Rapid Model Building using Diagrams and Examples

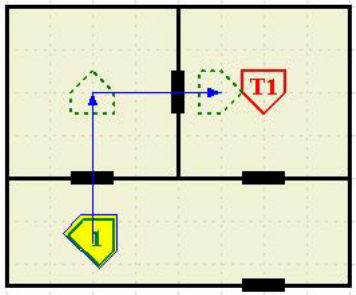
Professor John Laird
University of Michigan
laird@umich.edu

Douglas Pearson, Ph.D.
ThreePenny Software
doug@threepenny.net

Supported by Office of Naval Research
Start Date: January 1, 2003

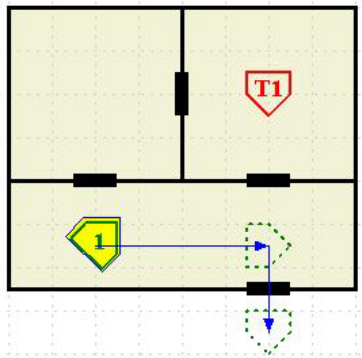
Why Building Models is Hard

Desired Behavior



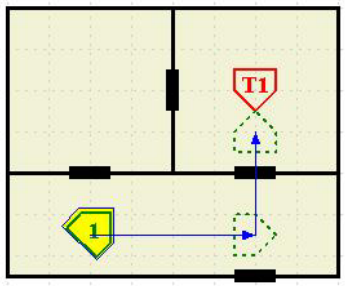
Errors and unvalidated behavior

Actual Behavior



Rules produce the behavior

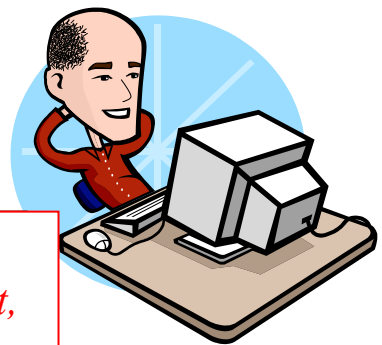
Intended Behavior



SME

Slow, Difficult, Error prone

Slow, Difficult, Error prone



KE

Slow, Difficult, Error prone

A -> B
C -> D
E, J -> F
G, A, C -> H
E, G -> I
J, K -> L

Rules

ThreePenny SOFTWARE

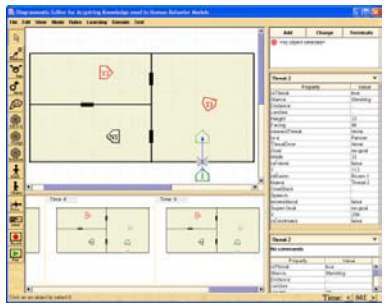
Diagram-based Example-driven Development Tool

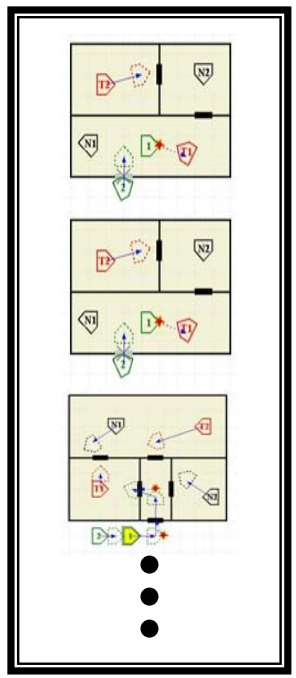
Define behavior with diagram-based examples

Library of validated behavior examples

Analysis & generation tools

Executable Code





Detect inconsistency

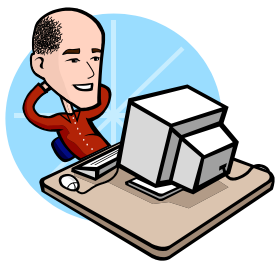
Generalize

Generate rules

Simulate execution

A -> B
C -> D
E, J -> F
G, A, C -> H
E, G -> I
J, K -> L

SME



KE

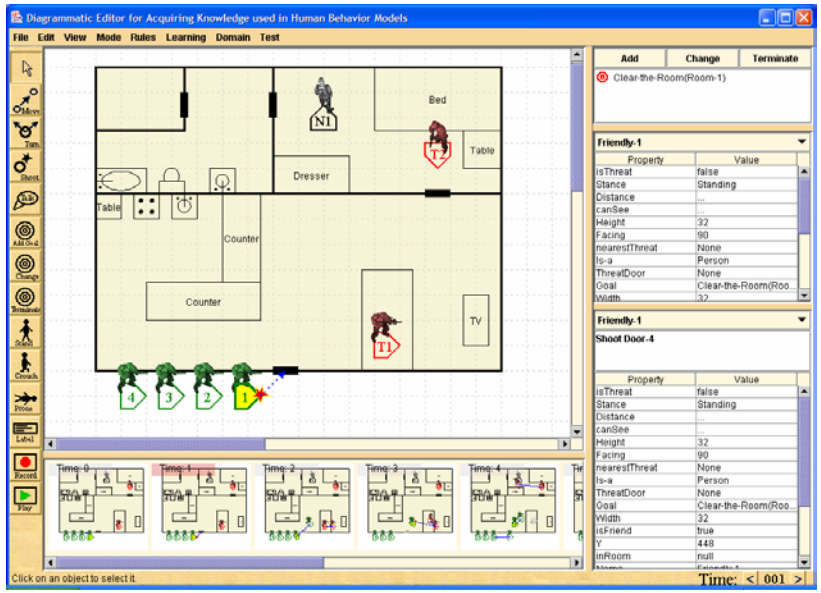
Simulation Environment

ThreePenny SOFTWARE

Tool demo

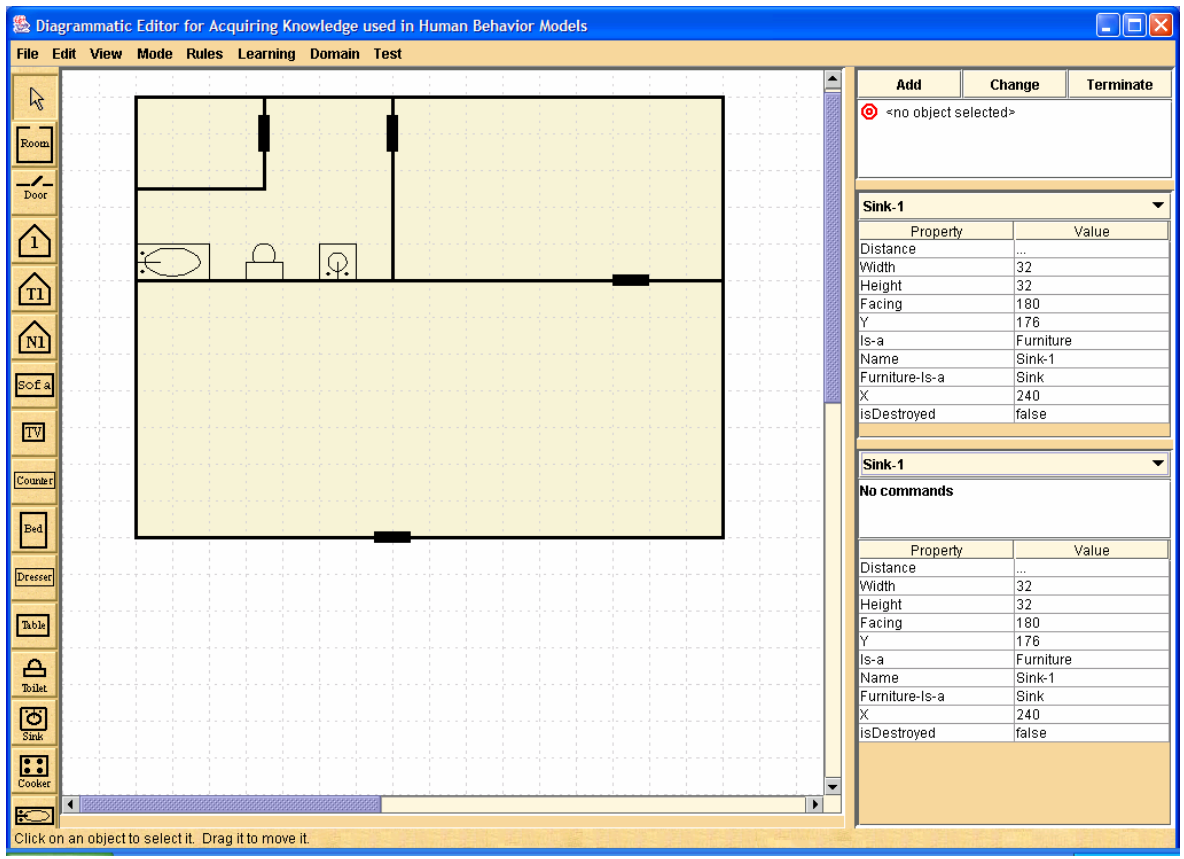
HBM Development Phases:

1. SME defines scenario
2. SME defines correct behavior
3. SME+KE indicate why behavior is chosen
4. Rules generated and automatically



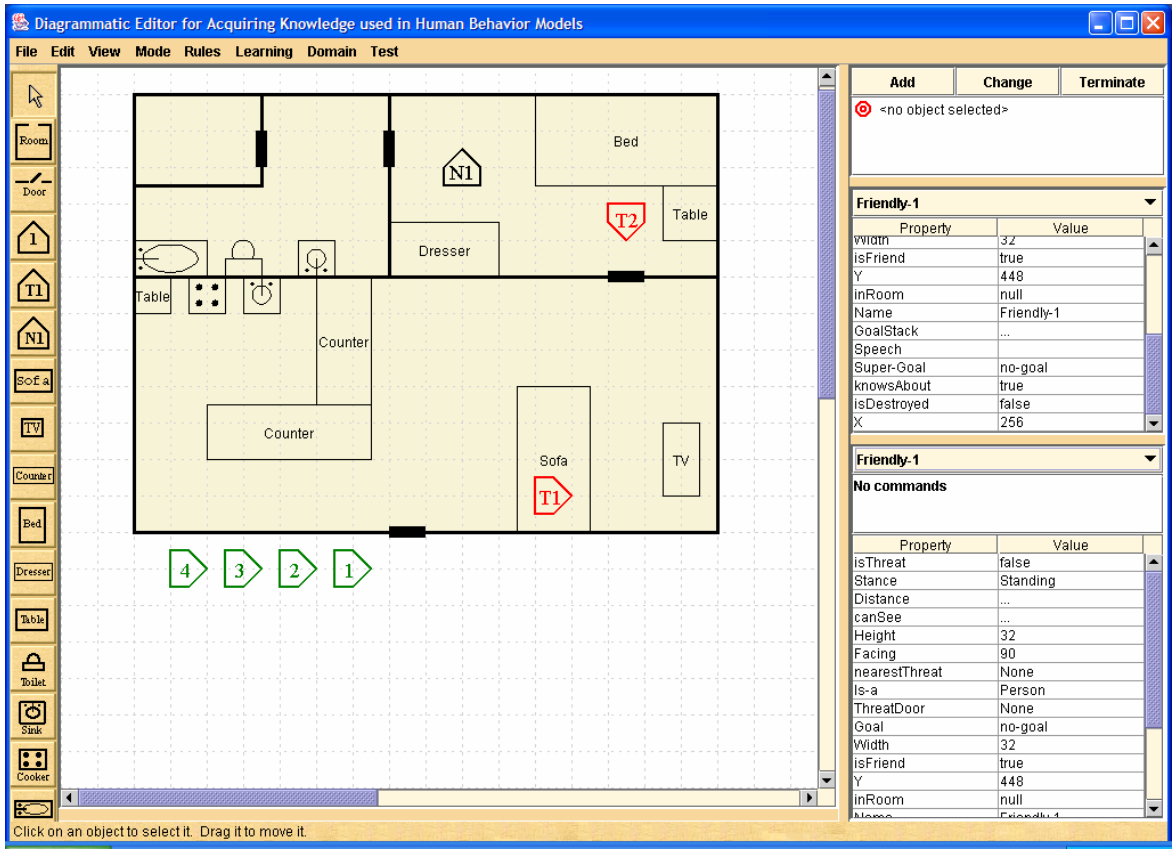
ThreePenny SOFTWARE

Demo Screenshots: Layout Rooms



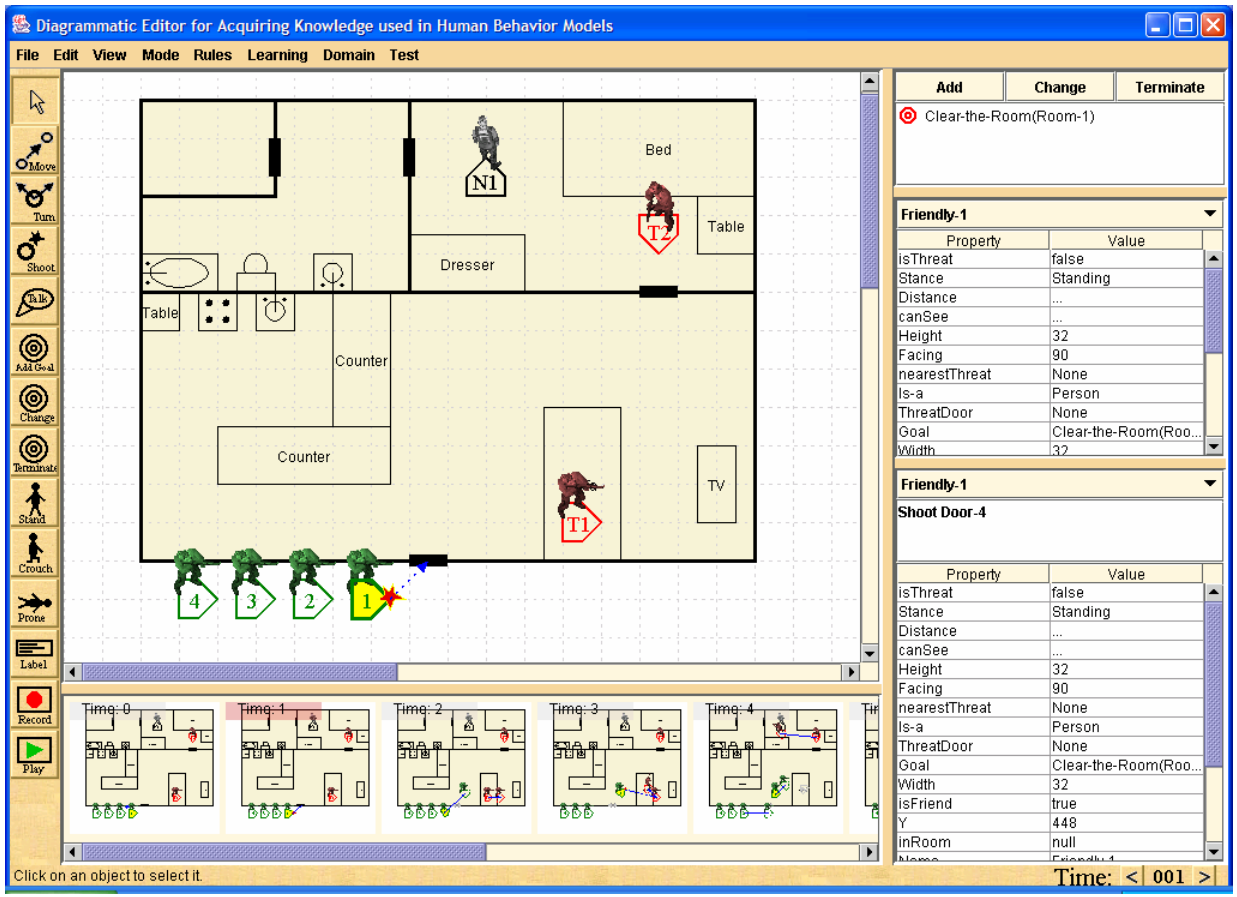
ThreePenny SOFTWARE

Demo Screenshots: Populate Rooms



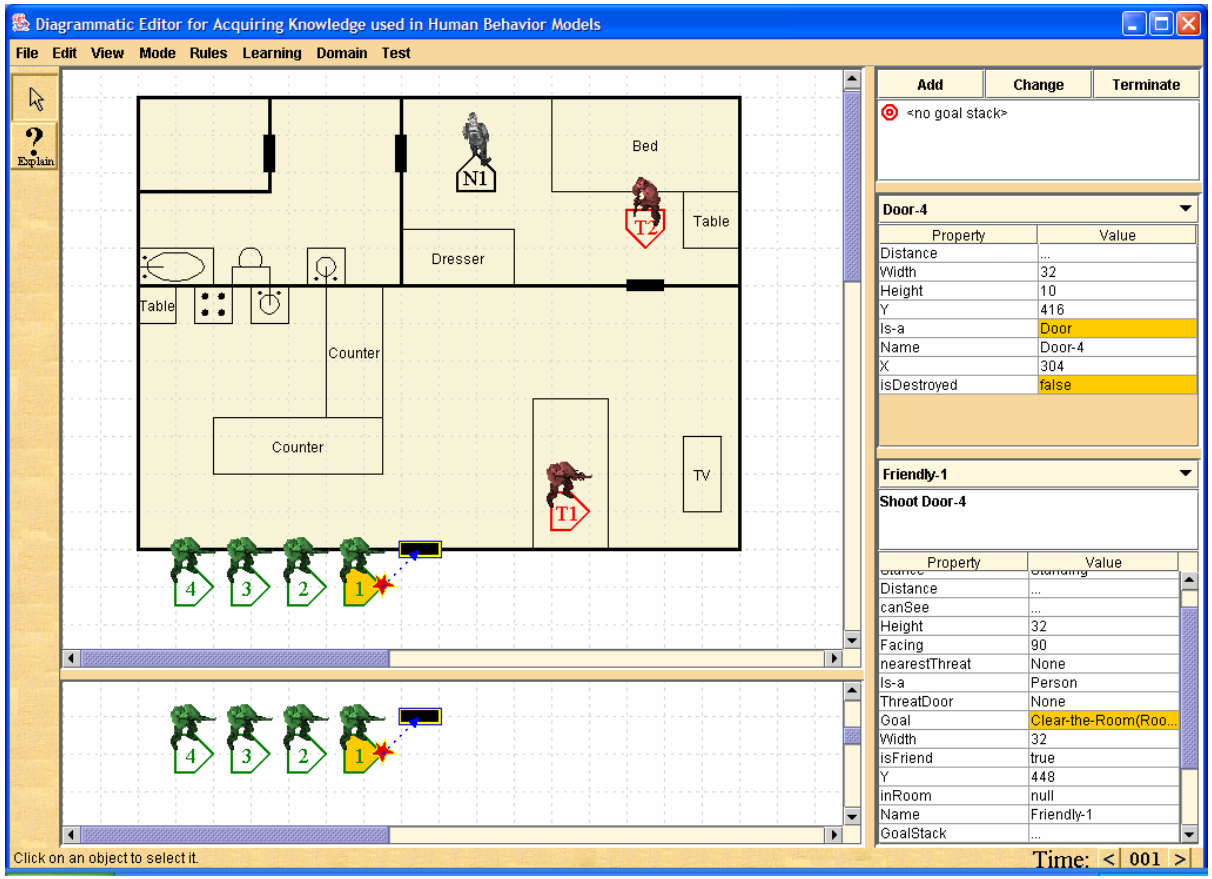
ThreePenny SOFTWARE

Demo Screenshots: Define Scenario



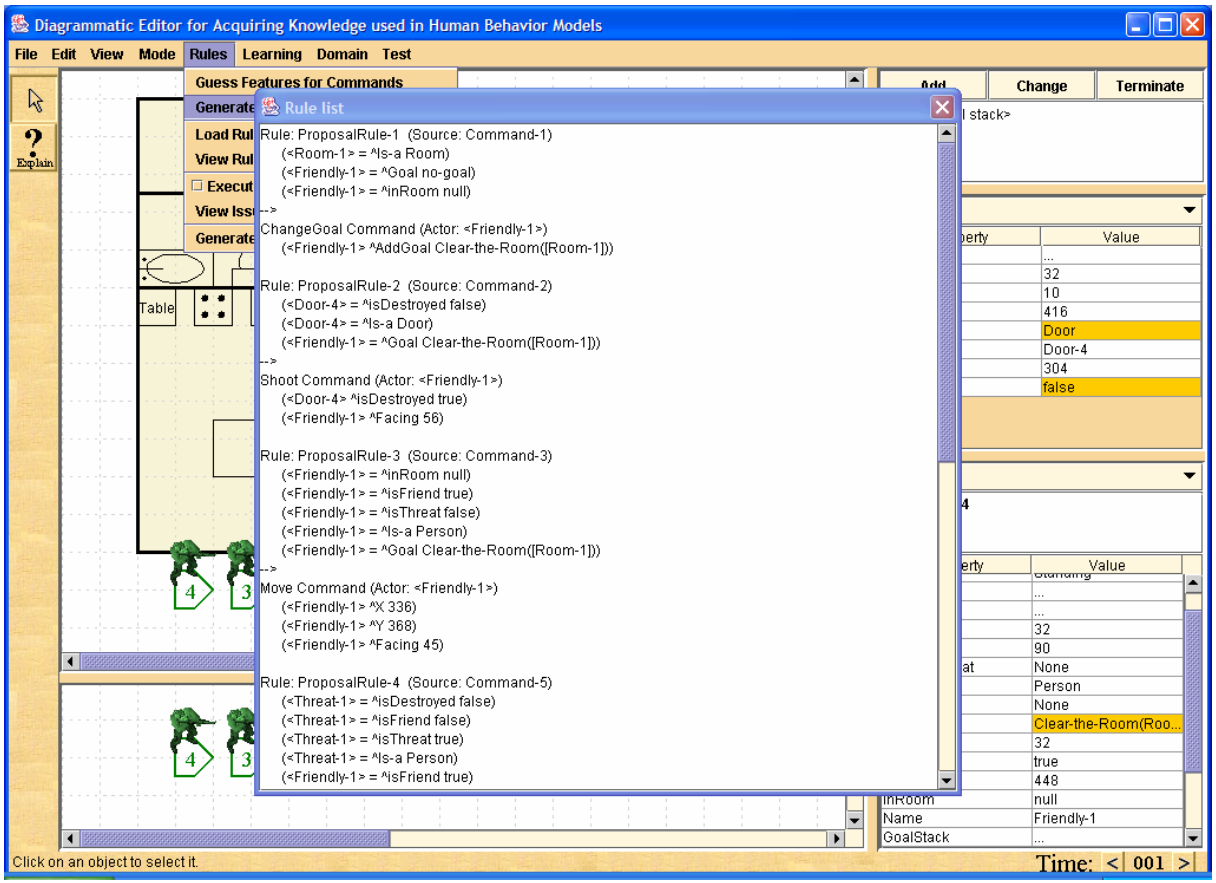
ThreePenny SOFTWARE

Demo Screenshots: Define Behavior



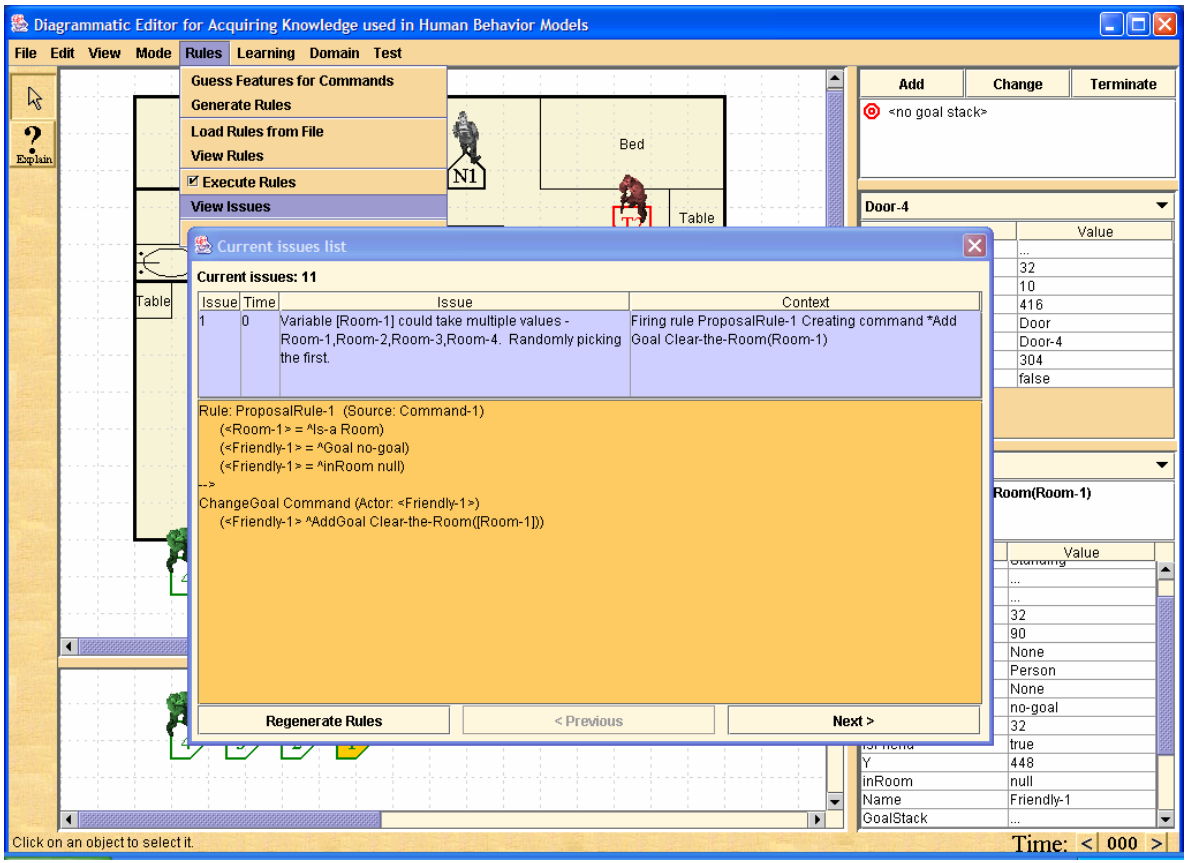
ThreePenny SOFTWARE

Demo Screenshots: Generate Rules



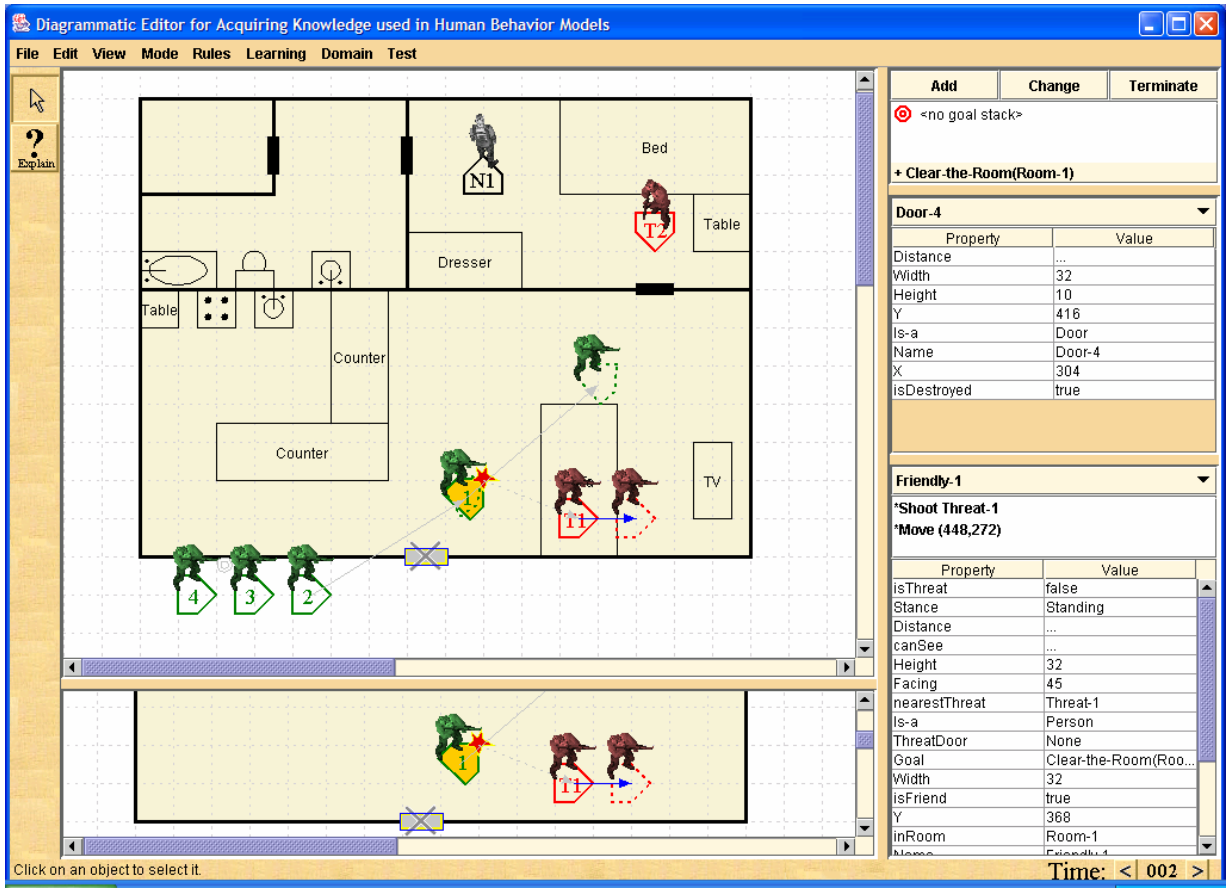
ThreePenny SOFTWARE

Demo Screenshots: Detect Errors



ThreePenny SOFTWARE

Demo Screenshots: Simulate Behavior



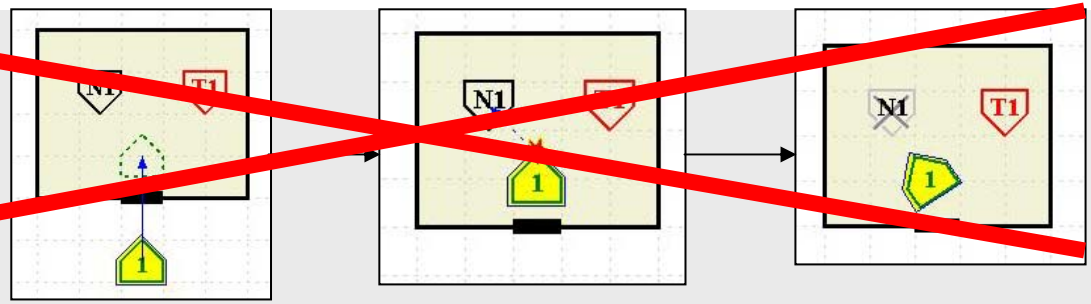
ThreePenny SOFTWARE

Improved HBM Development

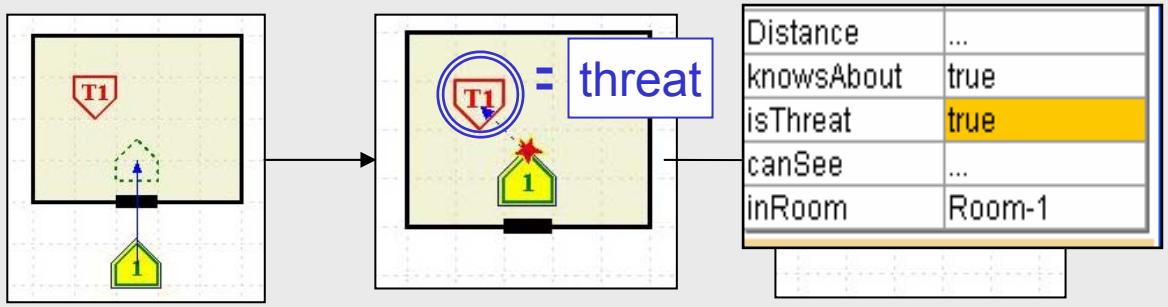
- Minimize HBM Development Cost
  - Extract maximal knowledge from examples
- Ensure Correctness of Behavior
  - Immediate verification on current example
  - *Consistency analysis across example library*
- Minimize HBM Maintenance Cost
  - Maintain validated examples, not code
  - Greatly decrease need for KE
- Minimize Cost of Tool Development & Use
  - Use same tool for multiple domains
  - *Use same tool for multiple target architectures*

Define General Tactics, not Scripted Behavior

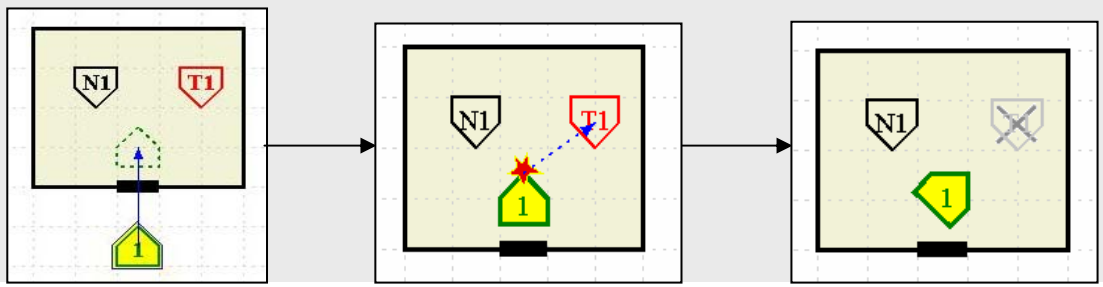
Incorrect Scripted Behavior



Original Example



Correct General Behavior



ThreePenny SOFTWARE

User Controls the Generality of the Examples

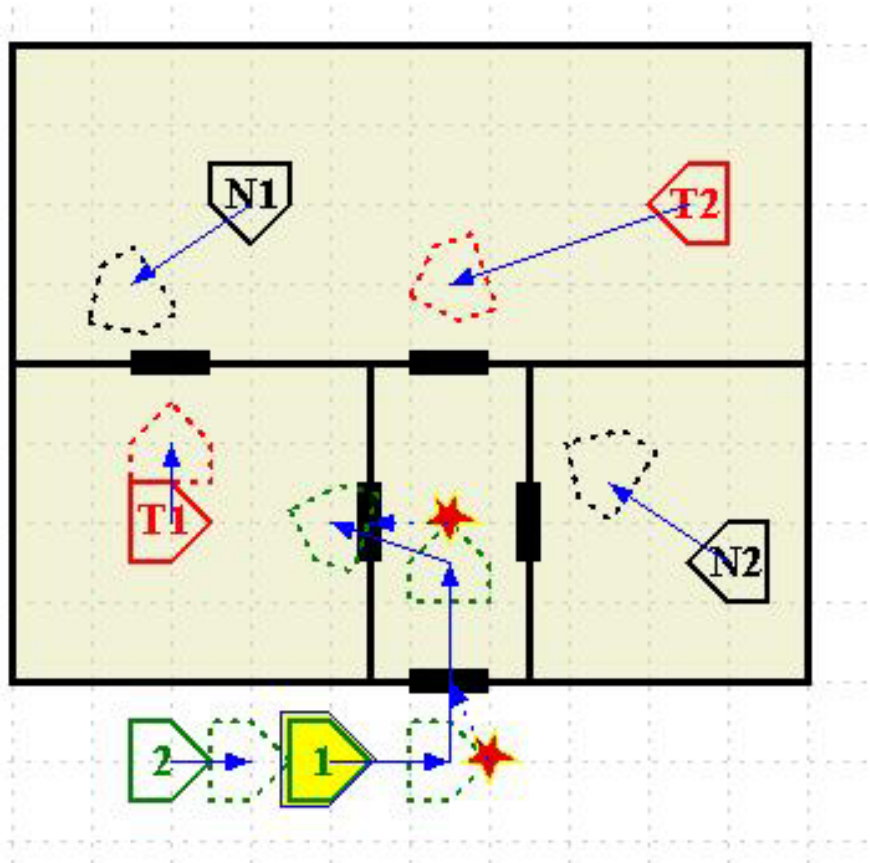
- Select few features
  - General rule
  - More coverage
  - But easier to introduce errors
- Select lots of features
  - Specific rule
  - Less coverage
  - But less chance of errors

Friendly-1	
Property	Value
canSee(Threat-1)	true
Distance(Threat-1)	Medium
Threat-1	
Property	Value
canSee	...
canSee(Friendly-1)	true
Distance	...
Distance(Friendly-1)	Medium
inRoom	Room-1
Is-a	Person
isDestroyed	false
isFriend	false
isThreat	true
nearestThreat	Friendly-1
Speech	
Stance	Standing

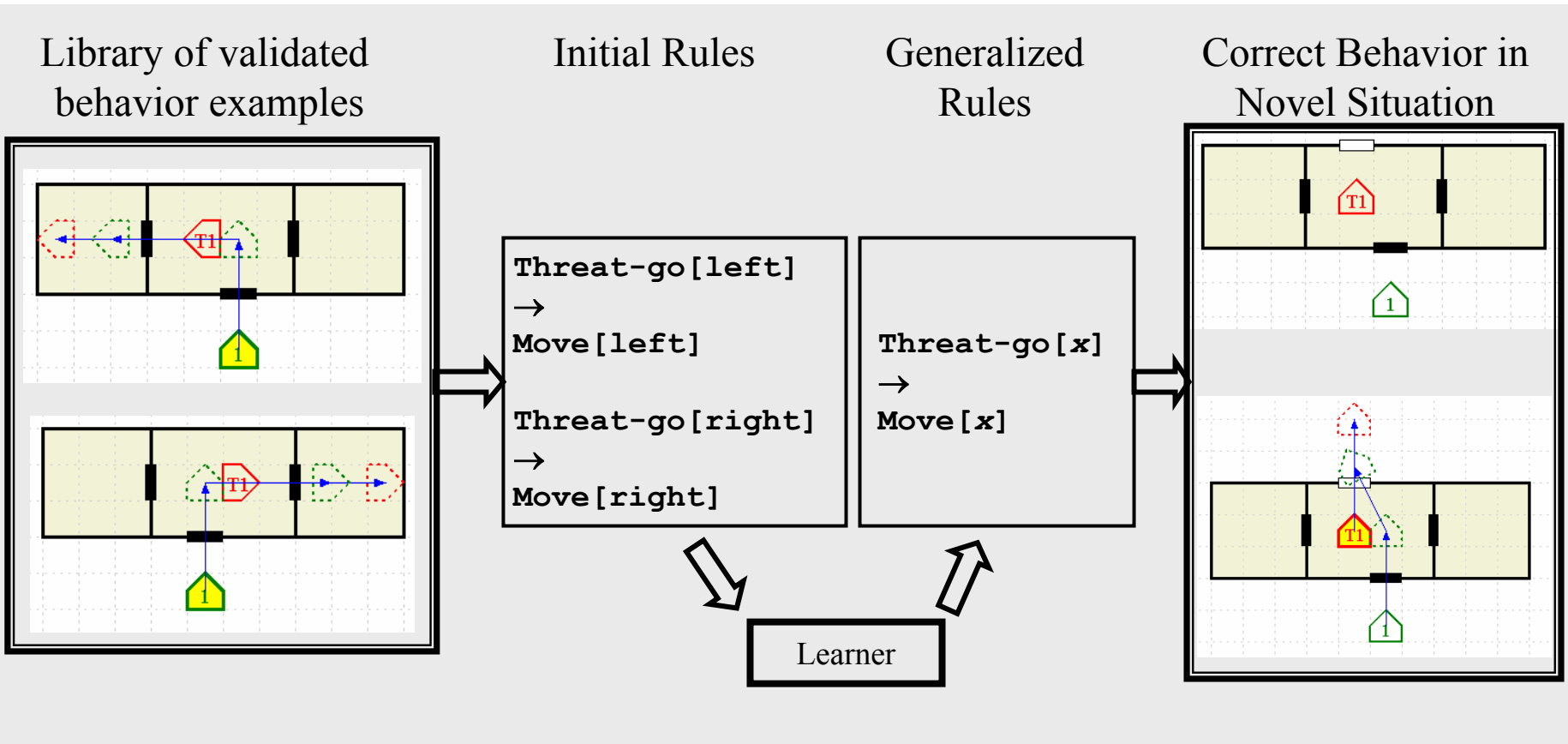
Learn Tactics for all Entities

Specify tactics for many entities in single example

- Friendly forces
- Adversary forces
- Neutrals

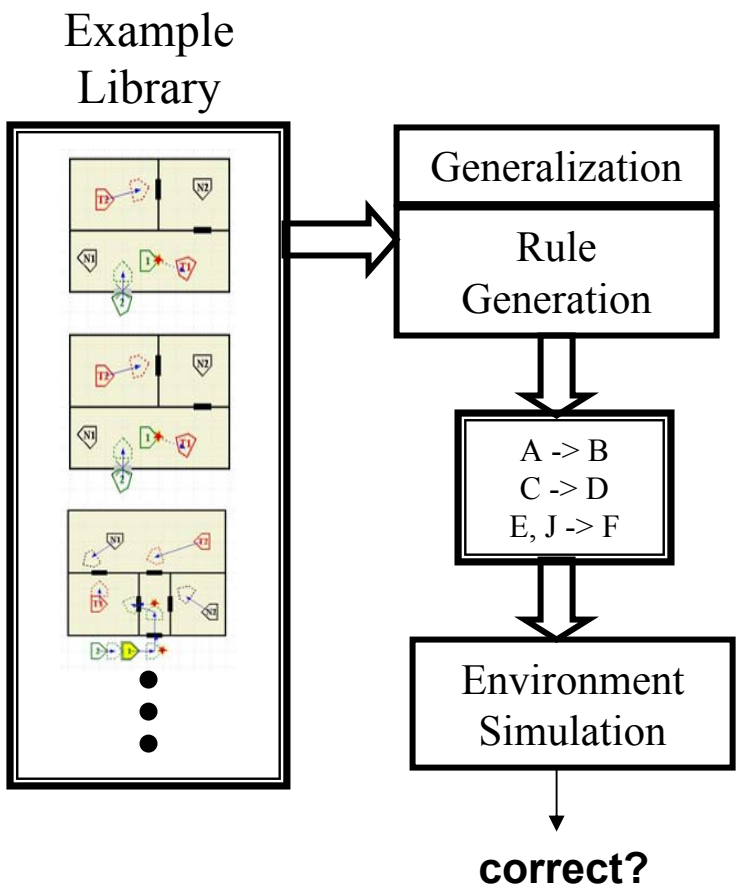


Learn Behavior for Novel Situations

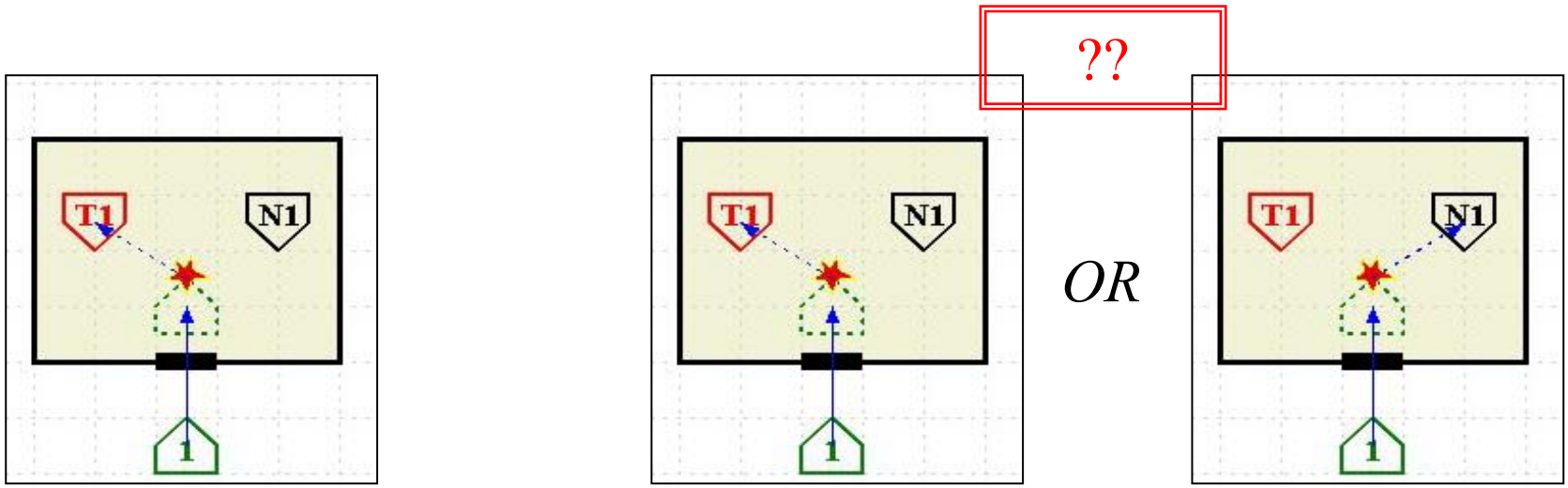


Compare to Desired Behavior Automatically

- Generate rules from example library
- Execute rules on example scenarios
- Detect when rule behavior ≠ expert behavior
- Ensures correctness



Errors are Detected Immediately

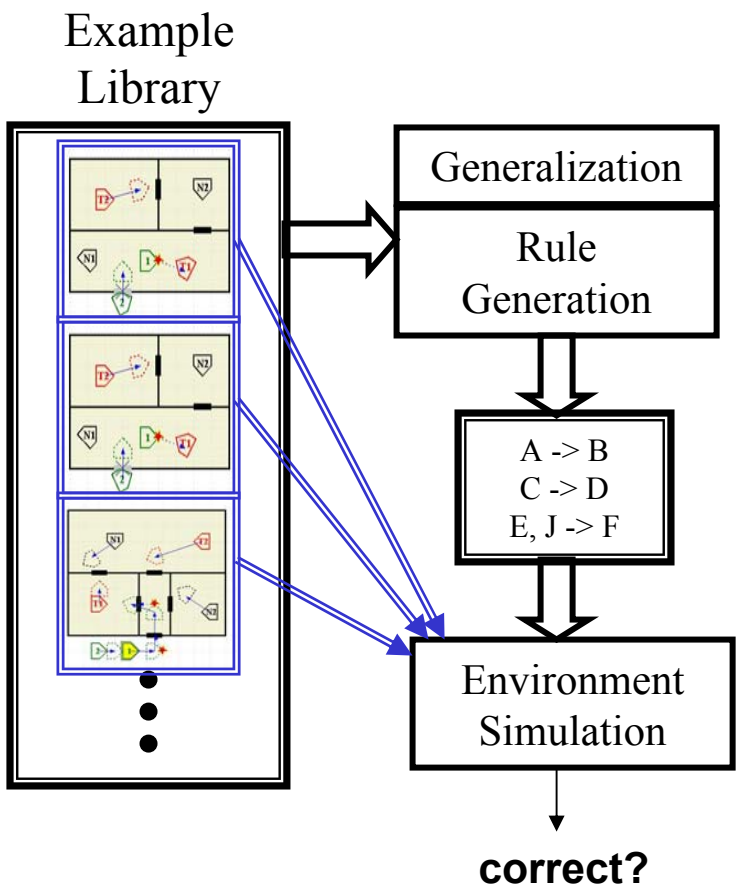


Specified behavior

- Detects if behavior could be interpreted two ways
- Tool detects this at design time – not run time

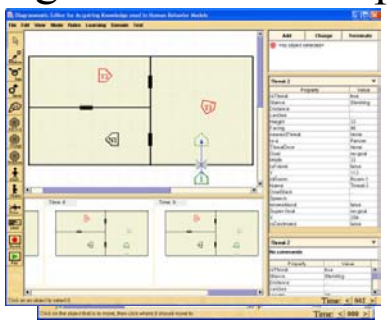
Detect Inconsistencies Across Examples

- Not limited to just testing within scenarios
- Test knowledge automatically across all defined scenarios
- With standard KA this is too expensive – usually not done at all.

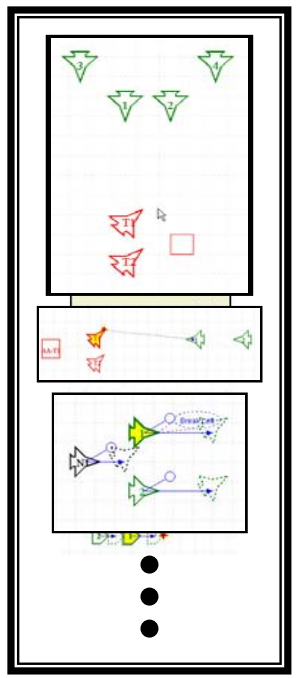


Use Tool in Multiple Domains

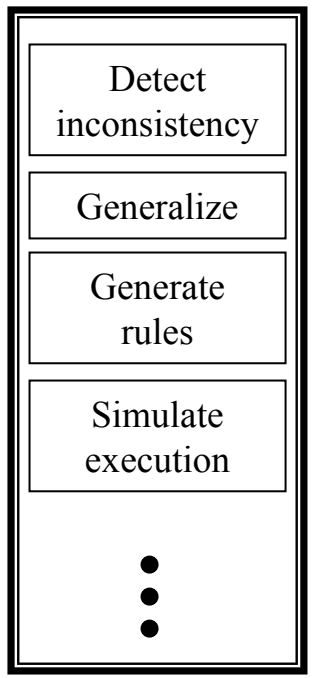
Define behavior with diagram-based examples



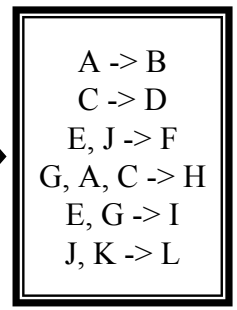
Library of validated behavior examples



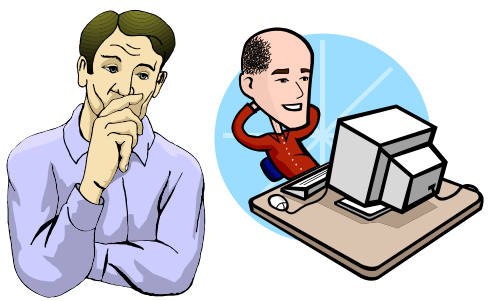
Analysis & generation tools



Rules (Soar)



Simulation Environment

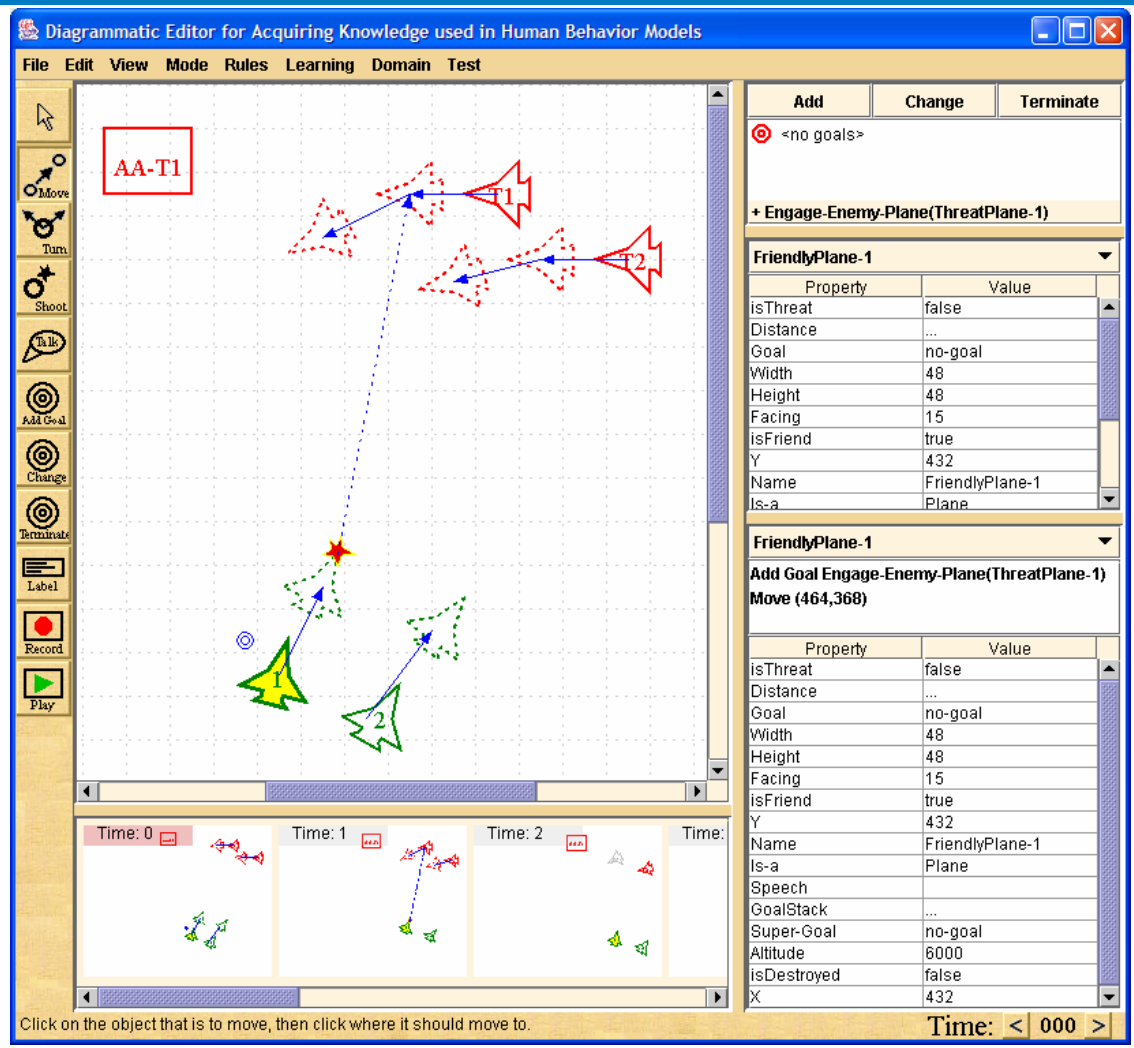


SME KE

90% code base is domain independent

ThreePenny SOFTWARE

Use Same Tool for Air Domain



ThreePenny SOFTWARE

Planned Evaluation

- Build an agent in the MOUT domain and compare against current development efforts.

- Additional funding required to formally evaluate.

ThreePenny SOFTWARE

Nuggets

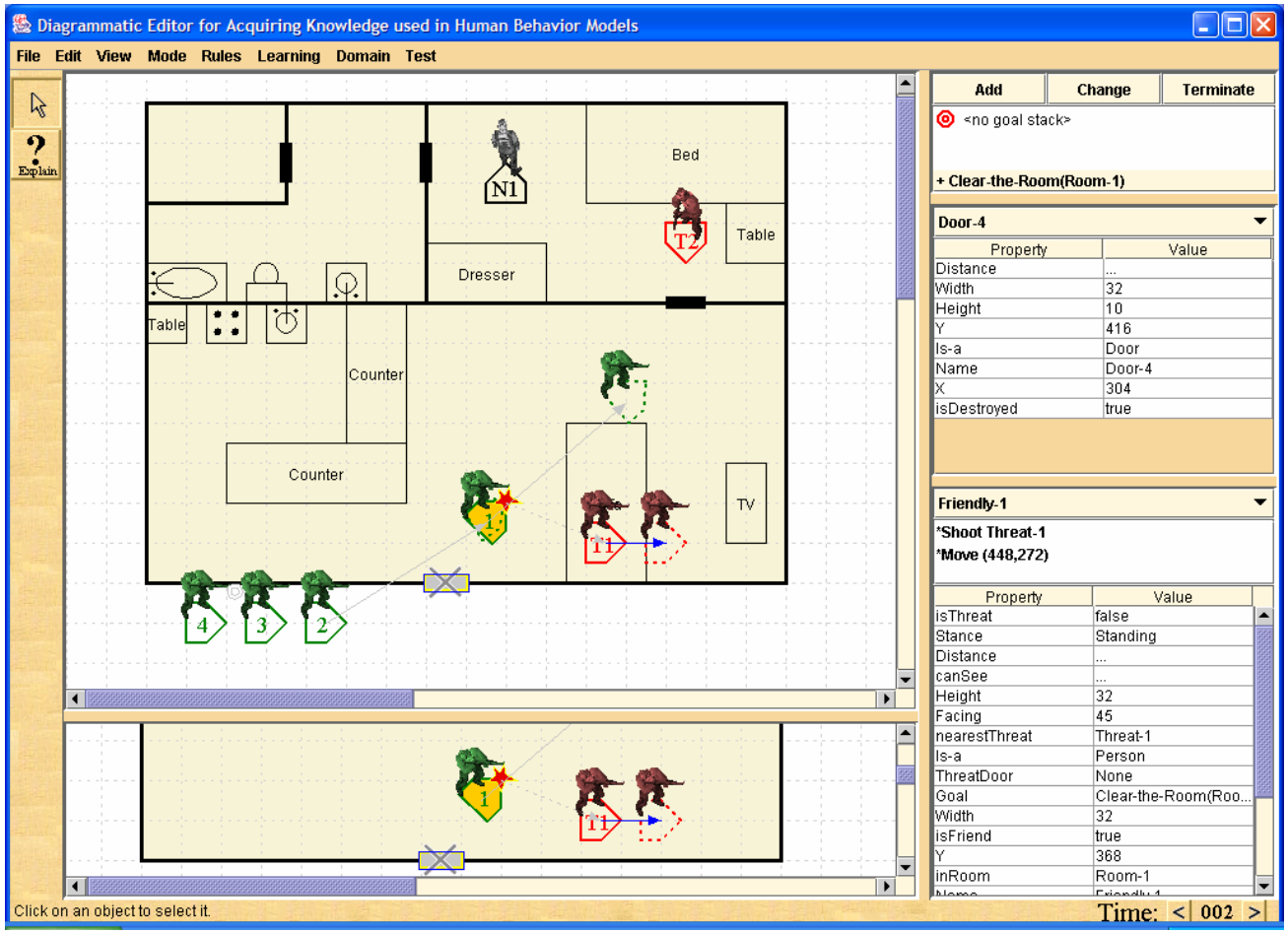
- Development Cost: 5x faster
  - 1-2 hours with tool = 1-2 days without
- Validation: 5x fewer errors
  - Eliminate errors during development
- Maintenance: 5x less effort
  - May eliminate need for KE

- Completely new opportunities for HBM use

Coal

- Still early days – mostly "potential"
- Extending the representation language (e.g. adding a new concept)
- Mapping between simulations and rules
- Interface doesn't make full use of diagrams and too hard for SMEs to use on their own
- Getting new tools adopted is always hard

The End



ThreePenny SOFTWARE