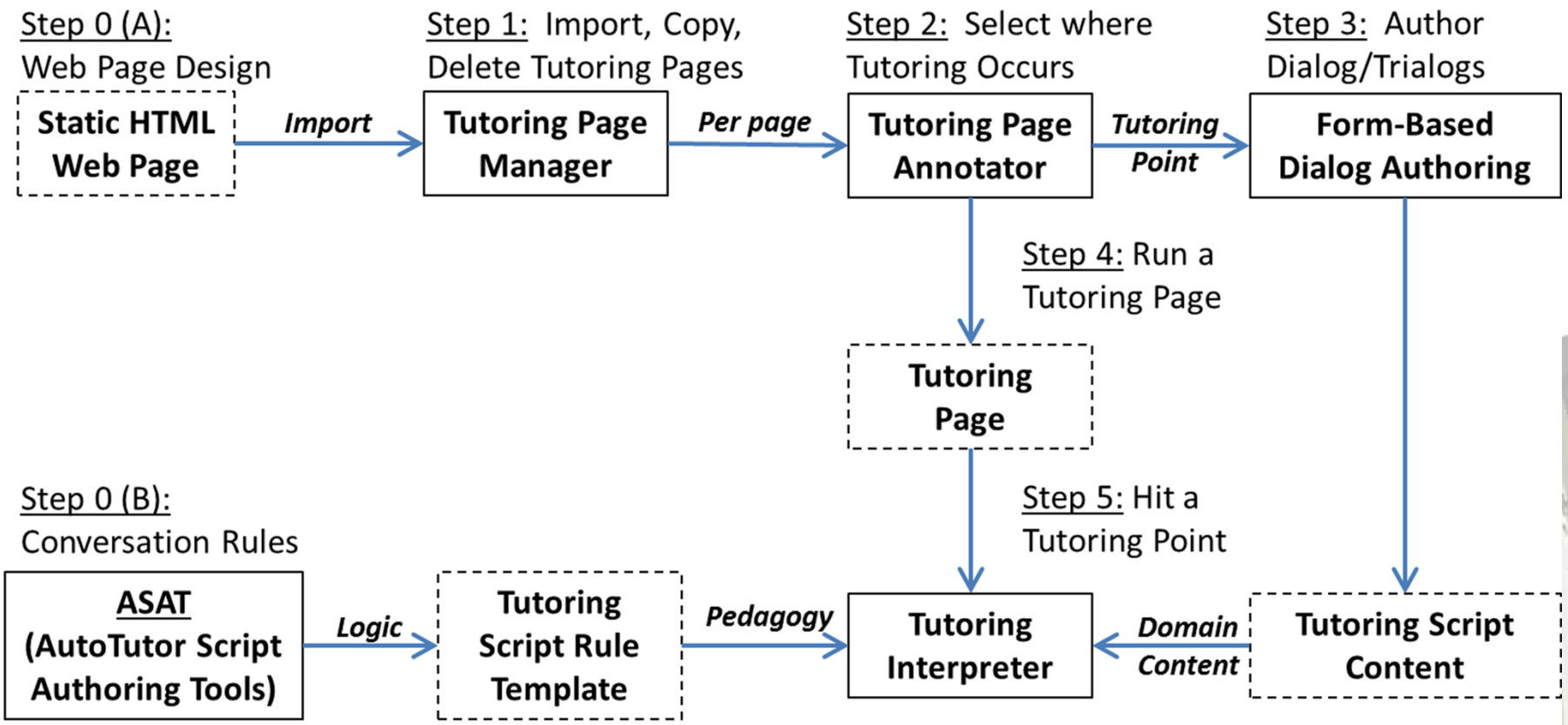
Workflow: Iterative Team Collaboration



Workflow: Suite-Based Collaboration



Tool Suite


Step 0 (A):
Web Page Design
Static HTML Web Page
Import
Step 1: Import, Copy, Delete Tutoring Pages
Tutoring Page Manager
Per page
Step 2: Select where Tutoring Occurs
Tutoring Page Annotator
Tutoring Point
Step 3: Author Dialog/Trialogs
Form-Based Dialog Authoring
Step 4: Run a Tutoring Page
Tutoring Page
Step 0 (B):
Conversation Rules
ASAT (AutoTutor Script Authoring Tools)
Logic
Tutoring Script Rule Template
Pedagogy
Step 5: Hit a Tutoring Point
Tutoring Interpreter
Domain Content
Tutoring Script Content

Tool Suite

- Tutoring Page Manager (Domain Expert)
- Tutoring Page Annotator (Domain Pedagogy Expert)
- Form-Based Authoring Tool (Domain Pedagogy Expert)
- ASAT(Pedagogy Expert w/some Programming Concepts)
- Python Scripts (Programmer w/NLP skills)
- HTML/CSS Editors (Programmer w/HCI skill)

Tutoring Page Manager

Name	Weblink	Template
Algebra 512	./location.html	layout.html

New | Copy | Remove | 3 Save | 4 Edit | View

Name: 2

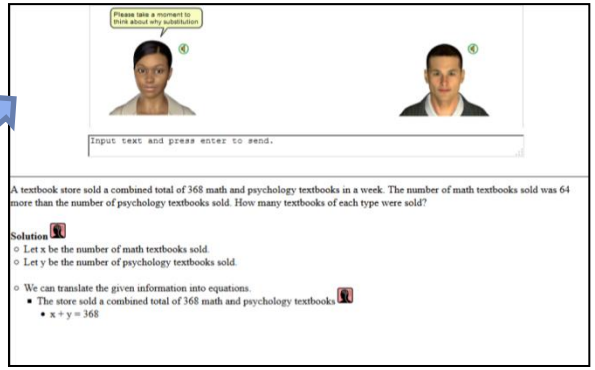
Weblink:

Template:

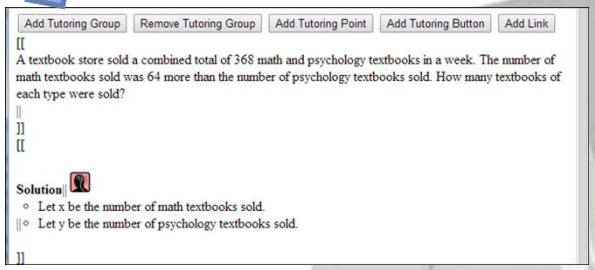
Content: www.somepage.com/ 1 Import

```
<ul style='padding:0;margin:0;padding-
left:20px'>
<li type=circle>Let x be the number of math
textbooks sold.</li>
<li type=circle>Let y be the number of
```

TEST

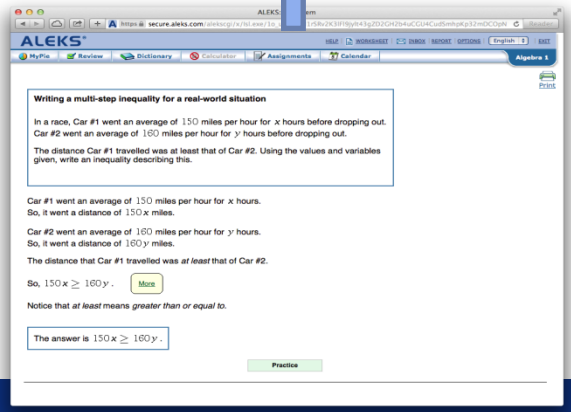


ANNOTATION



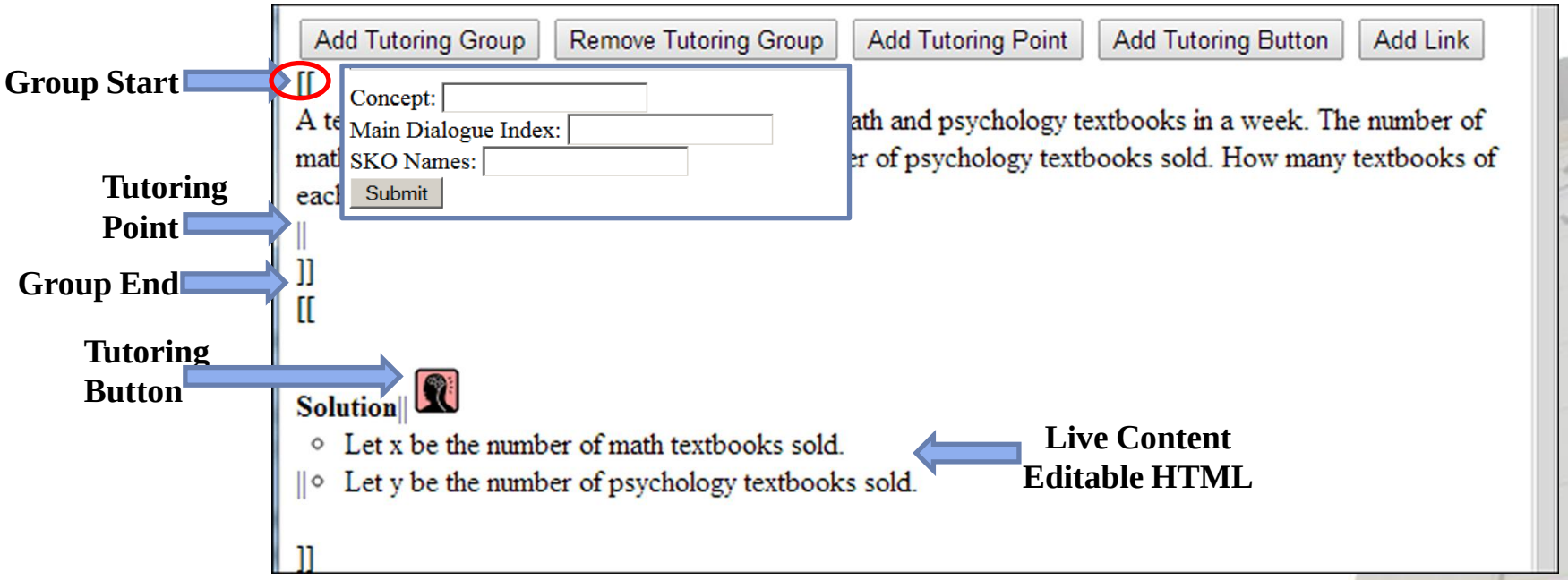
1. Import
2. Name
3. Save
4. Edit

www.somepage.com/math.html



Tutoring Page Annotator

- Author where tutoring happens in page
- Tutoring Group: One or more tutoring sessions
- Content-Editable: Delete/modify page text
- Sequencing: Rest of page hidden until group done



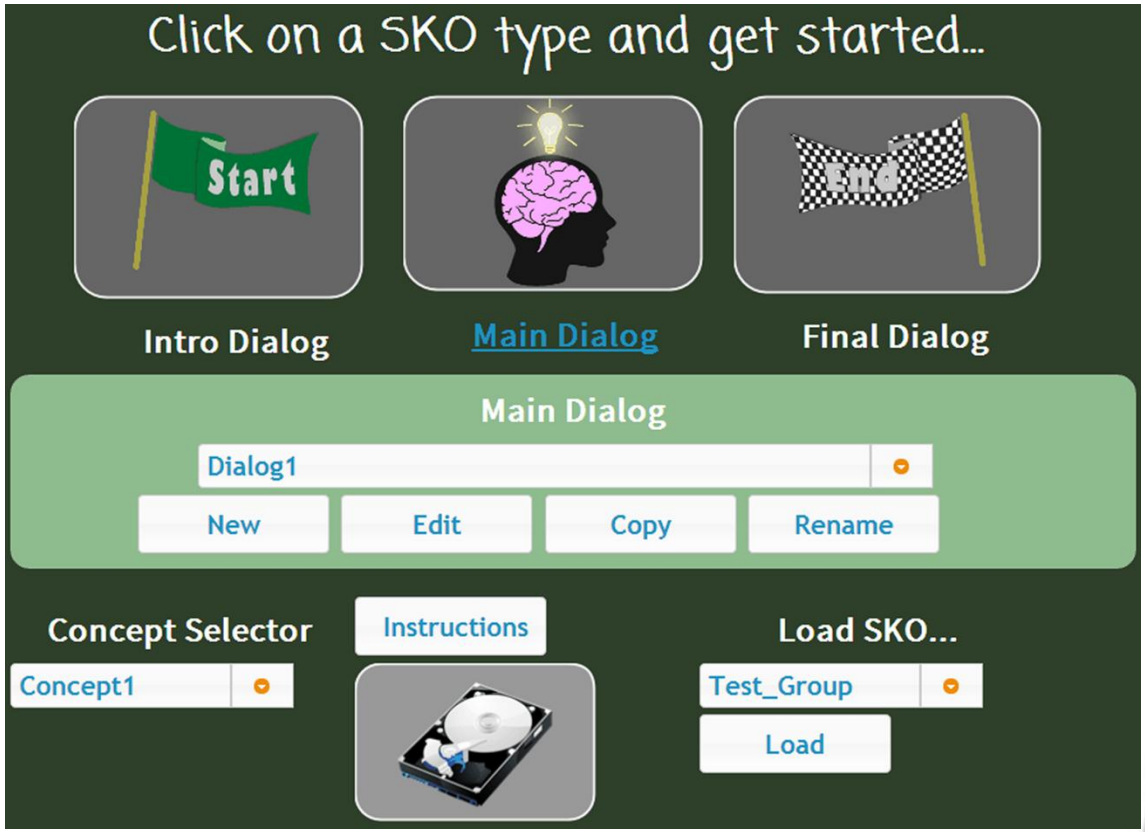
Tutoring Page Annotator

- Can import/export annotated HTML
- This HTML is combined with tutoring dialog by JavaScript inside “Tutoring Page” UI templates
- This is converted to JSON for storage and delivery

```
<body>
    <TutorGroupStart id="Reality Problem to Algebra " concept="RealProblemToAlg" SKO_names="['ALGE078_1','ALGE078_2']" maindialogindex="0"/>
        <span>A
        textbook store sold a combined total of 368 math and psychology
        textbooks in a week.</span>
        <span class='inline-block'>
        <span> </span>
        <span>The number
        of math textbooks sold was 64  more than the number of
        psychology textbooks sold</span>
        <span class='inline-block'>
    <TutorGroupEnd/>
    <TutorGroupStart id="Finding the Number of Solutions" concept="FindNumSolutions" SKO_names="['ALGE078_3']" maindialogindex="0"/>
        <span>. How many
        textbooks of each type were sold?</span>
        <span class='inline-block'>
    <TutorGroupEnd/>
  </p>
<p>
```

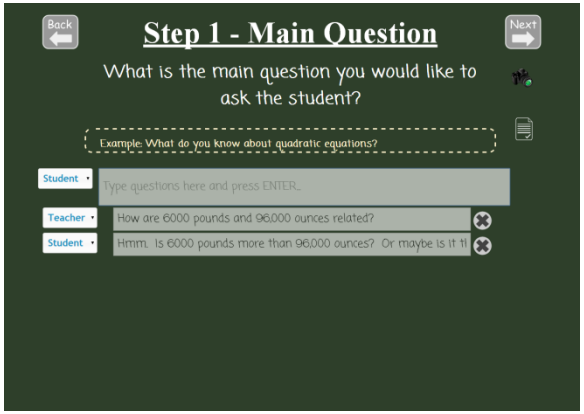
Form-Based Dialog Authoring Tool

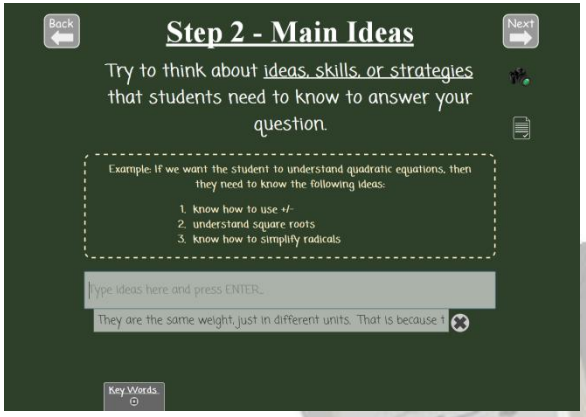
- Author content of tutoring trialogs (up to 3/group)
- Create new dialogs or select from existing ones

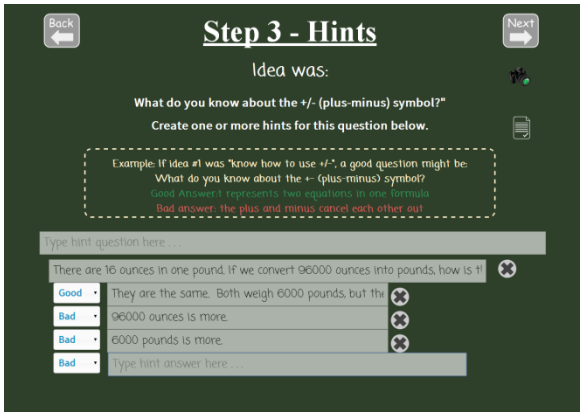


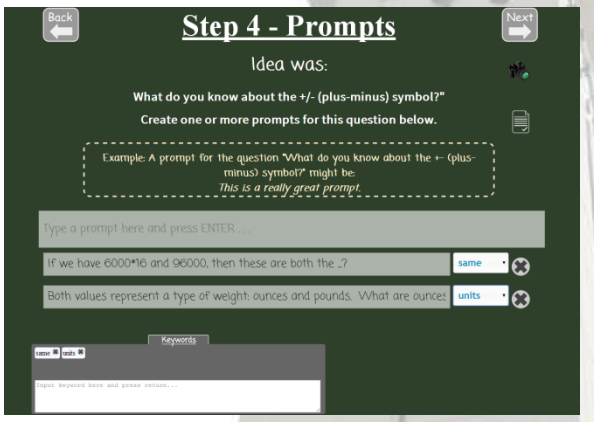
Form-Based Dialog Authoring Tool

- Author content one short form at a time
- Content is used for NLP and for agent speech
- Steps:
  1. Main questions
  2. Expectations
  3. Hints
  4. Prompts
  5. Misconceptions



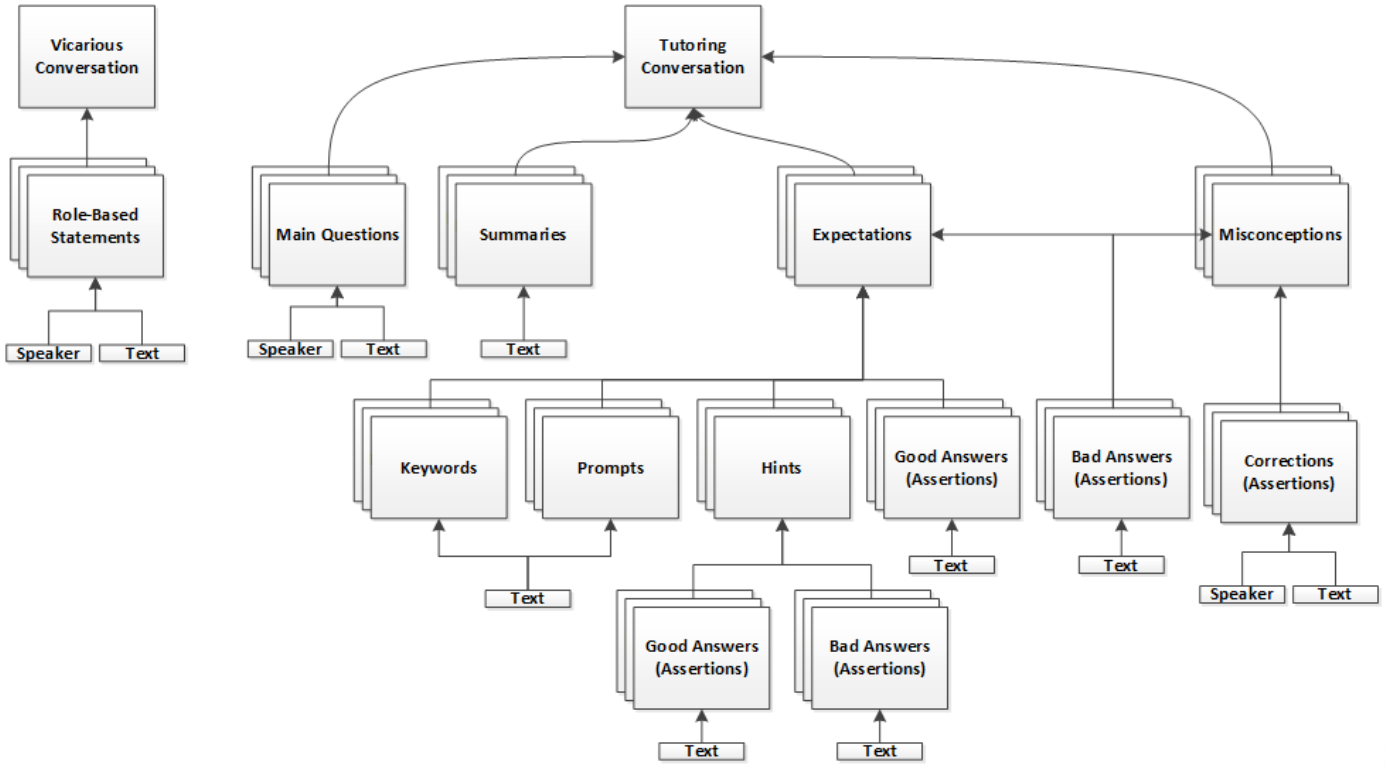






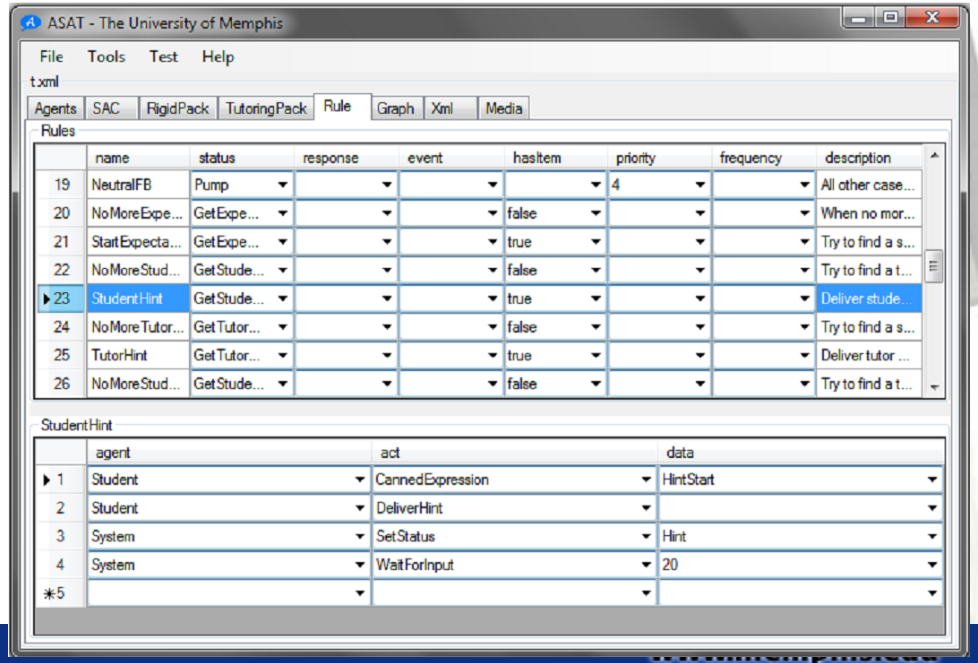
Form-Based Dialog Authoring Tool

- Content is authored in a minimal fashion
- Some elements **duplicated/pre-processed** before combining with an ASAT template



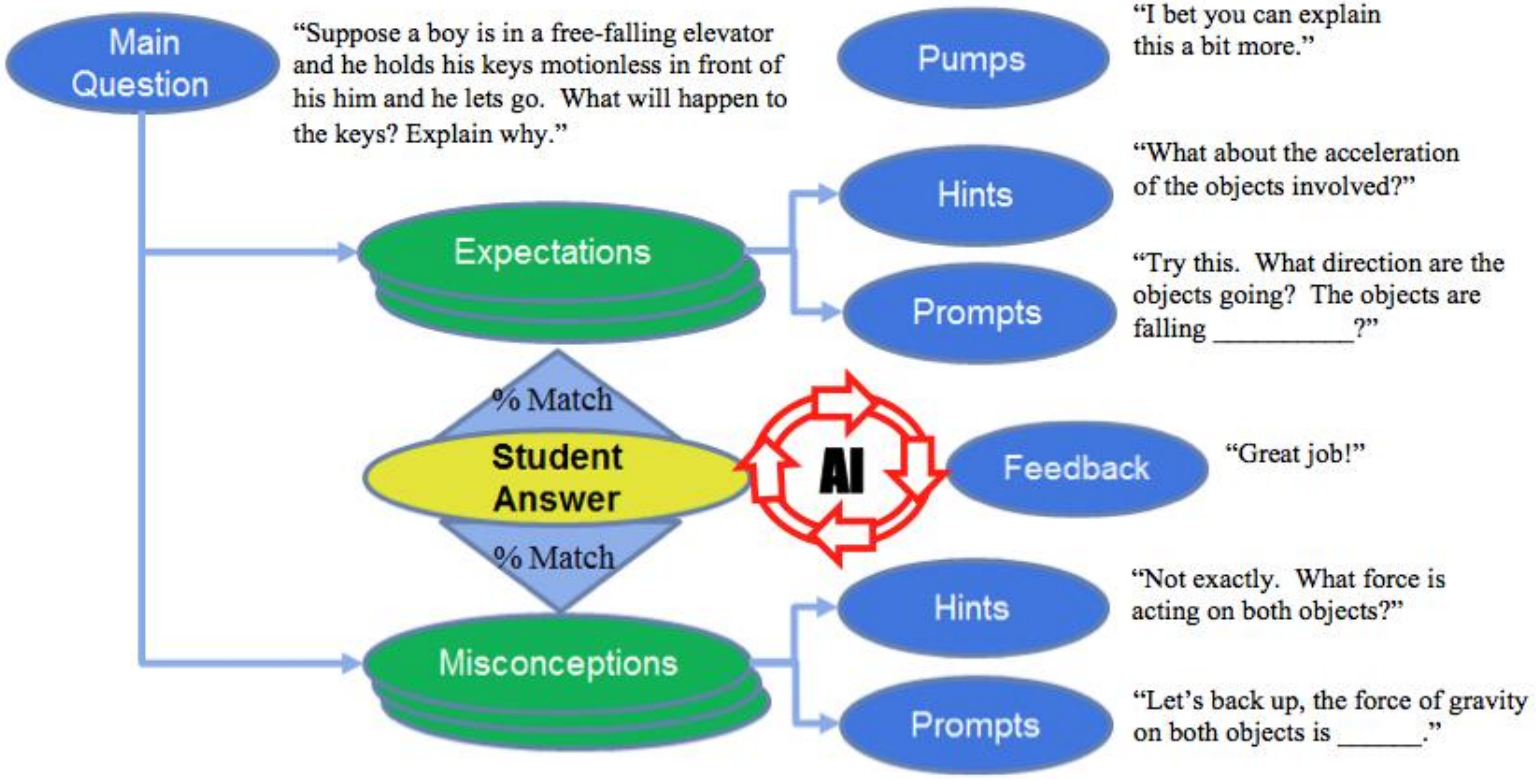
ASAT: Trialog Templates

- Form-based trialogs are content only, no logic
- Combined at run-time with ASAT rule templates
- Templates:
  - Vicarious: No human input, agents talk with each other
  - AutoTutor-style Expectation-Misconception Tutoring
  - Collaborative Lecturing (In Progress)
- Preprocessing Content
  - Depends on template type
  - Done with code scripts
  - Make speech-friendly text
  - Remove stopwords
  - Canonicalize keywords



ASAT: Trialog Templates

- AutoTutor-Style Expectation-Misconception Tailored (EMT) Dialog



Future Directions

- Suite online (internal use) starting this month
  - Form-based tool needs to be connected to cloud storage server
  - First use: Revising math tutoring dialogs
  - Bugfixes and usability checks will occur throughout the summer
- Enhancements on deck
  - Detect and flag semantically ill-formed content
  - Support authoring test inputs (good/bad responses)
  - Store rule templates in storage service(currently static)
  - New template for collaborative lecturing (e.g., linear lecture with checks for knowledge and attention)
  - Looking at better updating for collaborative editing: Multiple users on form-based tool could overwrite
  - This approach may work for interleaving w/video