SYSTEMS ENGINEERING RESEARCH CENTER

SERC TALKS

# WELCOME

***Can Graphical Models Provide a Sufficient Basis for General Intelligence?***

Dr. Paul S. Rosenbloom, Institute for Creative Technologies, University of Southern California

**April 5, 2017 | 1:00 pm ET**

- Today's session will be recorded.
- An archive of today's talk will be available at: www.SERCuarc.org
- Use the Q&A box to queue questions, reserving the chat box for comments, and questions will be answered during the last 5-10 minutes of the session.
- If you are connected via the dial-in information only, please email questions or comments to Ms. Mimi Marcus at mmarcus@stevens.edu.
- Any issues? Use the chat feature for any technical difficulties or other comments, or email Ms. Mimi Marcus at mmarcus@stevens.edu.

The Systems Engineering Research Center (SERC) is a federally funded University Affiliated Research Center managed by Stevens Institute of Technology.


USC Institute for Creative Technologies

University of Southern California

# Can Graphical Models Provide a Sufficient Basis for General Intelligence?

**Paul S. Rosenbloom**
Professor of Computer Science
Director for Cognitive Architecture Research
rosenbloom@usc.edu

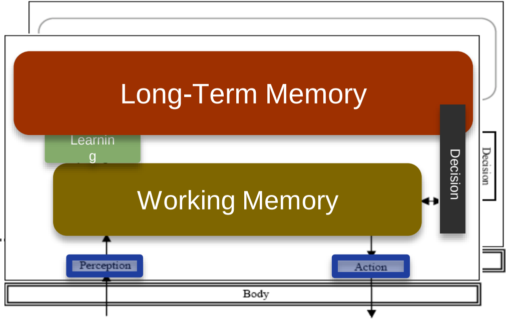
**4.5.2017**

The work depicted here was sponsored by the U.S. Army, the Office of Naval Research, and the Air Force Office of Scientific Research. Statements and opinions expressed do not necessarily reflect the position or the policy of the United States Government, and no official endorsement should be inferred.

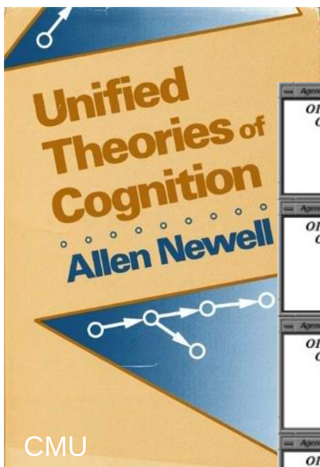
# Preliminary Definitions

- Intelligence
  - "… a very general mental capability that, among other things, involves the ability to reason, plan, solve problems, think abstractly, comprehend complex ideas, learn quickly and learn from experience." *(Editorial in Intelligence with 52 signatories)*
- General intelligence
  - What is common across cognitive tasks
- Artificial Intelligence
  - "… the scientific understanding of the mechanisms underlying thought and intelligent behavior and their embodiment in machines." (AAAI)
- Artificial general intelligence
  - The ability of a machine to perform any (human) cognitive task

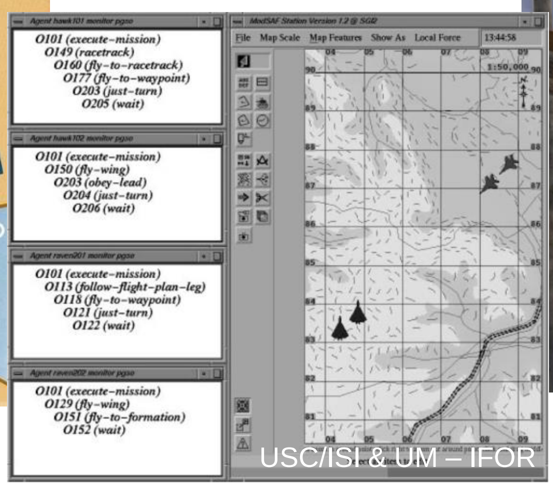
# Cognitive Architecture



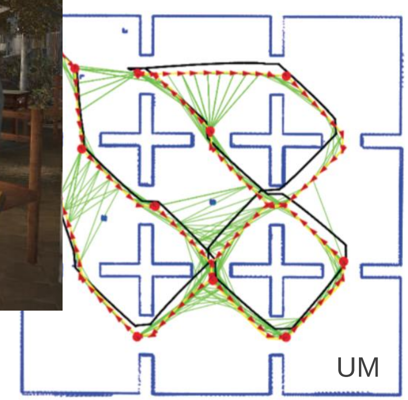

- Model of the fixed structure of a/the mind
  - Memory, reasoning, learning, interaction, ...
  - Integration across these capabilities
- Supports knowledge and skills above the architecture



CMU


USC/ISI & UM – IFOR


USC/ICT – SASO


UM

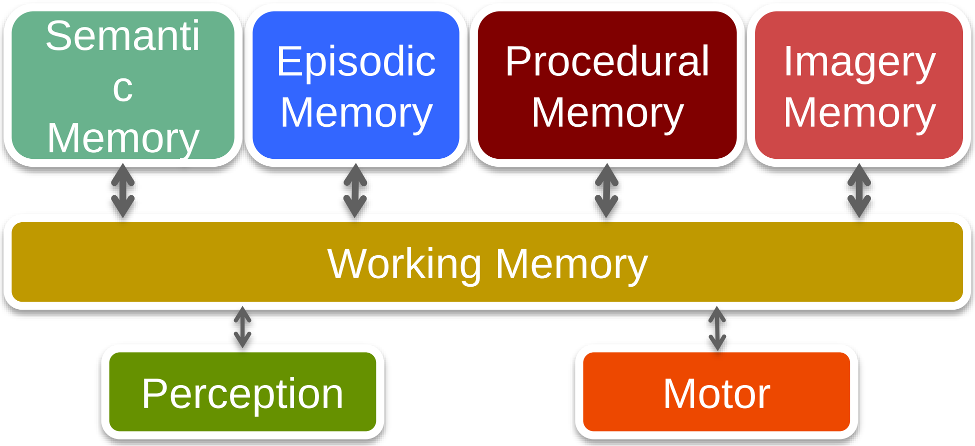
Examples are from **Soar**, an effort I co-led for 15 years

# Overall Desiderata for the Sigma (Σ) Architecture

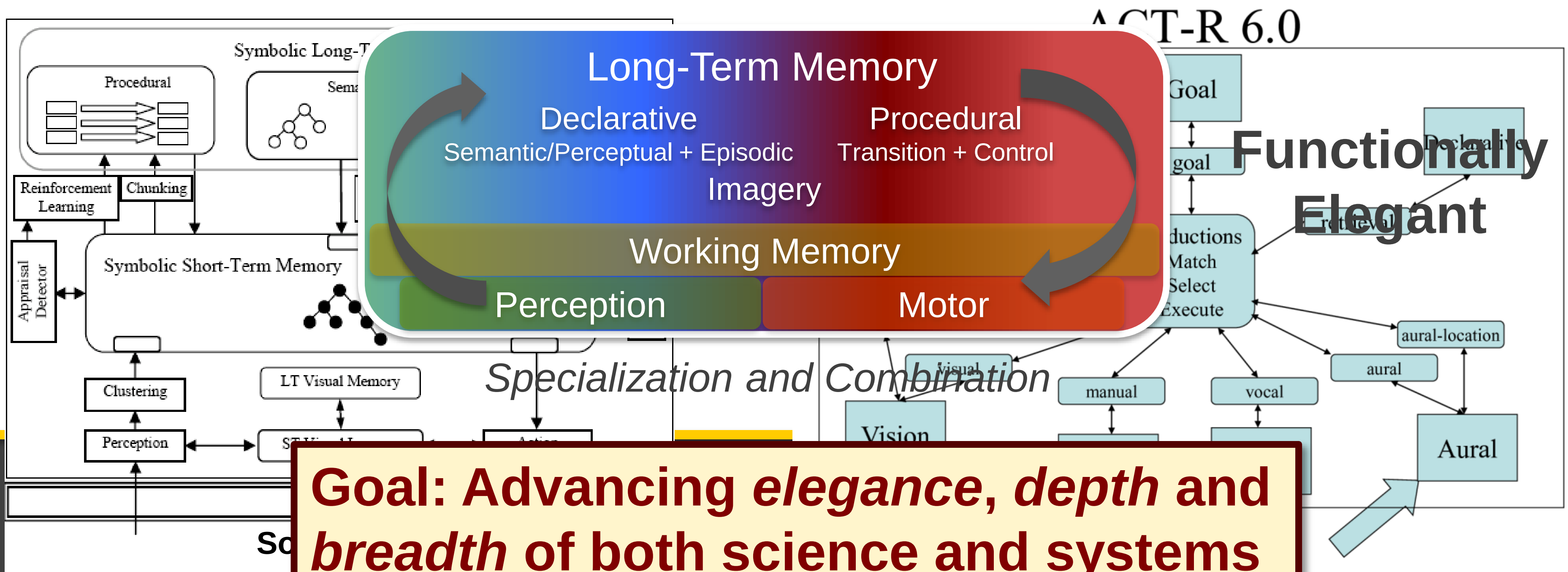
- A new breed of cognitive architecture that is
  - *Grand unified*
    - Cognitive + key non-cognitive (perceptuomotor, affective, attentive, …)
  - *Generically cognitive*
    - Spanning both natural and artificial cognition
  - *Functionally elegant*
    - Broadly capable yet simple and theoretically elegant
      - "cognitive Newton's laws"
  - *Sufficiently efficient*
    - Fast enough for anticipated applications
- For virtual humans & intelligent agents/robots that can
  - Think – Broadly, deeply and robustly *cognitive*
  - Behave – *Interactive* with their physical and social worlds
  - Learn – *Adaptive* given their interactions and experience

*Hybrid*: Discrete + Continuous
*Mixed*: Symbolic + Probabilistic

# Modular versus Functionally Elegant




Modular

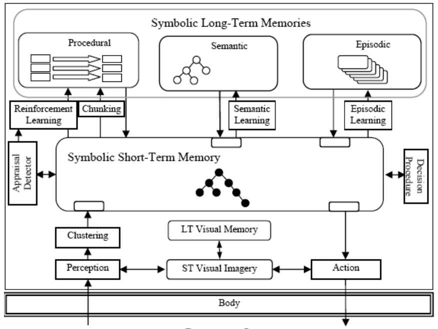



**Goal: Advancing *elegance*, *depth* and *breadth* of both science and systems**

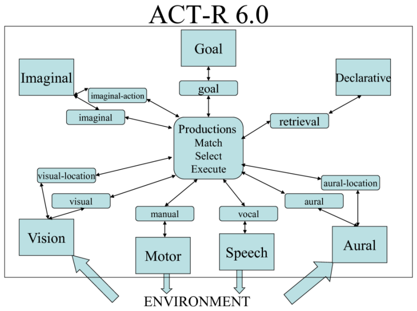
# Approach: Graphical Architecture Hypothesis

Key to success is *blending what has been learned from over three decades of independent work* in **cognitive architectures** and **graphical models**

## Cognitive Architectures

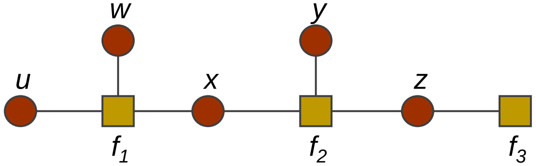



Soar 9




## Graphical Models

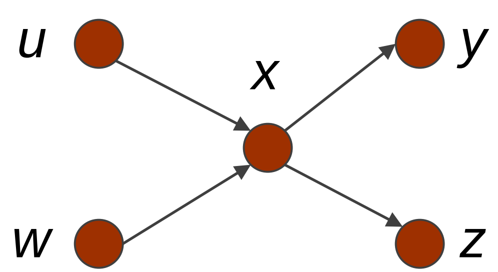
$f(u,w,x,y,z) = f_1(u,w,x)f_2(x,y,z)f_3(z)$



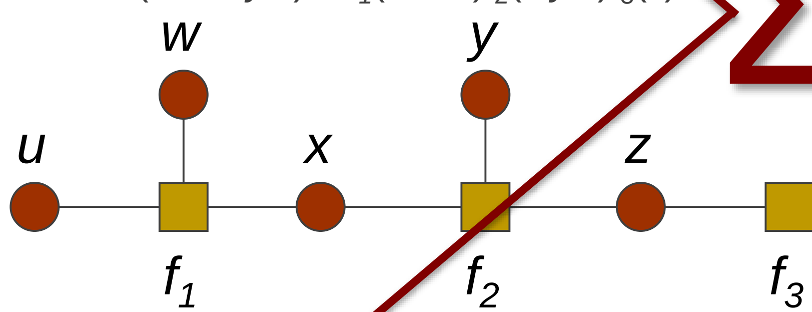
# Graphical Models

- Efficient computation over multivariate functions by leveraging forms of independence to decompose them into products of simpler subfunctions
  - Bayesian/Markov networks, Markov/conditional random fields, factor graphs

$p(u,w,x,y,z) = p(u)p(w)p(x|u,w)p(y|x)p(z|x)$




$f(u,w,x,y,z) = f_1(u,w,x)f_2(x,y,z)f_3(z)$



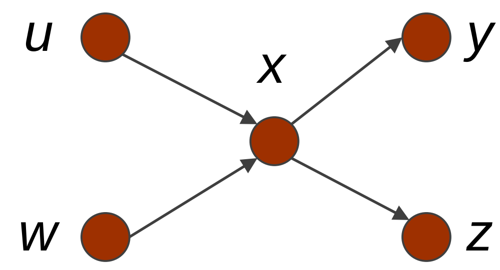

$\Sigma$

- Solve typically via some form of message passing or sampling
- State of the art performance across *symbols*, *probabilities* and *signals* from uniform representation and reasoning algorithm
  - (Loopy) belief propagation, forward-backward algorithm, Kalman filters, Viterbi algorithm, FFT, turbo decoding, arc-consistency, production match, …
- Can support mixed and hybrid processing
- Several neural network models map directly onto them

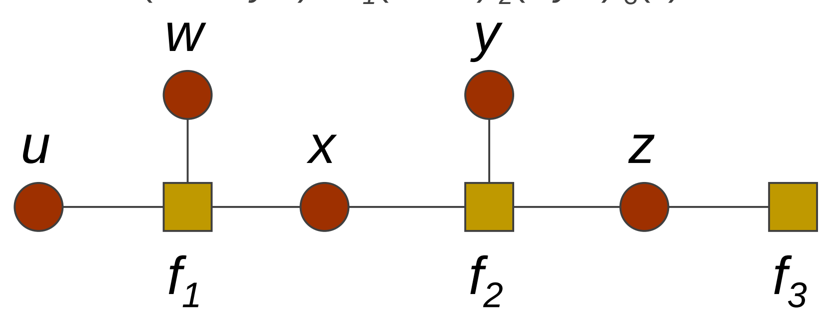
# Bayesian Network vs. Factor Graph

- Bayesian network
  - Directed graph
  - Only variable nodes
  - A distribution at each node *n*
    - $p(n \mid parents_n)$
  - Decompose probabilities
- Factor graph
  - Undirected graph
  - Variable and factor nodes
  - A function at each factor node *n*
    - $f_n(vs_n)$
  - Decompose arbitrary functions

$p(u,w,x,y,z) = p(u)p(w)p(x|u,w)p(y|x)p(z|x)$




$f(u,w,x,y,z) = f_1(u,w,x)f_2(x,y,z)f_3(z)$

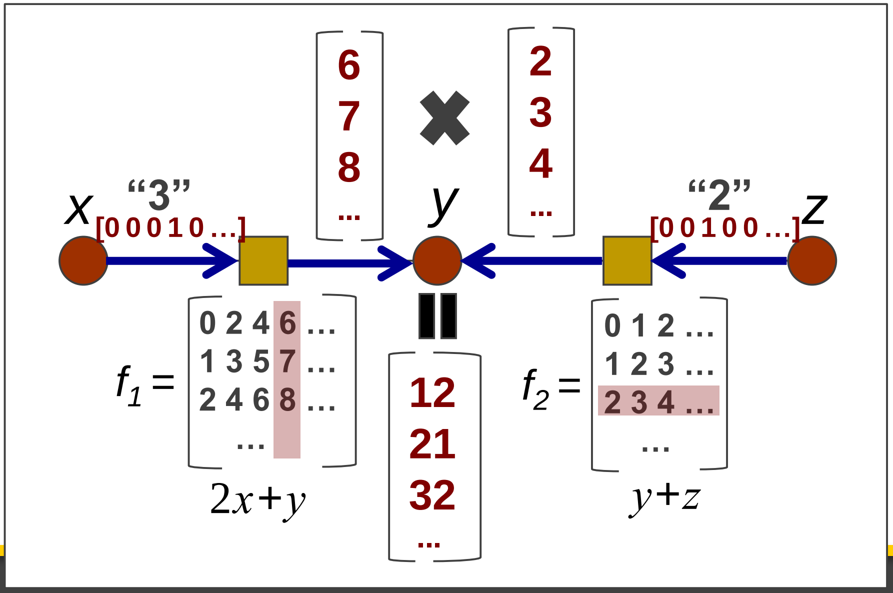


# Summary Product Algorithm

- Compute variable marginals (*sum/integral-product*) or mode of entire graph (*max-product*)
- Pass messages on links and process at nodes
  - Messages are distributions over link variables (starting w/ *evidence*)
  - At variable nodes messages are combined via *pointwise product*
  - At factor nodes do products, and summarize out unneeded variables:

$$m(y) = \sum_x m(x) \times f_1(x,y)$$

$$f(x,y,z) = y^2 + yz + 2yx + 2xz$$
$$= (2x+y)(y+z) = f_1(x,y) f_2(y,z)$$



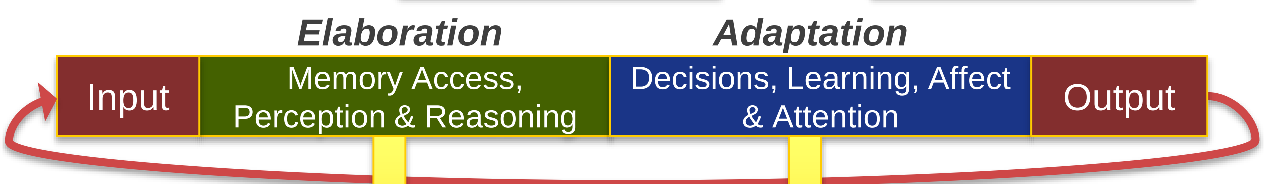
# The Structure of Sigma

## Conjoining the Two Halves of the Hypothesis

$\Sigma$

| Computer System | | Σ Cognitive System |
|---|---|---|
| Programs & Services | | Knowledge & Skills |
| Computer Architecture | ⟷ | Cognitive Architecture |
| Microcode Architecture | ⟷ | Graphical Architecture |
| Hardware | | Lisp |

**Cognitive Architecture:**
- Predicates
- Conditionals
- Nested tri-level control




**Graphical Architecture:**
- Graphical models
- Piecewise linear functions
- Gradient-descent learning

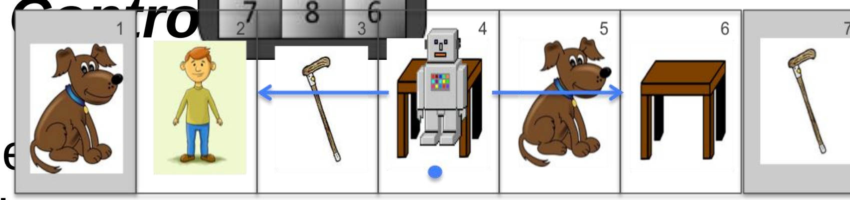
# Sigma's Cognitive Architecture

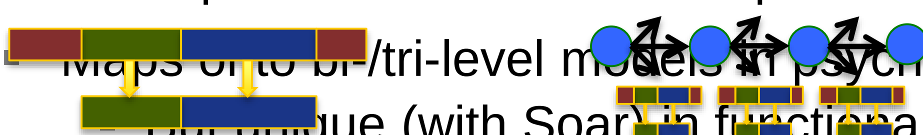
- ***Predicates*** **define relations among typed elements**
  - Both symbolic and numeric (discrete and continuous)
- ***Conditionals*** **yield patterns over pr… s**
  - Deep blend of rule and probabilistic-gra… or

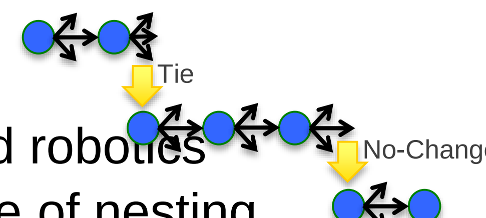
```
Board(x[0-3], y[0-3], tile[0:9])
```


- **(Soar-like) *Nested Tri-Level Contro…***
  - A (parallel) *reactive* layer
    - Single graph/cognitive cycle
  - A (serial/iterative) *deliberative* layer
    - Repeated operator/action selection & application
  - A (recursive) *reflective* layer
    - Impasse-driven meta-level processing
  - Maps onto bi-/tri-level models in psychology and robotics
    - But unique (with Soar) in functional elegance of nesting

```
CONDITIONAL Object-Location-Map
  Conditions: Object(value:o)
  Condacts:   Location(x:x)
  Function(x,o): .25
```



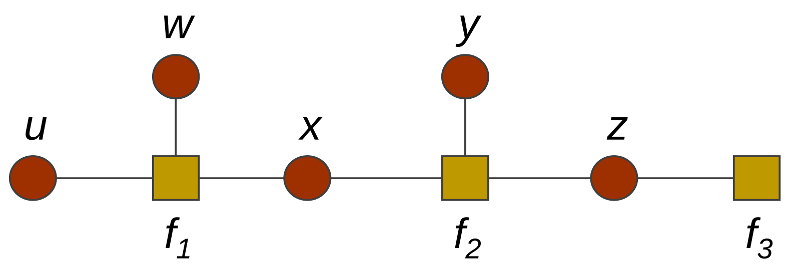





# Sigma's Graphical Architecture

- *Graphical models:* Factor graphs & summary product algorithm

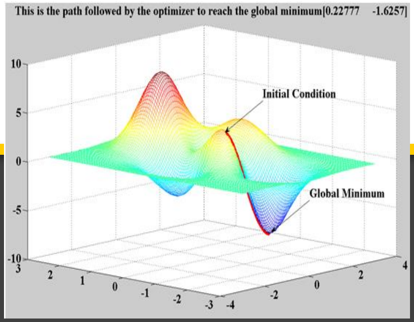
$$f(u,w,x,y,z) = f_1(u,w,x)f_2(x,y,z)f_3(z)$$



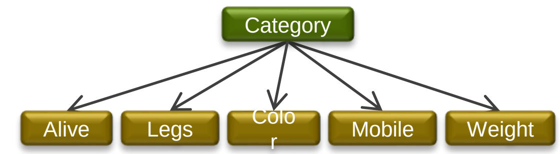

- *Piecewise linear* functions and messages
  - Continuous, discrete & symbolic

| | | |
|---|---|---|
| $.7x+.3y+.1$ | 1 | $.5x+.2$ |
| $.6x-.2y$ | 0 | $x+y$ |
| 3 | .4 | 0 |

- *Gradient-descent learning* of functions locally at factor nodes



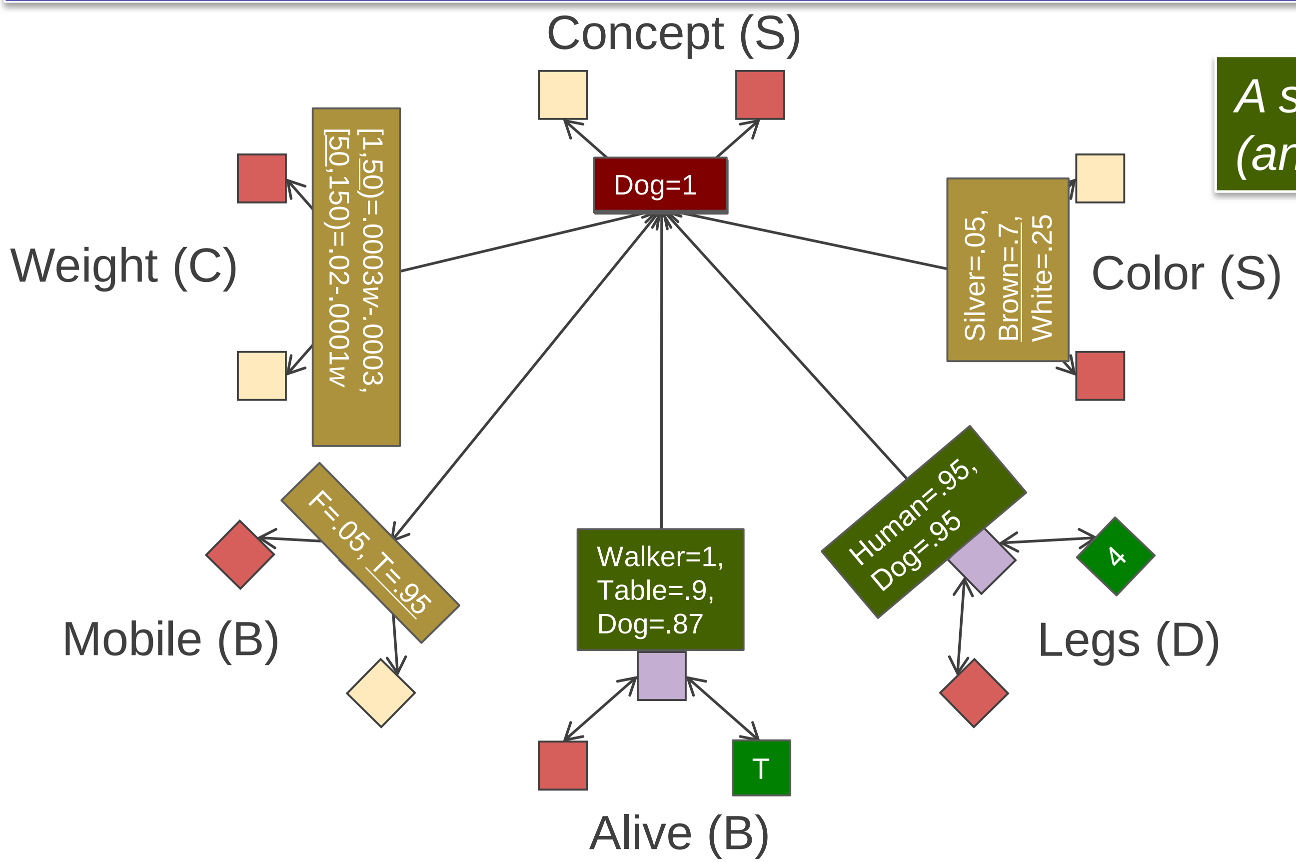
# Example: Semantic Memory (SM) Graph




Naïve Bayes classifier

**Given** cues, **retrieve/predict** object category and missing attributes

E.g., **Given** *Alive*=T & *Legs*=4 **Retrieve** *Category*=Dog, *Color*=Brown, *Mobile*=T, *Weight*=50

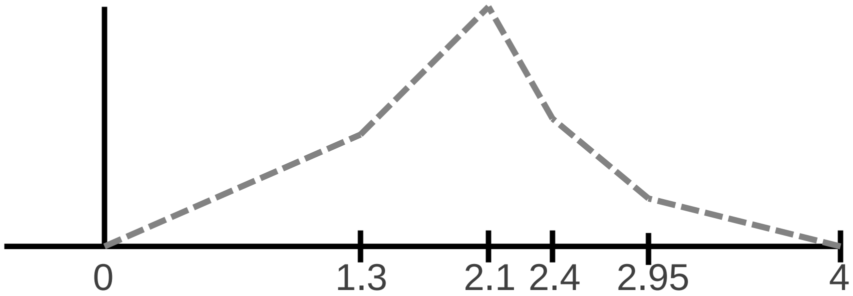



*A subset of factor nodes (and no variable nodes)*

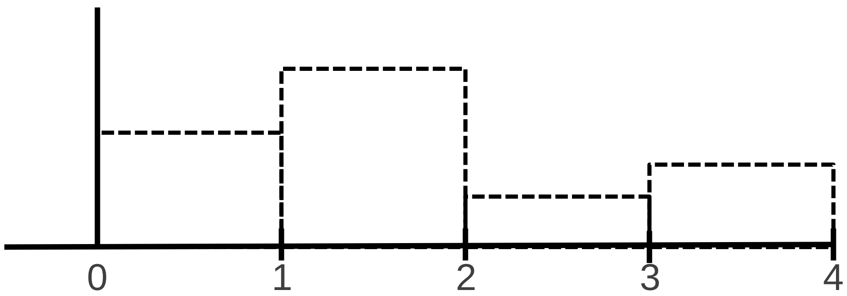
- Function
- Perception
- Join

B: Boolean
S: Symbolic
D: Discrete
C: Continuous

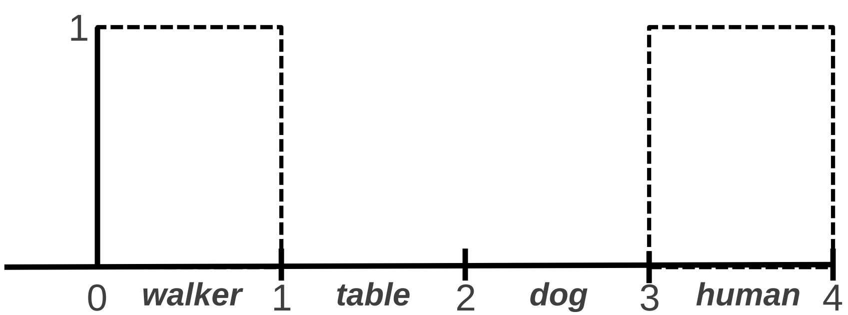
# Piecewise Linear Functions




(a) Continuous (approximation)

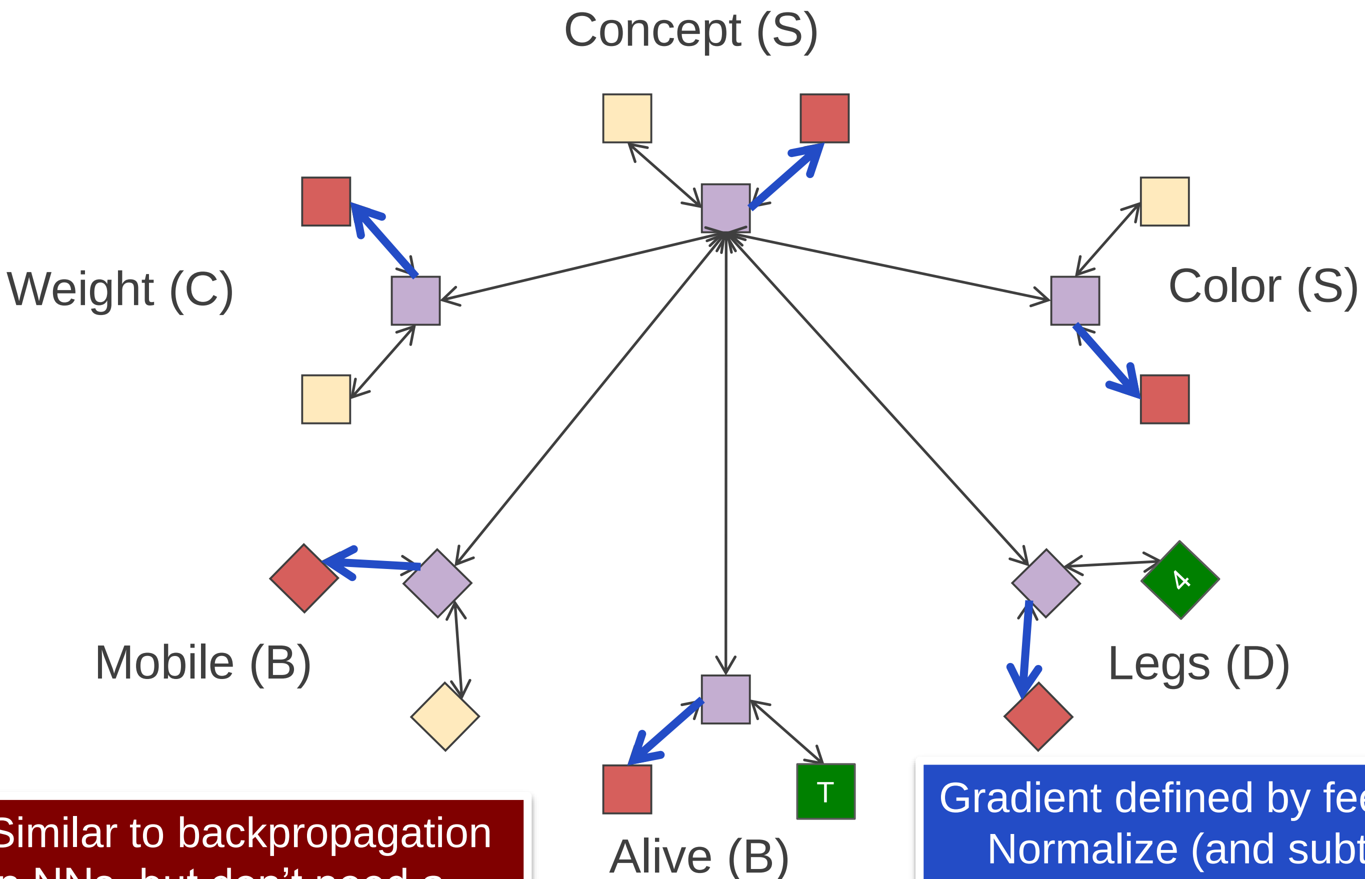



(b) Discrete

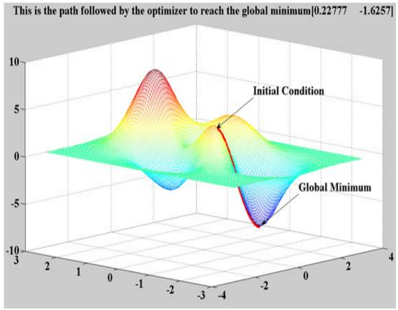



(d) Symbolic

# Learning by Local Incremental Gradient Descent




Local, incremental search for optimal weights




http://www.mathworks.com/matlabcentral/fx_files/27631/1/fff.png

Similar to backpropagation in NNs, but don't need a separate backprop phase

Gradient defined by feedback to function node
- Normalize (and subtract out average)

Multiply by learning rate
Add to function, smooth and normalize

Kurzweil Awards at AGI 2011 & 2012

# Overall Progress on Sigma [JAGI 16]

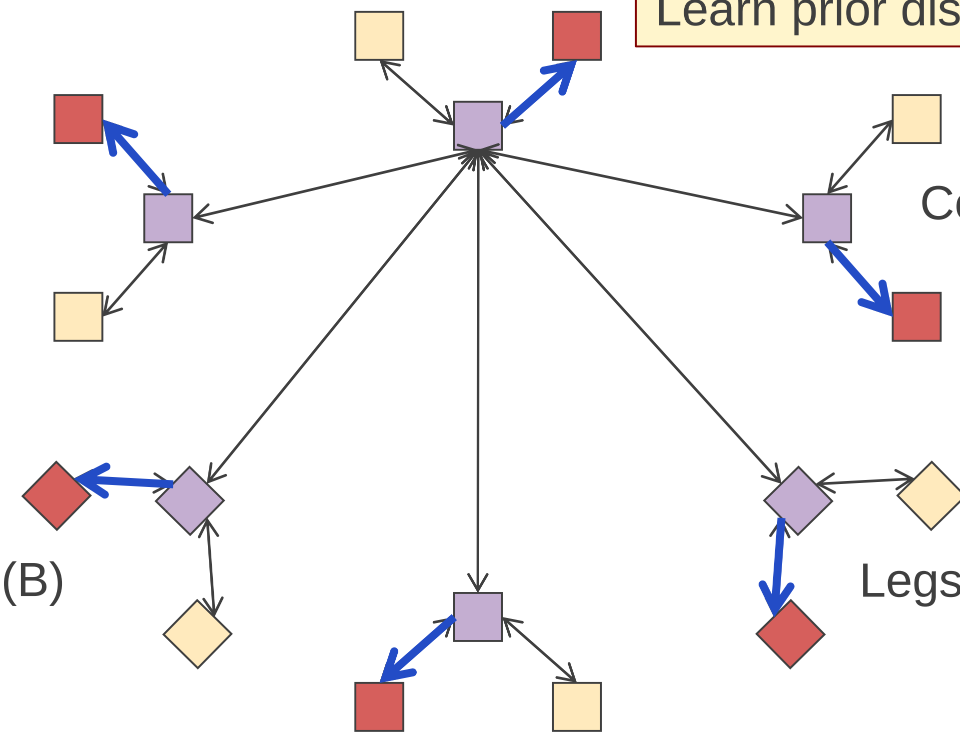
- Memory
  - Procedural (rule) [ICCM 10]
  - Declarative (semantic/episodic) [ICCM 10, CogSci 14]
  - Constraint [ICCM 10]
  - Distributed vectors [AGI 14a]
  - Perceptual [BICA 14a, AGI 15]
  - Neural network [AGI 16]
- Problem solving
  - Preference based decisions [AGI 11]
  - Impasse-driven reflection [AGI 13]
  - Decision-theoretic (POMDP) [BICA 11b]
  - Theory of Mind [AGI 13, AGI 14b]
- Learning [ICCM 13]
  - Concept (supervised/unsupervised)
  - Episodic [CogSci 14]
  - Reinforcement [AGI 12a, AGI 14b]
  - Action/transition models [AGI 12a]
  - Models of other agents [AGI 14b]
  - Perceptual (including maps in SLAM)
  - Neural network
- Efficiency [ICCM 12, BICA 14b]
- Mental imagery [BICA 11a, AGI 12b]
  - 1-3D continuous imagery buffer
  - Object transformation
  - Feature & relationship detection
- Perception
  - Object recognition (CRFs) [BICA 11b]
  - Speech recognition (HMMs) [BICA 14a, BICA 16]
  - Localization [BICA 11b]
- Natural language
  - Word sense disambiguation [ICCM 13]
  - Part of speech tagging [ICCM 13]
  - Sentence identification [WS 15]
  - Dialogue [WS 15]
- Affect [AGI 15]
  - Appraisal
  - Attention
- Integration
  - CRF+Localization+POMDP [BICA 11b]
  - Rules+SLAM+RL+ToM+VH [IVA 15, WS 15]
  - SLAM+Appraisal+Attention+VH
  - SentenceID+Dialogue [WS 15, ICAVCD 16]

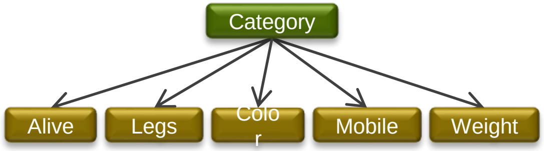
Rosenbloom, P. S., Demski, A. & Ustun, V. (2016). The Sigma cognitive architecture and system: Towards functionally elegant grand unification. *Journal of Artificial General Intelligence*, *7*, 1-103.

# LEARNING

# Example: Supervised Naïve Bayes Probabilistic Classifier Learning

Learn prior distribution on Concept: P(C)




Naïve Bayes classifier

P(C,A,L,Col,M,W) =
P(C)P(A|C)P(L|C) …

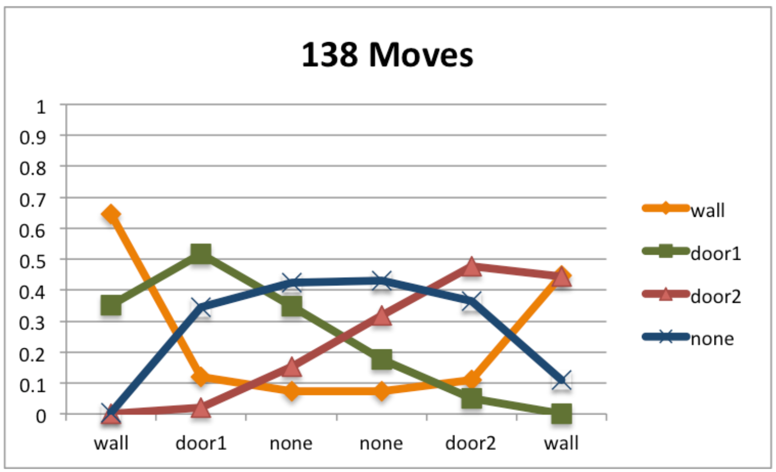
Learn conditional distributions on features given Concept: P(*f* | C)

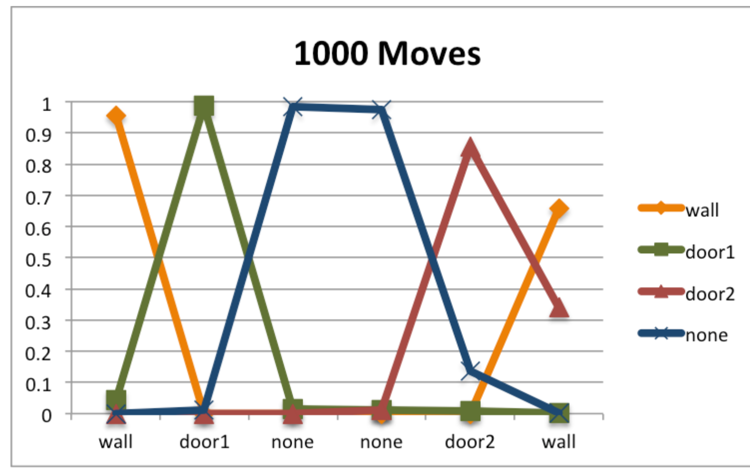
# Example: Learning Maps in SLAM

- Map: **P**(Objects | Locations)

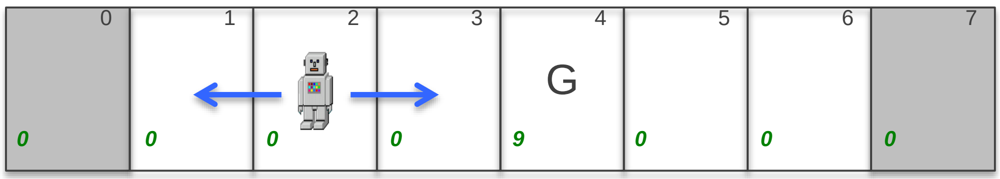
**CONDITIONAL** *Object-Location-Map*
    **Conditions:** Object(value:*o*)
    **Condacts:** Location(x:*x*)
    **Function(*x*,<u>*o*</u>):** .25

| | | | | | |
|---|---|---|---|---|---|
| wall | door1 | | | door2 | wall |



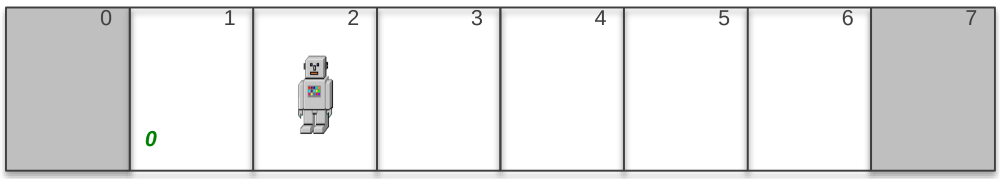
# Example: Reinforcement Learning




Learn values of actions for states by backwards propagation of rewards received during exploration:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]$$

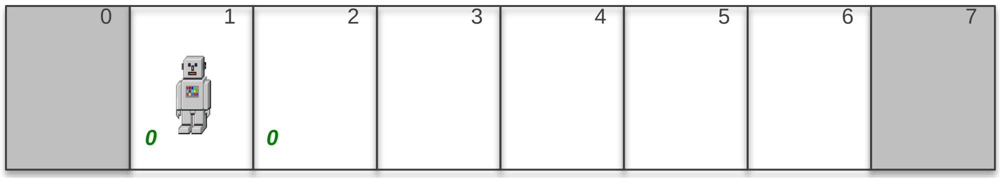
# Example: Reinforcement Learning




Learn values of actions for states by backwards propagation of rewards received during exploration:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]$$

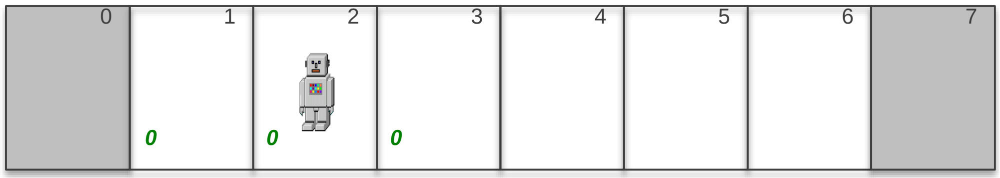
# Example: Reinforcement Learning




Learn values of actions for states by backwards propagation of rewards received during exploration:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]$$

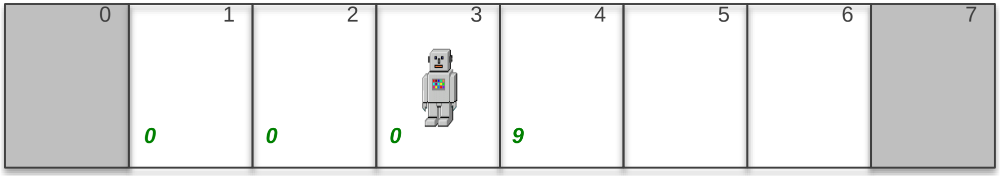
# Example: Reinforcement Learning




Learn values of actions for states by backwards propagation of rewards received during exploration:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]$$

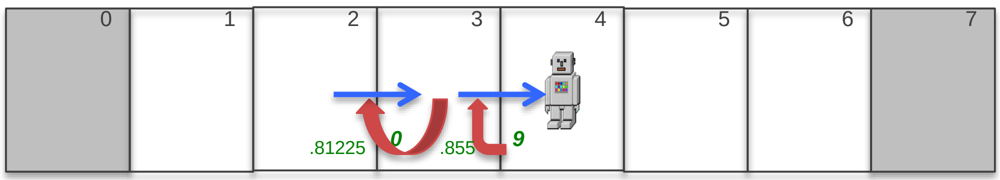
# Example: Reinforcement Learning




Learn values of actions for states by backwards propagation of rewards received during exploration:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]$$

# Example: Reinforcement Learning

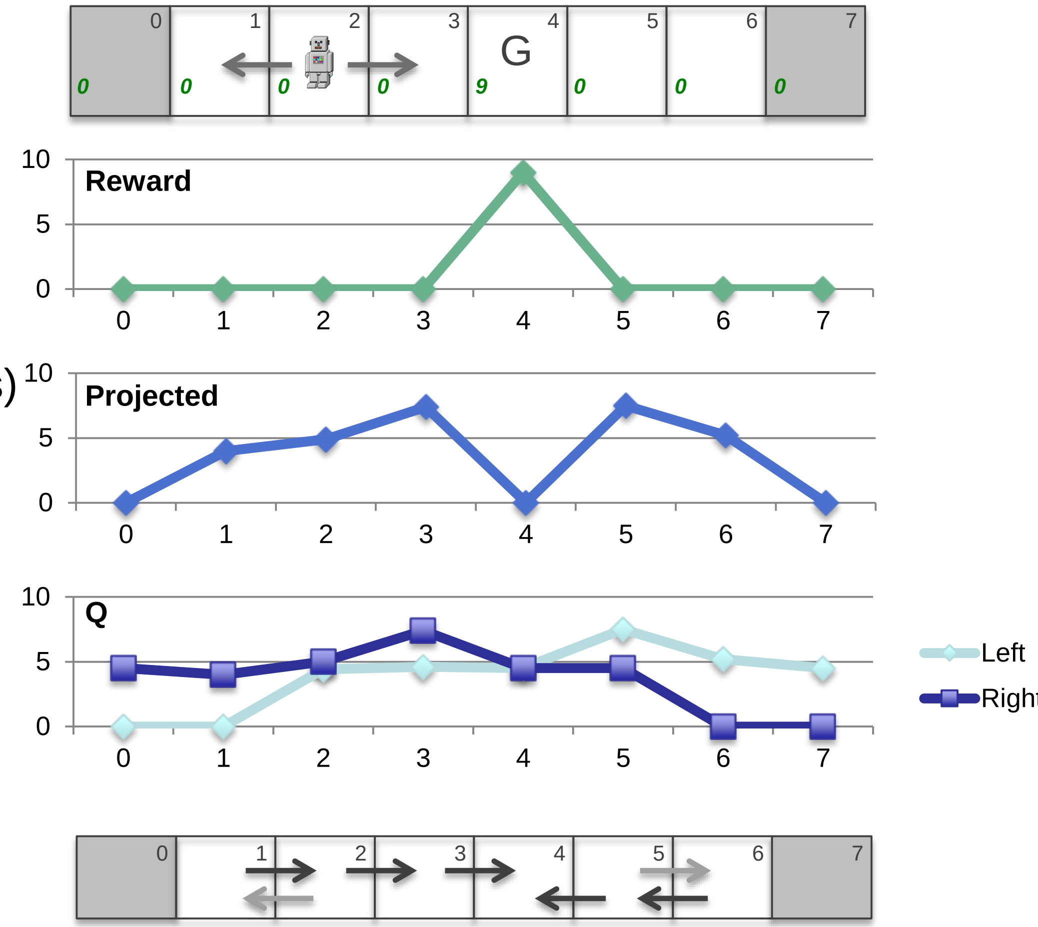



Learn values of actions for states by backwards propagation of rewards received during exploration:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha[r_t + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)]$$

# Deconstructing RL in Sigma

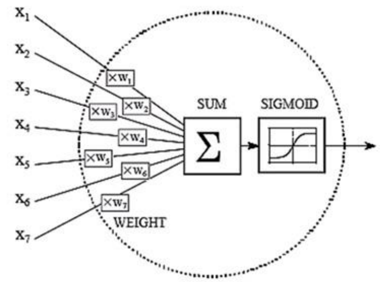
- Knowledge:
  - Initial uniform predictors for:
    - Current reward (*R*)
    - Projected future reward (*P*)
    - Action preferences (*Q*)
  - Regression (backup) knowledge
  - Action models (predict next states)
- Supervised learning of:
  - Current reward (*R*)
  - Projected future reward (*P*)
  - Action preferences (*Q*)
- Add *diachronic cycles* to also learn action models



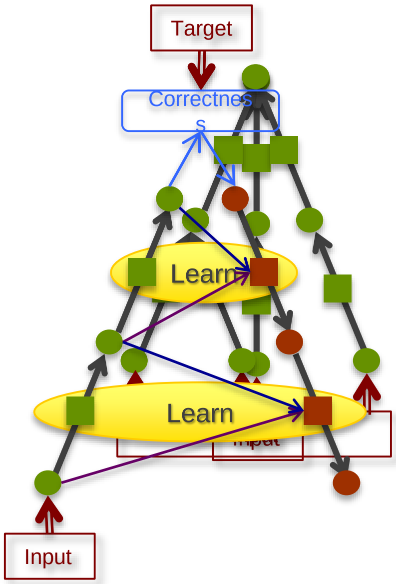

**Graphs are of expected values, but learning is actually of full distributions**

```
CONDITIONAL Layer-0
  Conditions: (Level-0 arg:v0)
  Actions: (Level-1 s arg:v1)
  Function: ws-0-1
```

# Example: Neural Network Learning in Sigma




- Implementing NNs in Sigma is quite simple w/o learning
  - Each link simply becomes a *rule* with the weight as its function
  - Extended GA's existing non-linear processing for sigmoid
- Can compress units at each layer (yielding speedups)
  - Yields one rule per layer, with weight matrix as function
- Implement *backpropagation* via "backward" rules
  - Use *correctness* to measure the error

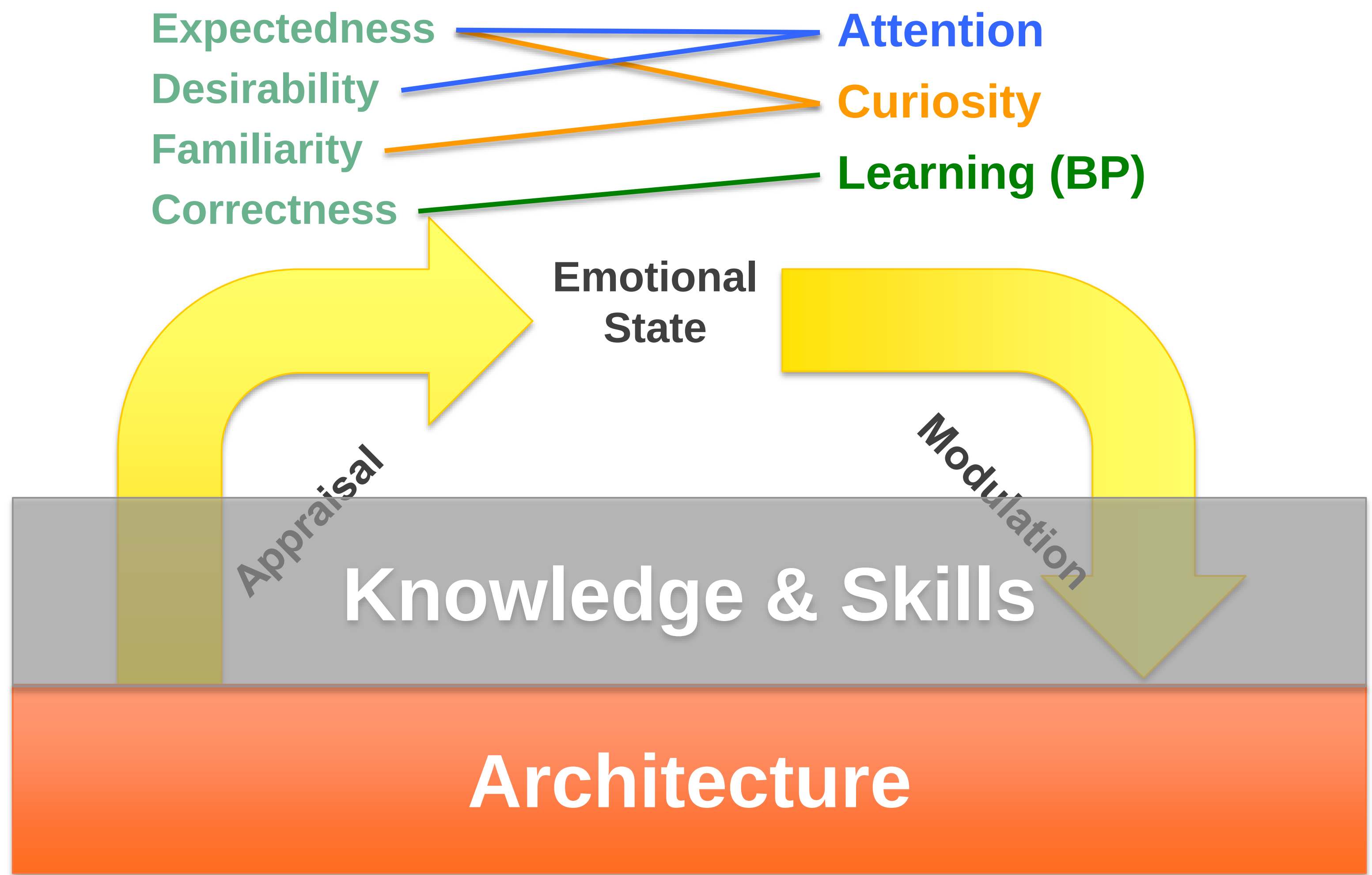


# Aside: Emotions in Sigma

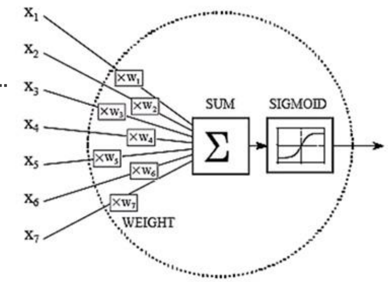
- Motivated by combination of:
  - Theoretical desiderata of *grand unification* and *generic cognition*
  - Practical goal of building useful *virtual humans*
  - Hypothesis that emotion is *critical for surviving and thriving in complex physical and social environments*
    - Part of the *wisdom of evolution*

- Largely non-voluntary and immutable
  - Likely a significant architectural component
- But also affected by knowledge and skills

# Appraisal-Driven Emotional Processing

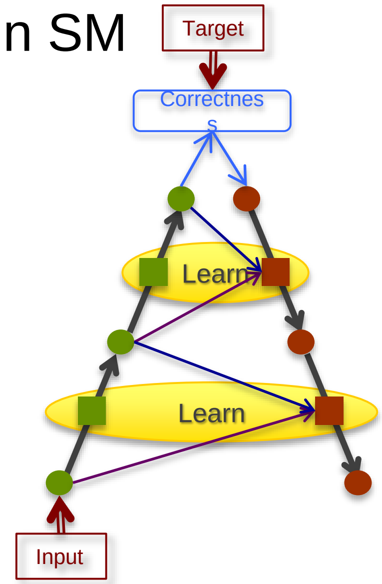


```
CONDITIONAL Layer-0
  Conditions: (Level-0 arg:v0)
  Actions: (Level-1 s arg:v1)
  Function: ws-0-1
```




# Example: Neural Network Learning in Sigma

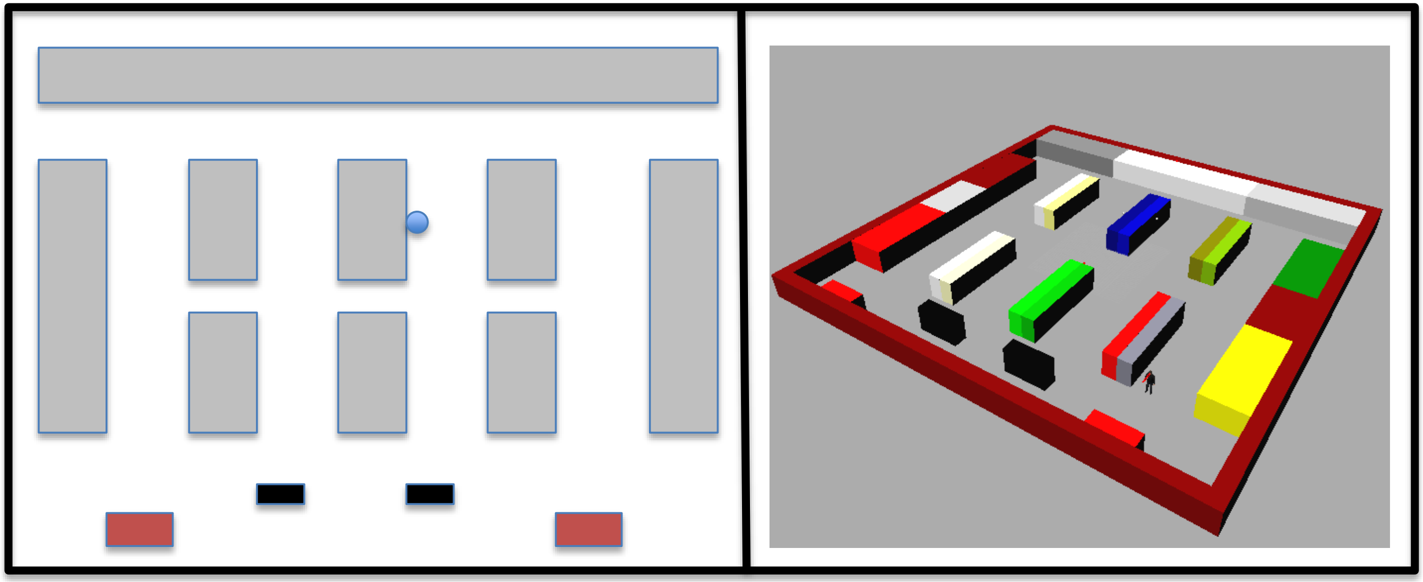
- Implementing NNs in Sigma is quite simple w/o learning
  - Each link simply becomes a *rule* with the weight as its function
  - Extended GA's existing non-linear processing for sigmoid
- Can compress units at each layer, yielding speedups as in SM
  - Yields one rule per layer, with weight matrix as function
- Implement *backpropagation* via "backward" rules
  - Use *correctness* to measure the error
  - *Tie* corresponding functions/messages together
  - Reuse forward messages as needed going backward
- Can replace standard GML in RL to yield *neural RL*
- Provides an architectural embedding for neural networks
  - Enables unification within a single graph of neural networks, probabilistic graphical models, symbolic rules, etc.
  - May provide guidance for how to combine deep learning with other necessary components, such as memories, search and attention



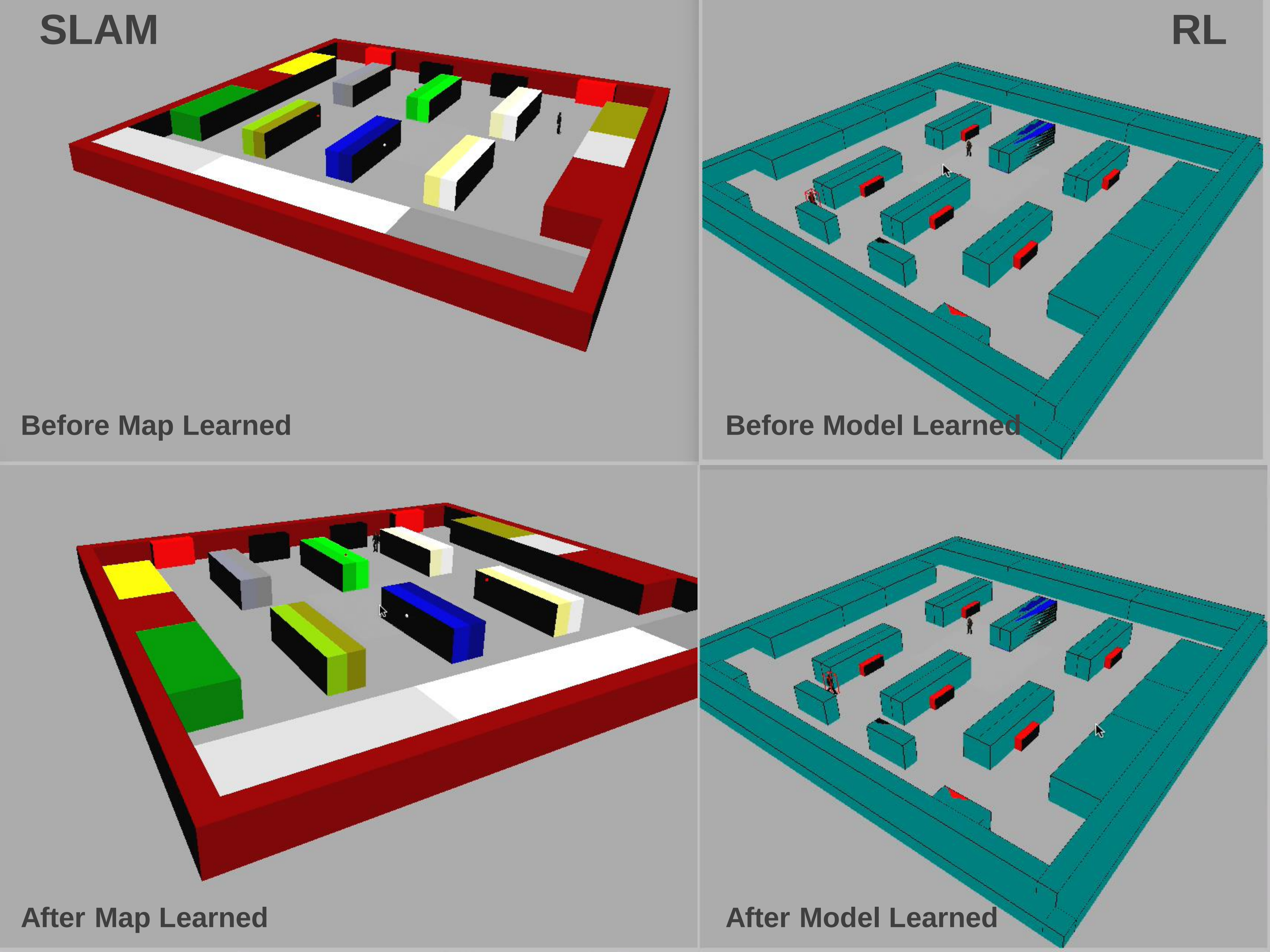
# INTEGRATION

# Example: Interactive, Adaptive Virtual Humans

- Control behavior of SmartBody VH(s) in a retail store scenario
  - A civilian instance of a *physical security system*



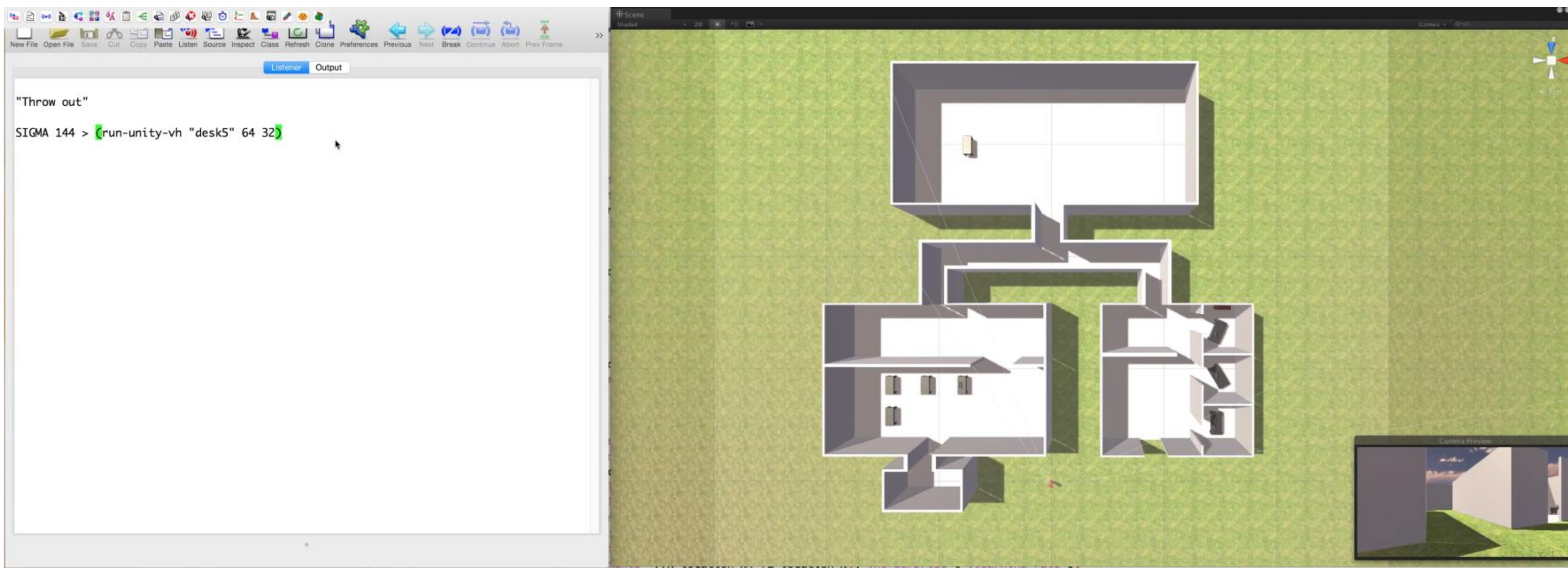
  - Rule-based, probabilistic and social reasoning (ToM)
  - Simultaneous localization and mapping (SLAM)
  - Multiagent reinforcement learning (RL)


SLAM
RL
Before Map Learned
Before Model Learned
After Map Learned
After Model Learned

# Example: Appraisal Based Exploration

Searching for an item only leveraging architectural *appraisal variables*

- Appraisal is the first stage of the full emotional arc
- Attention (surprise & desirability) and Curiosity (surprise & familiarity)



# Example: Conversational Virtual Human Mind

- Immersive Naval Officer Training System (INOTS)
  - Targets leadership and basic counseling for junior Navy leaders
  - Trained over 12,000 sailors since 2012


- INOTS “mind” based on two tools
  - Statistical query-answering tool (NPCEditor)
  - Transition diagram for dialogue management
- Both aspects reimplemented and integrated together in Sigma
  - Query answering via (naïve Bayes) semantic memory (*reactive*)
  - Dialogue management by sequences of operators (*deliberative*)
- Being extended to include speech via graphical models in Sigma

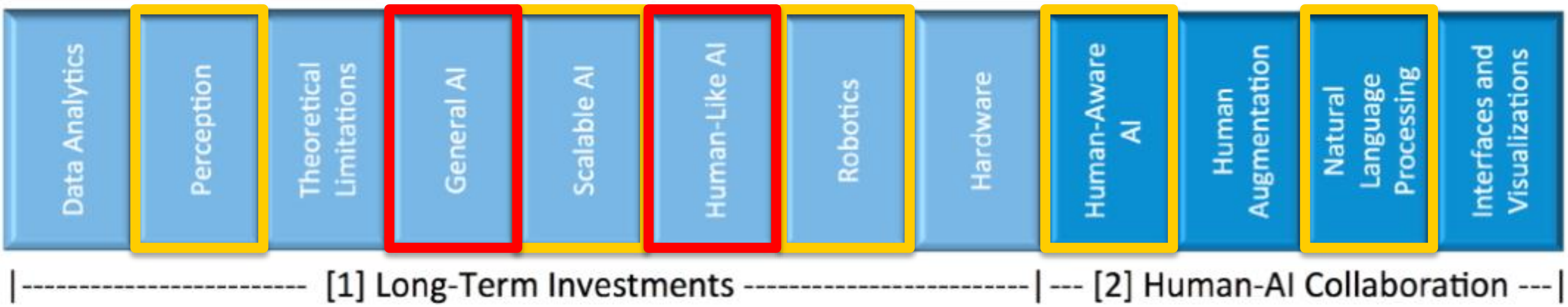
# CLOSING

# Fundamental Questions about Sigma

- **Can general intelligence be provided in this manner?**
- Can it all be sufficiently efficient for real time behavior?
- What are the functional gains?
- Can the human mind (and brain) be modeled?

# Conclusion

- Sigma embodies a new approach to cognitive architecture
  - Based on a broadly uniform, (largely) mathematically sound, and (potentially) efficient graphical architecture
  - For next generation virtual humans and intelligent agents/robots
- But does require some extensions to graphical models
- … and still has a ways to go for full general intelligence




*The National Artificial Intelligence Research and Development Strategic Plan* (Basic R&D segment of Fig. 4)

## JASON (1/17): Perspectives on Research in Artificial Intelligence and Artificial General Intelligence Relevant to DoD

3. The Deep Learning Revolution

5. Areas of Rapid Progress other than Deep Learning

5.1. Reinforcement Learning

**Sigma is an approach to combining all five!**

5.2. Graphical Models

5.3. Generative Models and Probabilistic Programming Languages

5.4. Hybrid Architectures

Selected Recommendations from report:

DoD should both track (via a knowledgeable cadre) and invest in (via a 6.1 research portfolio) the most dynamic and rapidly advancing areas of AI, including, but by no means limited to DL.

DoD's portfolio in AGI should be modest and recognize that it is not currently a rapidly advancing area of AI. …

But can AGI based on these five concepts advance more rapidly?

SYSTEMS ENGINEERING RESEARCH CENTER

SERC TALKS

# UPCOMING TOPICS:

Cyber-Physical Learning Systems

***What Are Cyber-Social Learning Systems And How Will We Form Them?***

Dr. Kevin Sullivan, University of Virginia

**June 7, 2017 | 1:00 pm ET**

Cybersecurity

Dr. Barry Horowitz, University of Virginia

Munster Professor of Systems and Information Engineering and Chair

**August 2, 2017 | 1:00 pm ET**

**Thank you for joining us!**

Please check back on the SERC website for today's recording and future SERC Talks information!