Authoring Tools, Complexity, Epistemic Forms, and Cognitive Development:

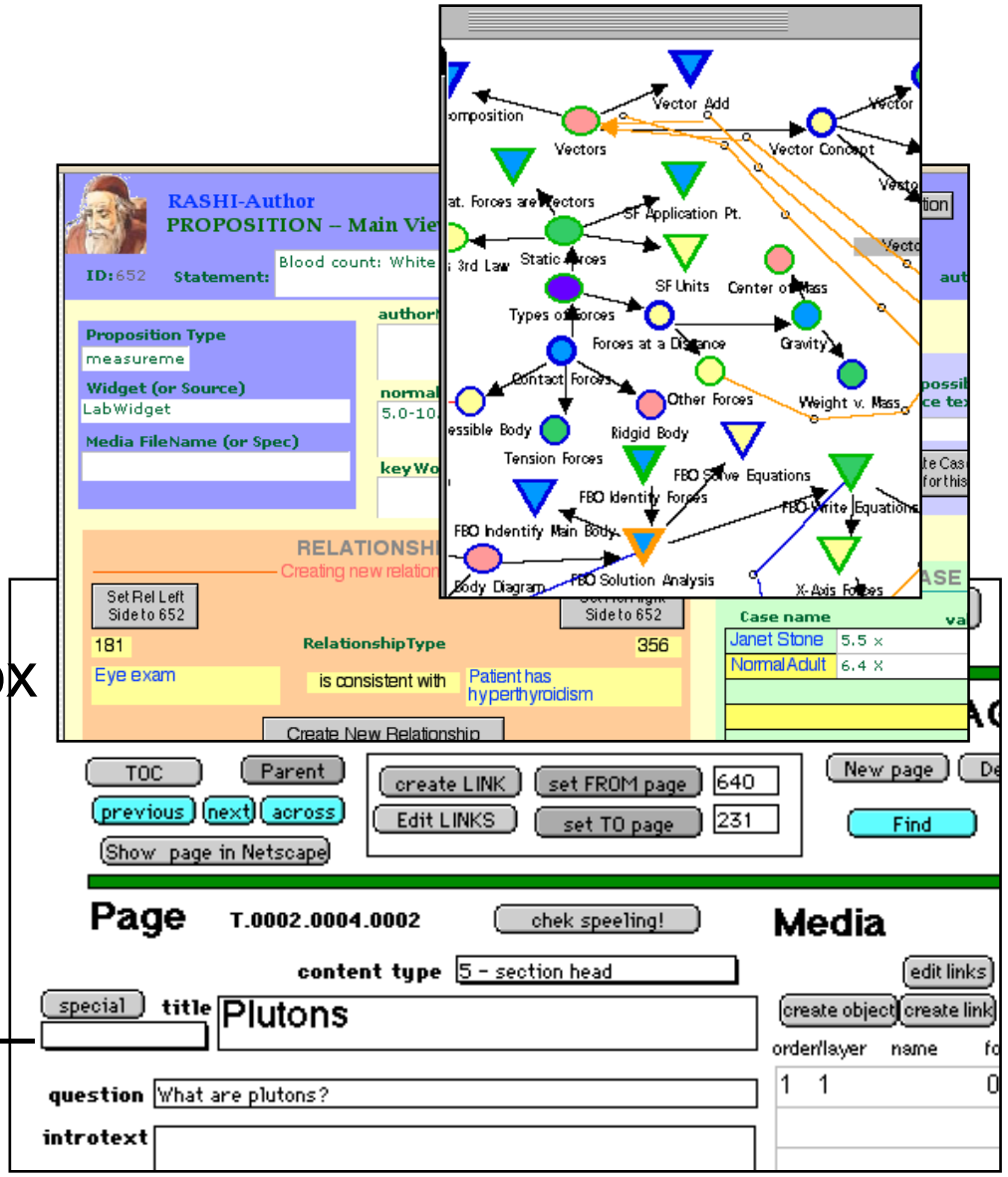
"What works for authors?"

Tom Murray
University of Massachusetts

ITS 2014 Workshop:
Intelligent Tutoring System Authoring Tools

Authoring Tool Projects

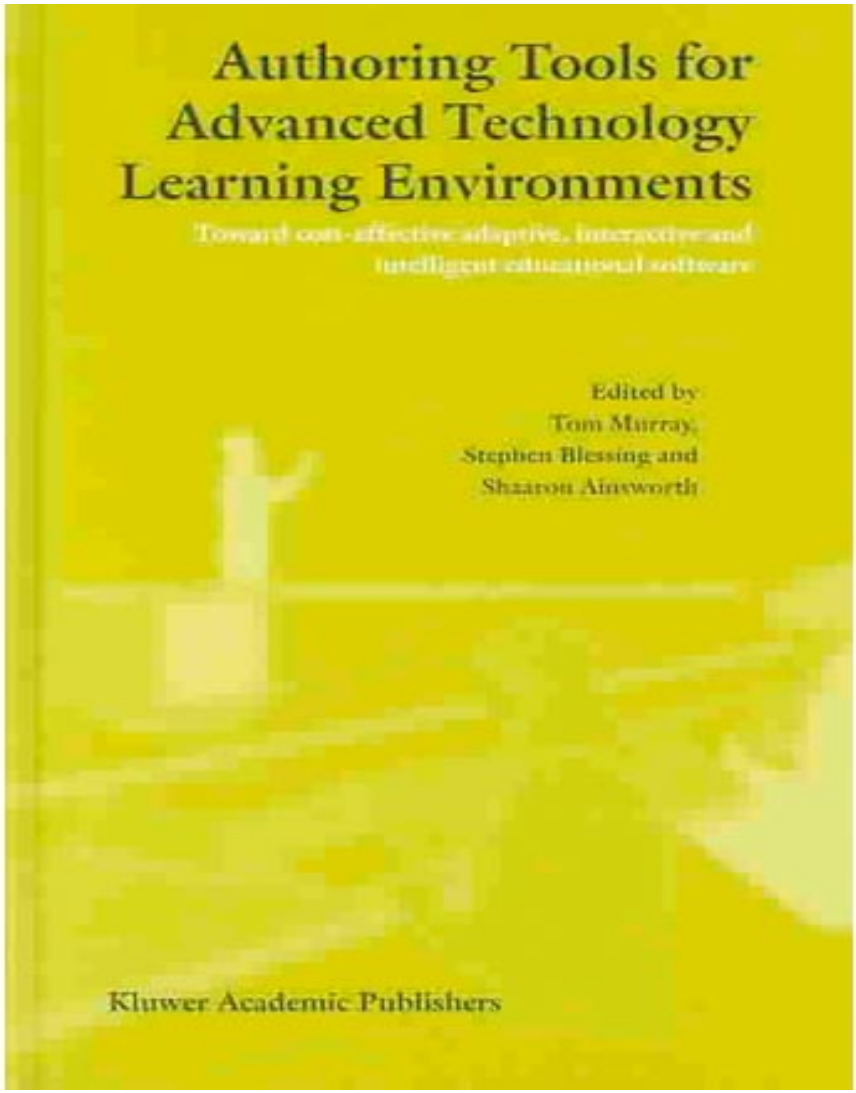
- SETS – (1991) Equipment maintenance training
- Eon - ITS authoring for domain, student, teaching models, and interface
- MetaLinks - hyperbook authoring tool
- SimForest-G - Glass box simulation authoring
- Rashi - Coached inquiry learning environment w/ authoring tool
- Wayang/MathSprings – (2014) teacher tools



Authoring Tools for Advanced Technology Learning Environments

Murray, Blessing, Ainsworth (Eds)

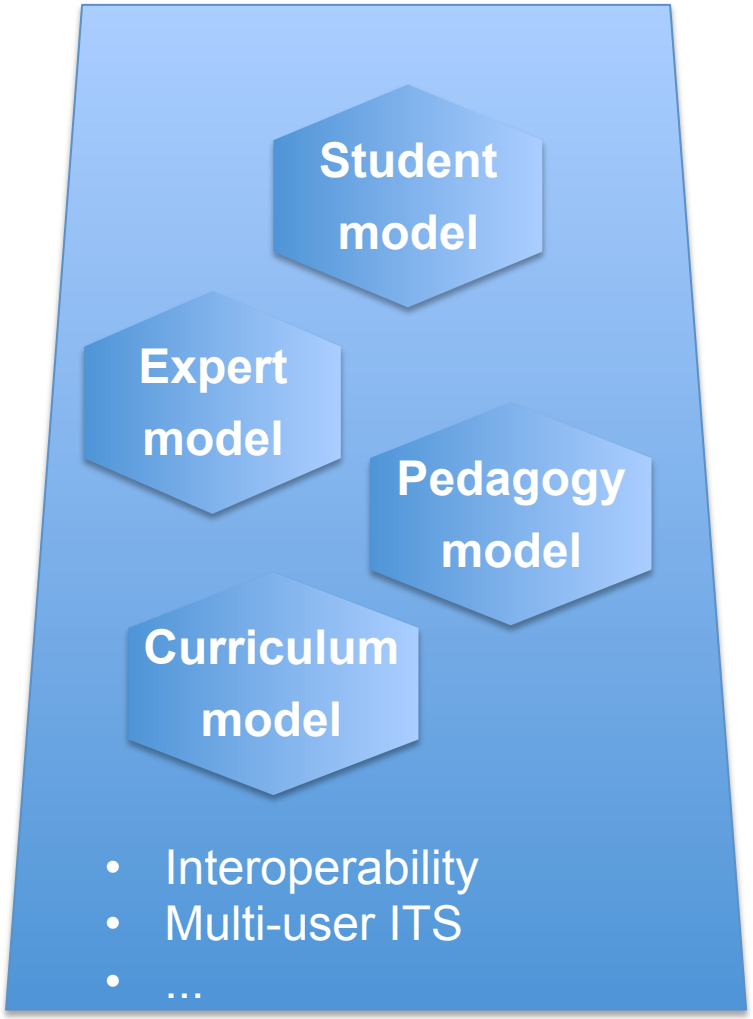
(2003)

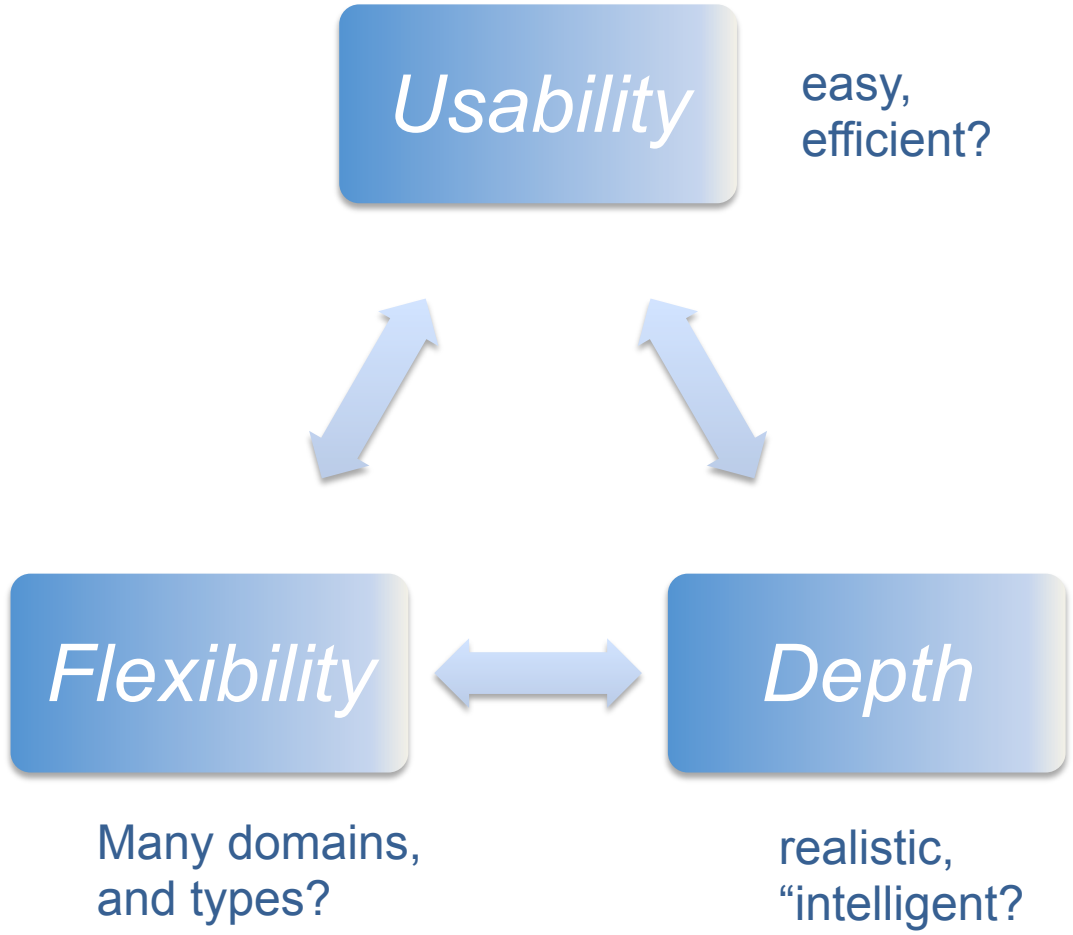


Recent interests applied to ITS authoring -- Theory

- User Roles vs Tools: Matching **Complexity**
- **Activity Theory**: Tools, Tasks, Users, Community
- **Epistemic Forms**/Games
- **Developmental** Theory (of complexity)

ITS Authoring Tool Design Tradeoffs





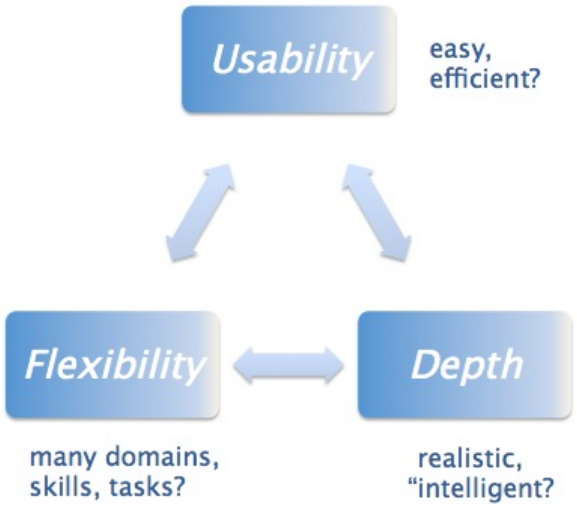
Authoring Tool Users

Roles (tool use roles)	**Benefits** (of that role)	**Problems** (of that role)
Teachers PRACTICAL	Practical experience	Not good at articulating or abstracting expertise
Domain Experts & Content Developers PARTIAL	Auth. tool infers the instructional methods	A fixed instructional method
Instructional Designers & Learning Theorists THEORETICAL	Know learning theories & research	Rare; not trained in knowledge engineering
Knowledge Engineers and ITS Developers EXPERIENCED	Know the tools; Are sometimes also plugged into user testing	May not know what it is like to teach or learn the material
Computer scientists & Software developers (ACTUAL?!)	Complexity capacity. Don't have to build to a real user base.	"its intuitively obvious to the casual observer…"

Matching Complexity— Tools vs. User capacity

Tools

Users

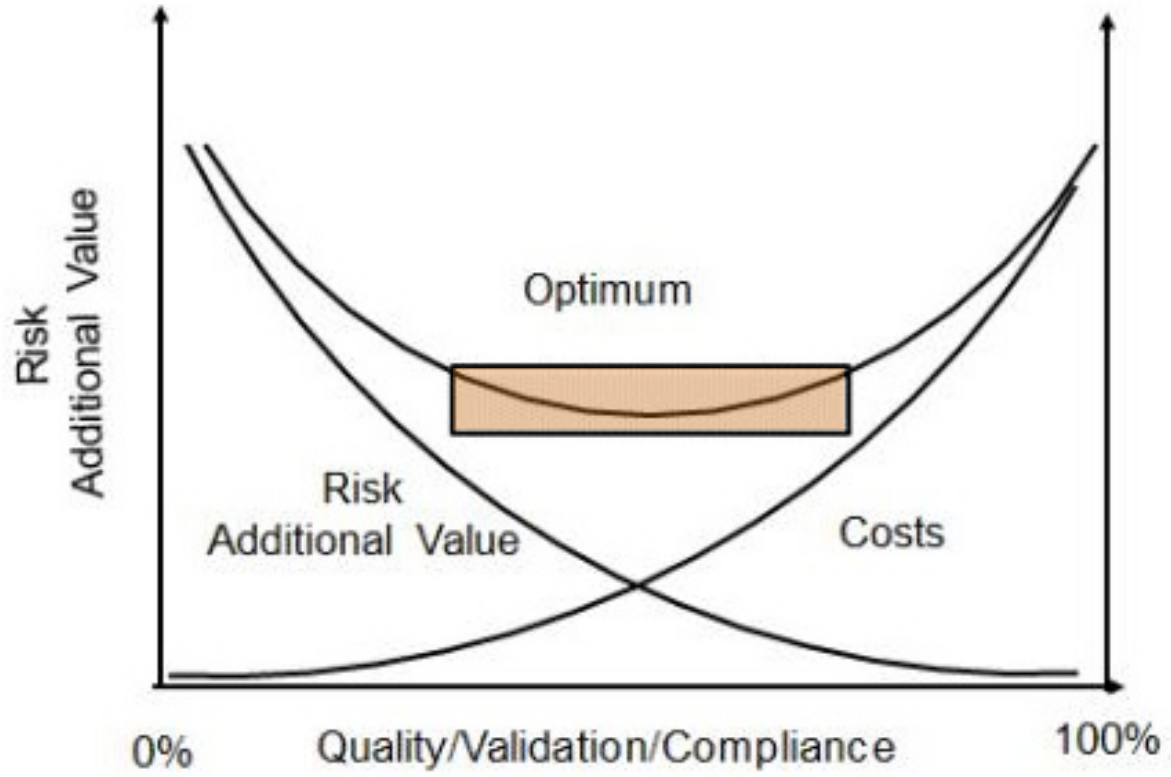


Complexity?

Roles (tool use roles)	Benefits (of that role)	Problems (of that role)
Teachers PRACTICAL	Practical experience	Not good at articulating or abstracting expertise
Domain Experts & Content Developers PARTIAL	Auth. tool infers the instructional methods	A fixed instructional method
Instructional Designers & Learning Theorists THEORETICAL	Know learning theories & research	Rare; not trained in knowledge engineering
Knowledge Engineers and ITS Developers EXPERIENCED	Know the tools; Are sometimes also plugged into user testing	May not know what it is like to teach or learn the material
Computer scientists & Software developers (ACTUAL?!)	Complexity capacity. Don't have to build to a real user base.	"its intuitively obvious to the casual observer…"

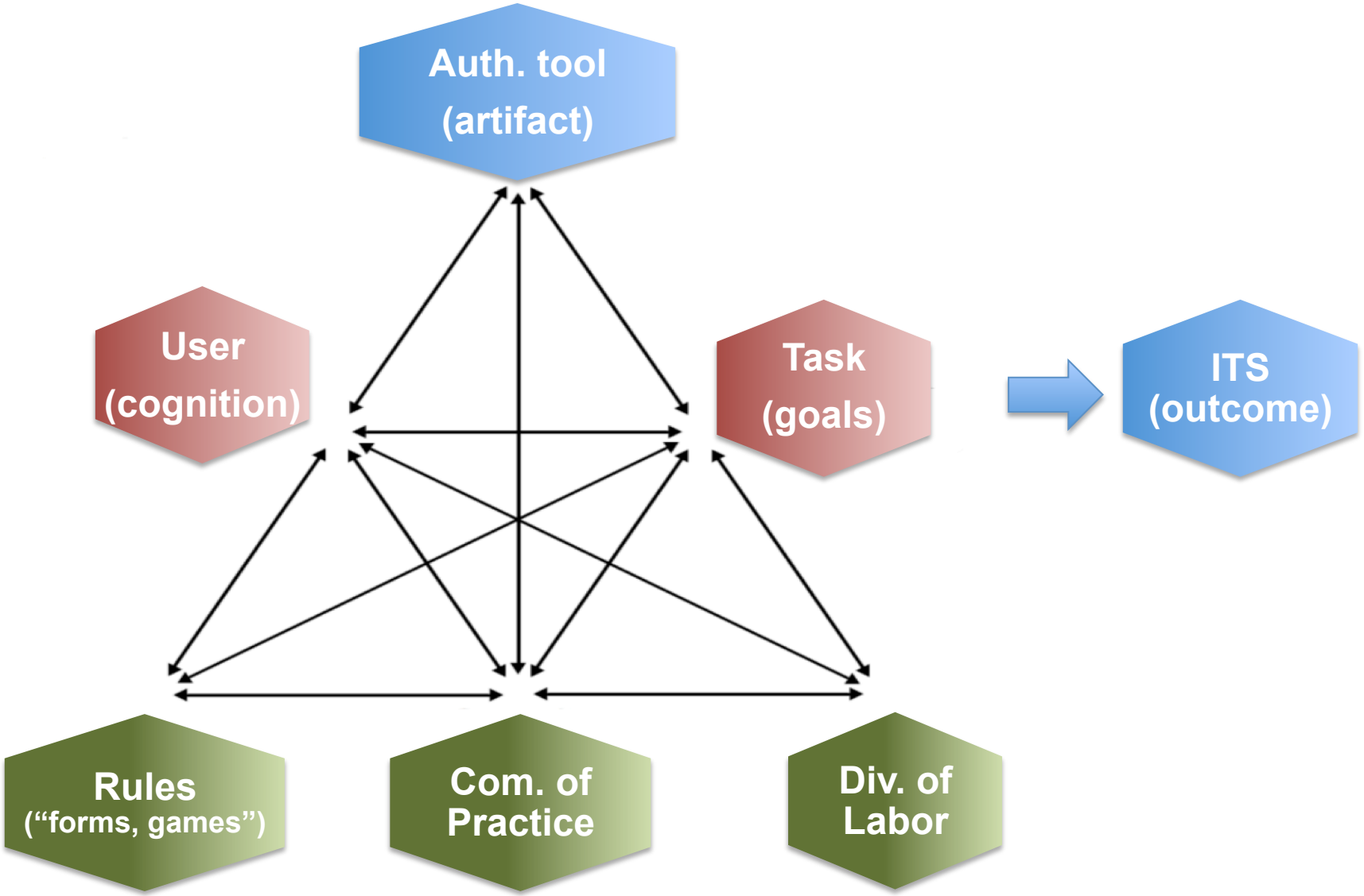
1) Know your user (anticipate user needs)
2) Usability Testing ("early and often")
3) Theory (of usability, cognition...)

Complex Software: (investment) Risk Assessment



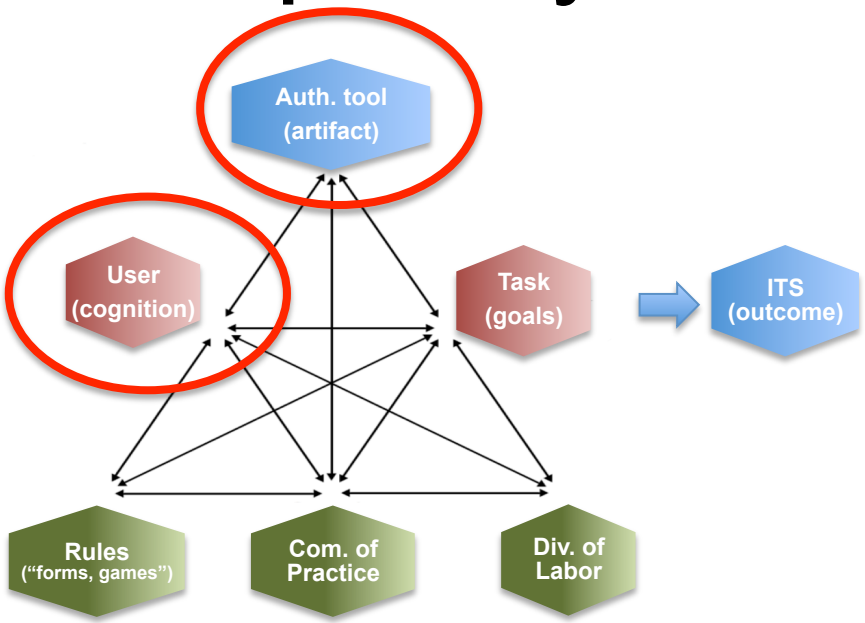
ITS design == AI-complete…….ITS A-Tool design == ITS-Complete!

Activity Theory



Jonassen & Rohrer-Murphy 1999; Engestrom et al. 1999;

Complexity Coordination



- *Cognitive Complexity (user)*
- *Task Complexity*
- *Tool Complexity*
- *Socio-cognitive complexity (COP)*

Complexity matching:

1. *User <> Tool*
2. *Task <> User*
3. *COP (& user) <> Tool*

Authoring Tools for All Users?

(Tiered Authoring/Work Flow)

Roles (tool use roles)	**Benefits** (of that role)	**Problems** (of that role)	**Complexity Design Capacity**
Teachers PRACTICAL	Practical experience	Not good at articulating or abstracting expertise	LOW
Domain Experts & Content Developers PARTIAL	Auth. tool infers the instructional methods	A fixed instructional method	MED
Instructional Designers & Learning Theorists THEORETICAL	Know learning theories & research	Rare; not trained in knowledge engineering	MED
Knowledge Engineers and ITS Developers EXPERIENCED	Know the tools; Are sometimes also plugged into user testing	May not know what it is like to teach or learn the material	MED-HIGH
Computer scientists & Software developers (ACTUAL?!)	Complexity capacity. Don't have to build to a real user base.	"its intuitively obvious to the casual observer…"	HIGH

Capacity is context-dependent

- **User complexity** capacity: f(S,I,T)

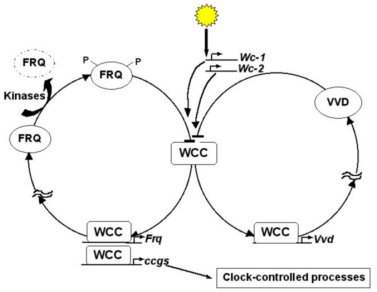
  Background **skill** (generic—see table)

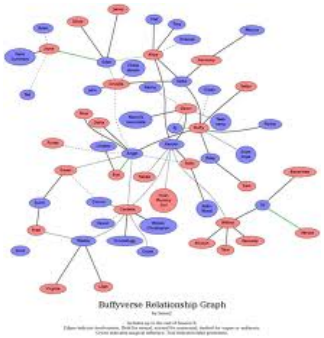
  + Investment in **training** A-Tool

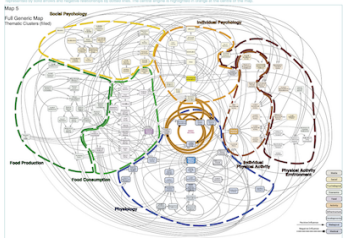
  + **Time** available to author this ITS

Sources of (software) system complexity

- Structural complexity (space)
  - Object has many properties
  - Many parts
  - Many types of parts
  - Many relationships
  - Many types of relationships
- Perspectival complexity
  - alternatives, hypotheticals, variables, decision spaces
- Dynamic complexity
  - Loops, Feedback, recursion relationships (“non-linearity”)





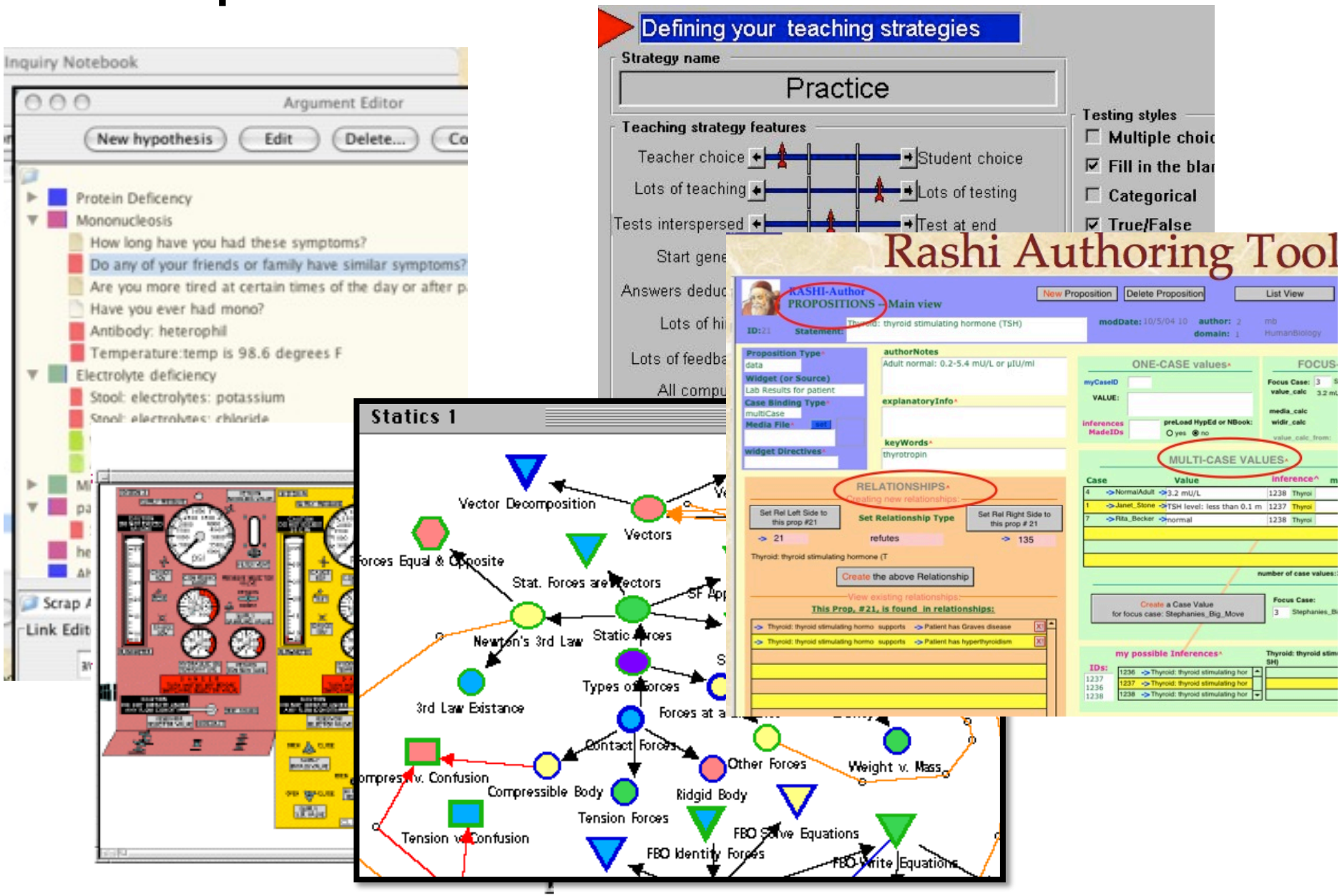


Epistemic Forms & Games (Mental Models)

(Collins & Furgeson, 1993)

- list
- matrix or table
- molecular model
- periodic table
- web page menu
- x-y graph
- pert chart
- binary tree
- floor plan
- street map
- org. chart
- musical score
- timeline
- cause/effect diagram
- network
- relational database
- sentence diagram
- term paper outline

Epistemic forms in interfaces



Can we estimate the complexity of epistemic forms/games in authoring tools?

- >> use cognitive developmental theory

Cognitive Developmentalists

Kegan; Fisher; Commons; Cook-Greuter...

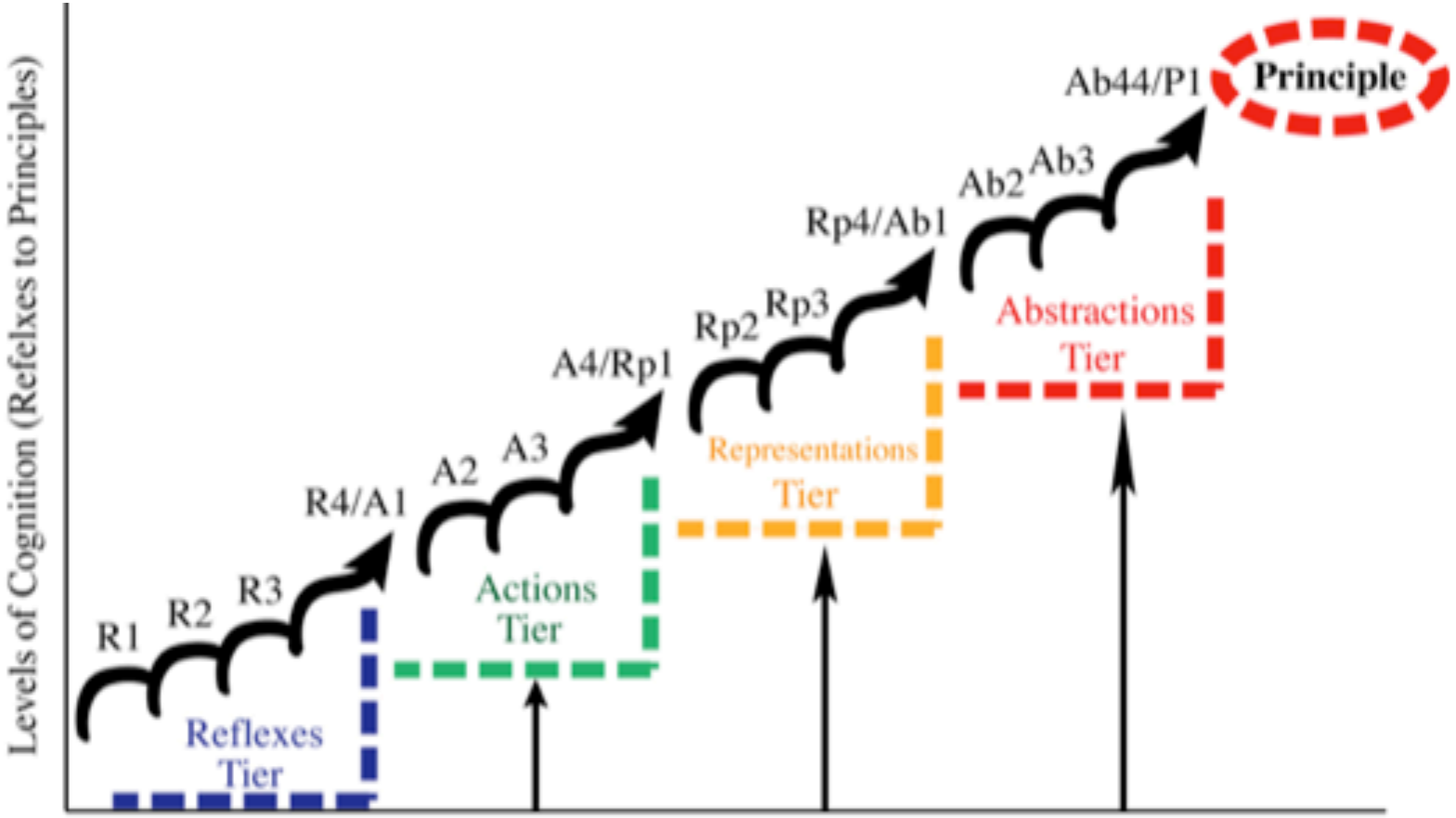
- Human development and learning can be described in terms of "**qualitative differences in mental complexity**."
- …that add a **hierarchical "structural perspective** in analyzing the organization of actions and thought."

Development: the concept of Fun

Single Rep. (unconnected list)	Fun is swinging on a swing. It's sliding on a slide.
Rep. Mapping (connections)	Fun is when Tommy and I put blocks together and then knock them down so that they make a loud noise that makes us laugh.
Rep. System (interconnections)	Fun is different things. Sometimes I like to climb… that makes me…
Single Abstr. (unconnected list)	Fun is a way of enjoying yourself. It is a form of pleasure.
Abst. Mapping (connections)	There are a variety of ways that a person can have fun. Some people enjoy physical activities, like sports or just exercise. Some people…

- Actions at a higher order of hierarchical complexity **organize and transform** the lower order actions
- Complexity level (or “order”) based on:
  1. complexity of **objects** operated upon (**vertical** complexity; order of abstraction)
  2. complexity of object **coordination** (**horizontal** complexity; structure of objects)

Fisher's Skill Theory



For any skill

algebra, reading, piano, parenting, tennis…

Addition > Multiplication > Algebra > Calculus > …

- **Single Set (e.g. a list)**

  (e.g. addition; subtraction…)

- **Mapping (e.g. linear causal link)**

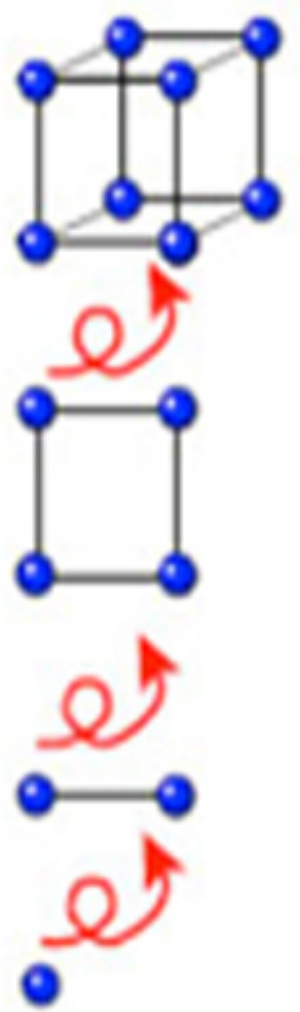
  (coordinating addition & subtraction)

- **System (e.g. many interconnected parts)**

  (e.g. coordinating +, -, x, /)

- **[System of Systems (an entire complex system)**

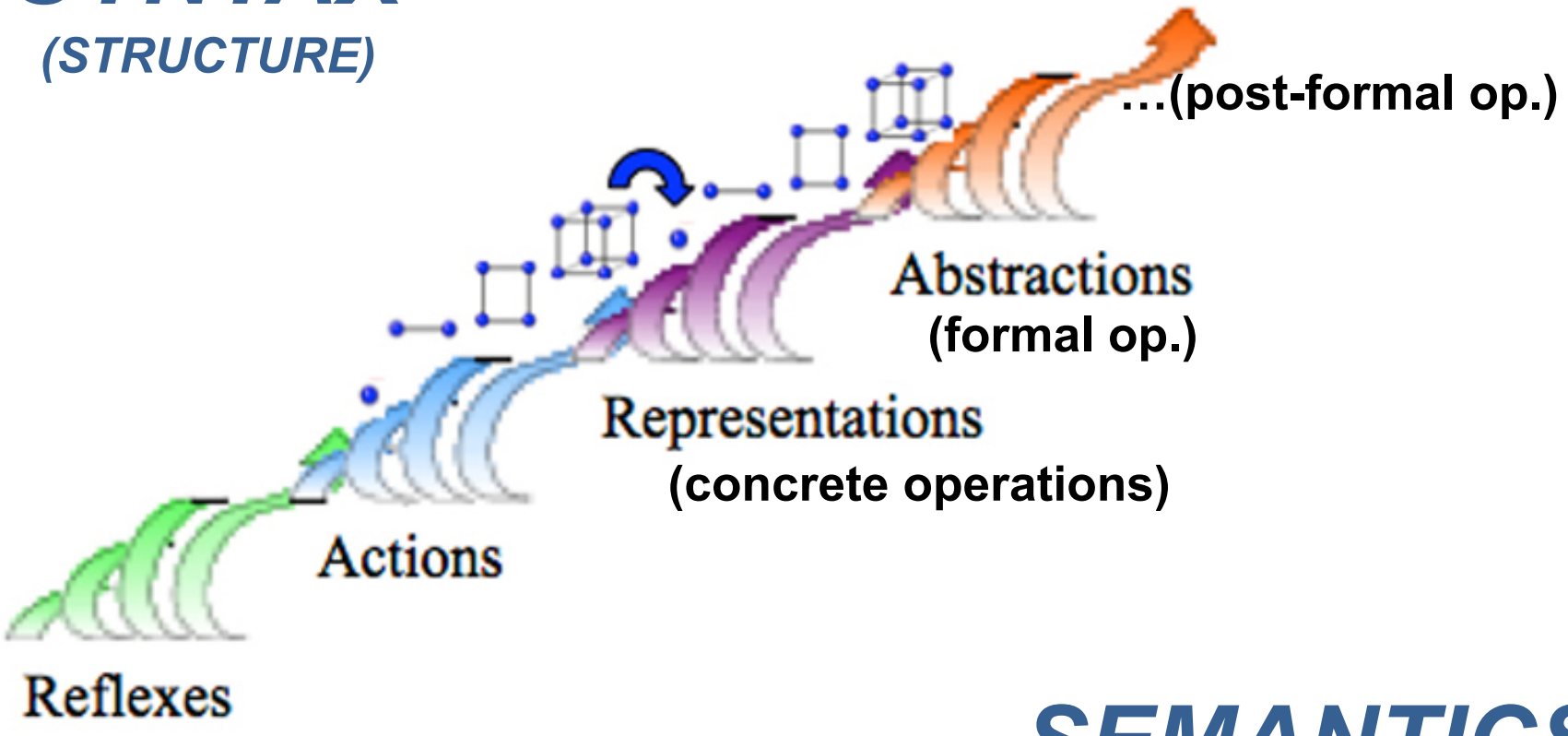
  (moving to Algebra)]



- **System of Systems** (an entire complex system with feedback loops)
- **System** (e.g. many interconnected parts)
- **Mapping** (e.g. linear causal link)
- **Single Set** (e.g. a list)

SYNTAX

(STRUCTURE)



SEMANTICS

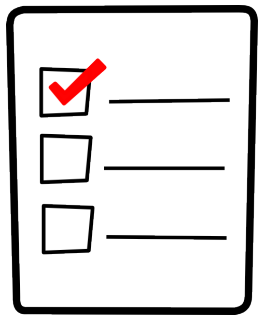
(OBJECTS)

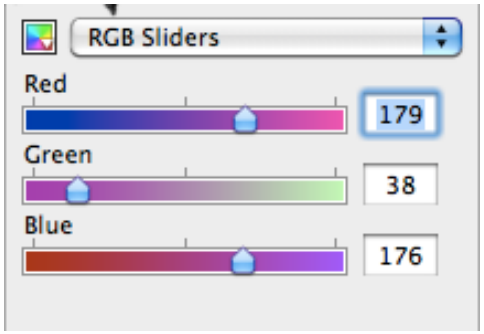
Increasing complexity of the Mental Model

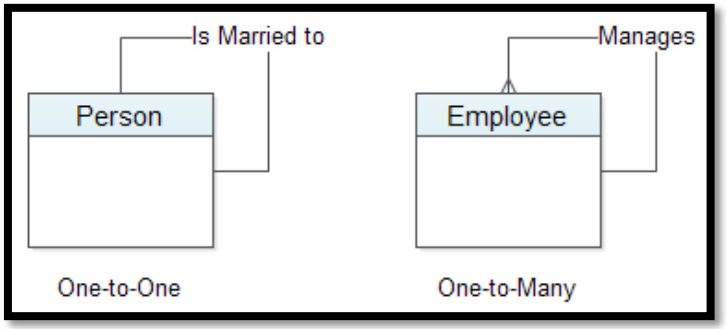
Complexity Level	Epistemic Form
1. Simple objects/info	Text information fill-in boxes Lists, choices, sliders, and check boxes
2. Mappings & Abstractions	Tables and matrices Hierarchies and trees Simple scripts, Forms, schemas, or templates
3. Formal Systems	Procedures with branches Variables/Equations and Boolean logic Structural models: concept networks, boxology diagrams
4. Dynamic Systems	Causal and constraint models, Decision Trees Behavioral/procedural models: If/then and rule-based procedural representations Complex interactions
5. Architectures & Ecosystems	Systems of systems, models, or rule-sets

Epistemic Forms Complexity

1. **Simple objects**: lists, sliders, simple relationships

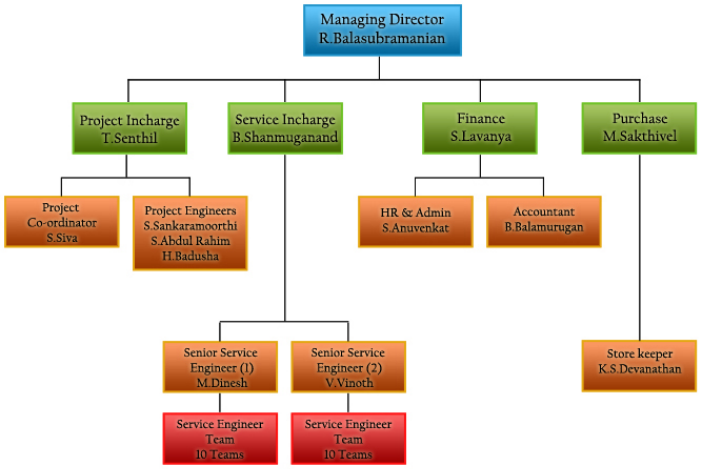


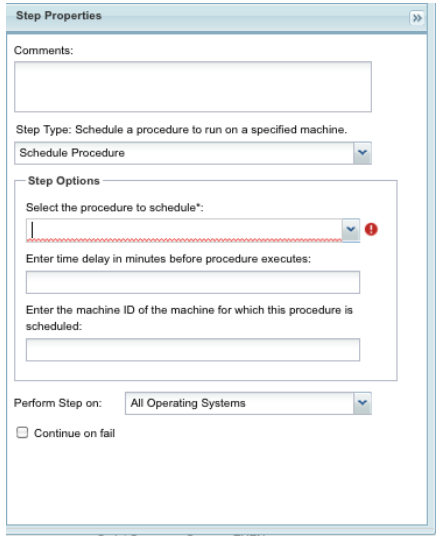




2. **Complex mappings**: tables, trees, scripts, concept nets

	A	B	C	D
1	Date	Appl	Orang	Total Fr
2	6/1/2012	125	75	200
3	6/2/2012	118	84	202
4	6/3/2012	164	72	236
5	6/4/2012	114	65	179
6	6/5/2012	98	96	194
7	6/6/2012	172	82	254
8	6/7/2012	122	82	204
9	6/8/2012	143	91	234
10	6/9/2012	137	87	224

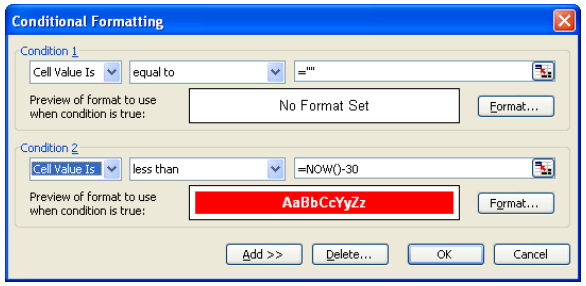


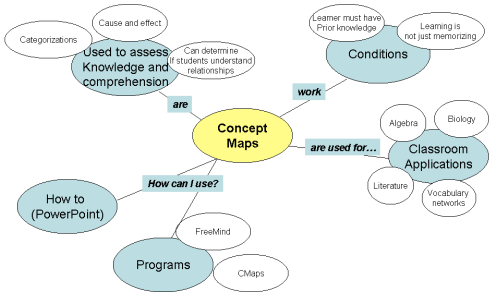


(cont.)

3. **Formal systems**: Add variables, equations, static models

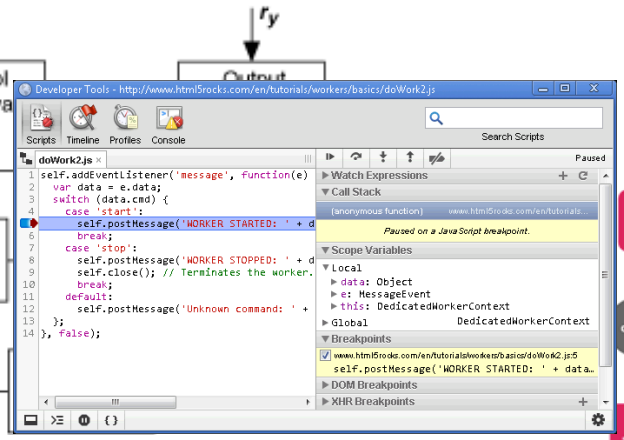
$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

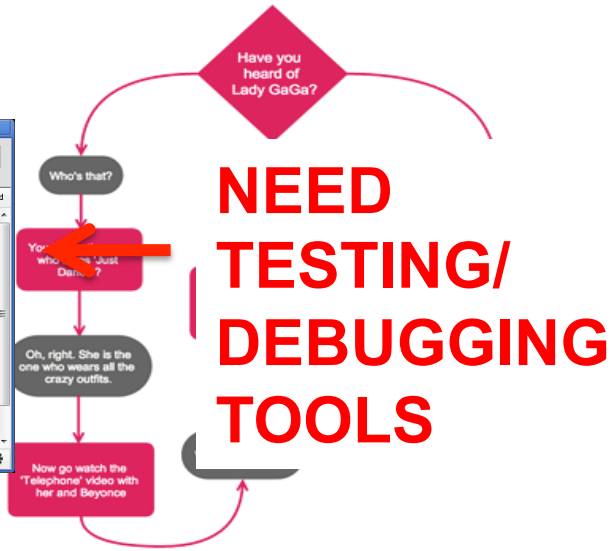




4. **Dynamic Systems**:
Loops, conditionals, dynamic/constrain models, rule systems

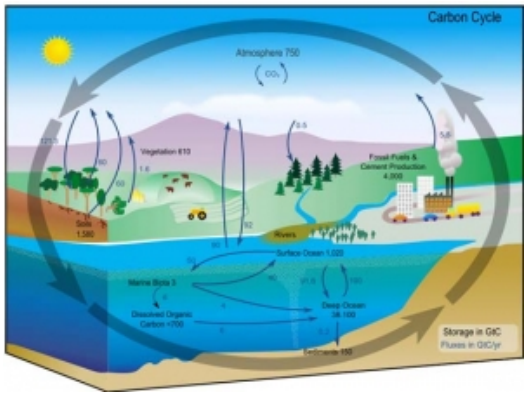
```
/**
 * Simple HelloButton() method.
 * @version 1.0
 * @author john doe <doe.j@example.com>
 */
HelloButton()
{
  JButton hello = new JButton( "Hello, wor
  hello.addActionListener( new HelloBtnList

  // use the JFrame type until support for t
  // new component is finished
  JFrame frame = new JFrame( "Hello Button"
  Container pane = frame.getContentPane();
  pane.add( hello );
  frame.pack();
  frame.show();              // display the fra
}
```

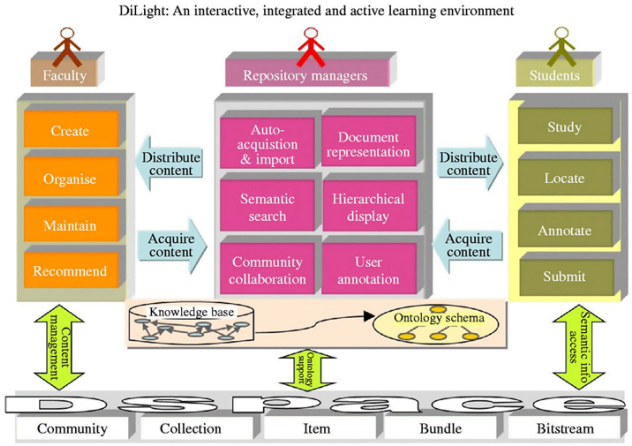


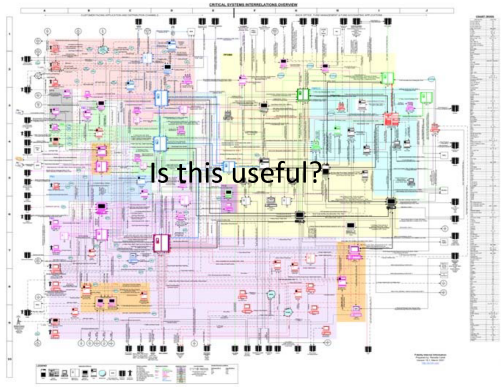


(cont.)

5. **Dynamic Systems/ Architectures**
(version control; constant monitoring)

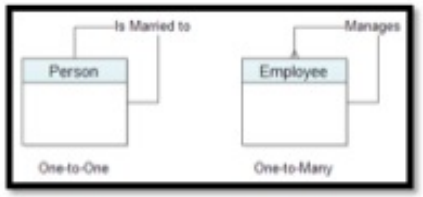


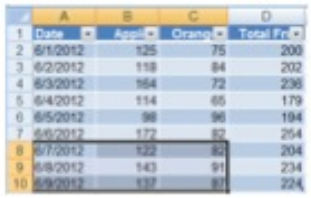


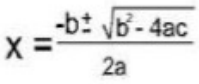


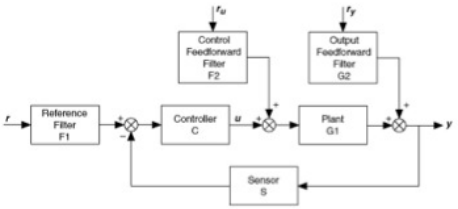
Increasing complexity of the Mental Model

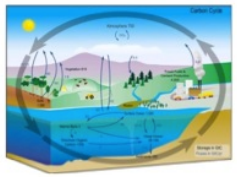
Complexity Level	Mental Model Characteristics
1. Simple objects	Facts, isolated info-bits
2. Mappings & Abstractions	Many relatinships, fairly linear, predicatable
3. Formal Systems	Abstractions, variables, decisions
4. Dynamic Systems	Non-linearities, sub-systems, decision trees
5. Architectures & Ecosystems	Complex interactions, whole-systems, evolving, unpredicatable

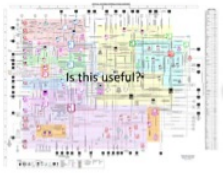












Complexity Design Capacity
LOW
MED
MED
MED-HIGH
HIGH

Review

- User Roles vs Tools: Matching **Complexity**
- **Activity Theory**: Tools, Tasks, Users, Community
- **Epistemic Forms**/Games
- **Developmental** Theory (of complexity)

Conclusions:
If “we build it will they come”?

- **Market & Buy-in**: Have ITSs demonstrated cost-benefit yet? In what situations?
- Creating a **pipeline for training** and trained ITS authors and knowledge engineers
- Building **communities of practice** (examples: CTAT, WISE, Knowledge Forum…)
- Expectation management: **matching tool and user complexity** (and constraining the breadth/ depth of the outcome ITSs)
- Building an ITS authoring tool is like…?? What known completed project? Using lessons from a **parallel domain**?

Thank You