Soar: An Architecture for Human Behavior Representation

Randall W. Hill, Jr.
Information Sciences Institute
University of Southern California
http://www.isi.edu/soar/hill

What is Soar?

- Artificial Intelligence Architecture
  - System for building intelligent agents
  - Learning system

- Cognitive Architecture
  - A candidate Unified Theory of Cognition (Allen Newell, 1990)

History

- Inventors
  - Allen Newell, John Laird, Paul Rosenbloom
- Officially created in 1983
  - Roots in 1950's and onwards
- Currently on version 8 of Soar architecture
  - Written in ANSI C for portability and speed
- In the public domain

User Community

- Academia
  – USC, U. of Michigan, CMU, BYU, others
- International
  – Britain, Europe, Japan
- Commercial
  – Soar Technology, Inc.
  – ExpLore Reasoning Systems, Inc.

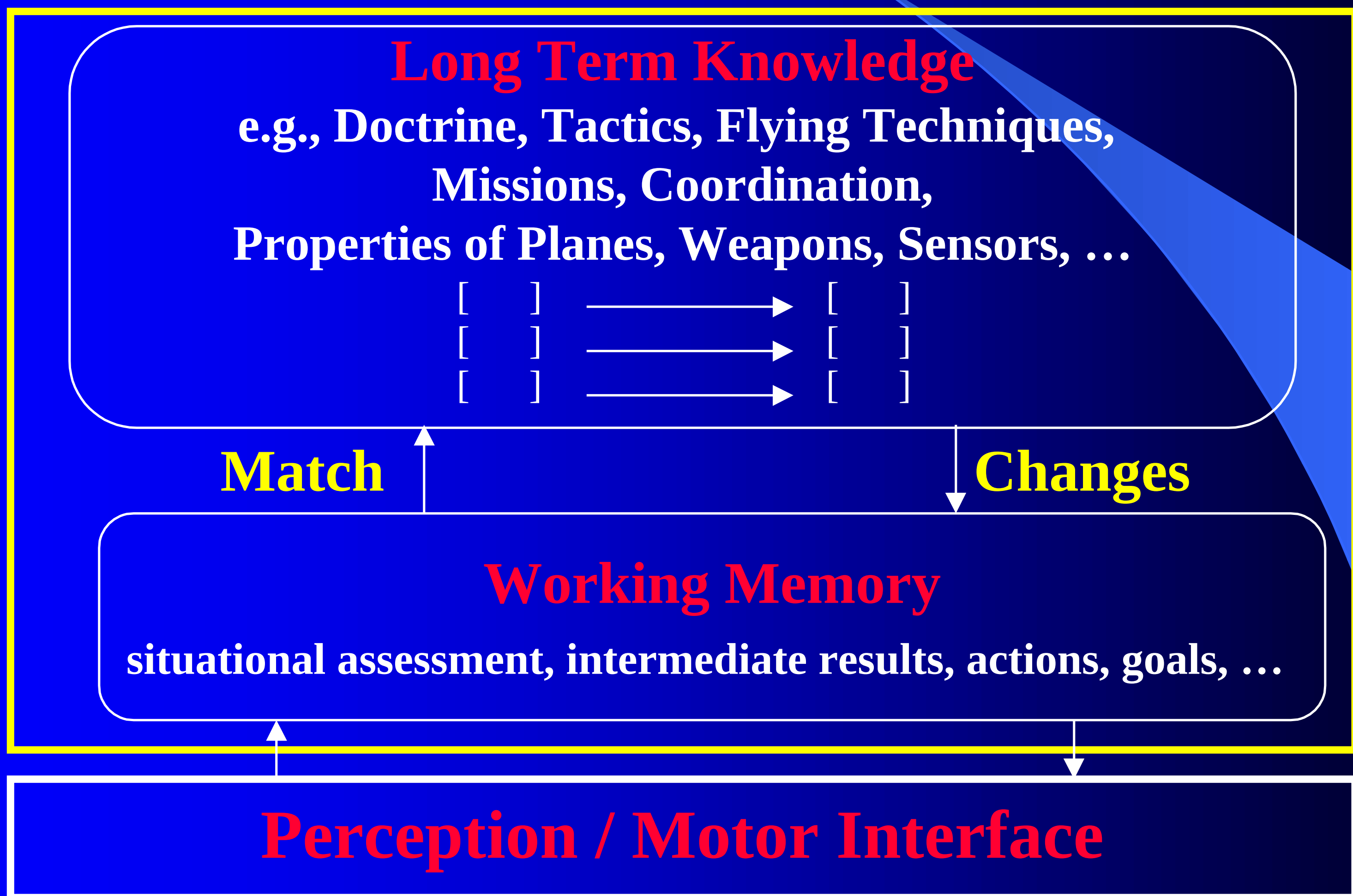
Objectives of Architecture

- Support multi-method problem solving
  - Apply to a wide variety of tasks and methods
  - Combine reactive and goal directed symbolic processing
- Represent and use multiple knowledge forms
  - Procedural, declarative, episodic, iconic
  - Support very large bodies of knowledge (>100,000 rules)
- Interact with the outside world
- Learn about all aspects of tasks

Cognitive Behavior: Underlying Assumptions

- Goal-oriented
- Reactive
- Requires use of symbols
- Problem space hypothesis
- Requires learning

Soar Architecture



Soar Decision Cycle

Perception | *Cognition* | *Motor*

Elaboration Phase

- Fire rules
- Generate preferences
- Update working memory

Input Phase

- Sense world
- Perceptual pre-processing
- Assert to WM

Output Phase

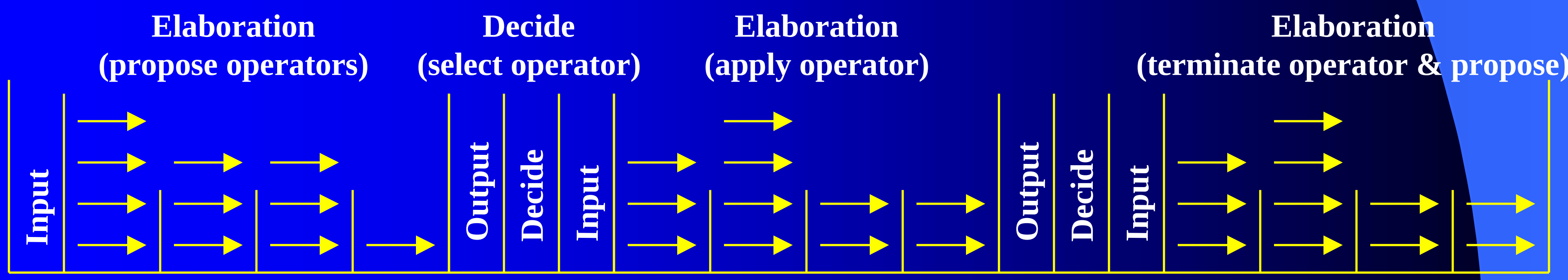
- Command effectors
- Adjust perception

Decision Phase

- Evaluate operator preferences
- Select new operator OR
- Create new state

Which Rule(s) Should Fire?

- Fire all matched rules in parallel until quiescence
- Sequential operators generate behavior
  - e.g., Turn, adjust-radar, select-missile, climb
  - Provides trace of behavior comparable to human actions
- Rules select, apply, terminate *operators*.
  - Select: create preferences to propose and compare operators
  - Apply: modify the current situation, send motor commands
  - Terminate: determine that operator is finished

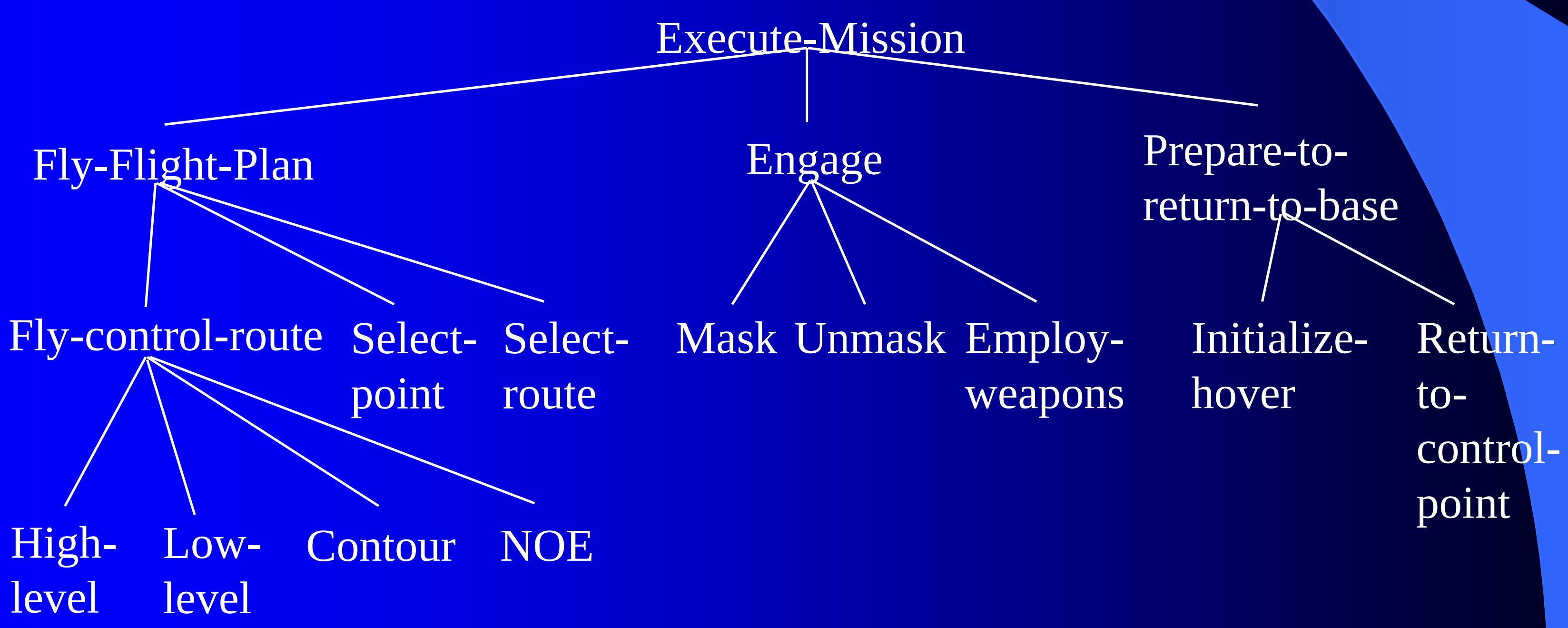


Example Rules

- **PROPOSE:** If I encounter the enemy, propose an operator to break contact with the enemy.
- **SELECT:** If I am enroute to my holding area and I come into contact with an enemy unit, prefer breaking contact over engaging targets.
- **APPLY:** If the enemy is to my left, break to the right.
- **APPLY:** If the enemy is to my right, break to the left.
- **TERMINATE:** If break contact is the current operator, and contact is broken, then terminate break operator.

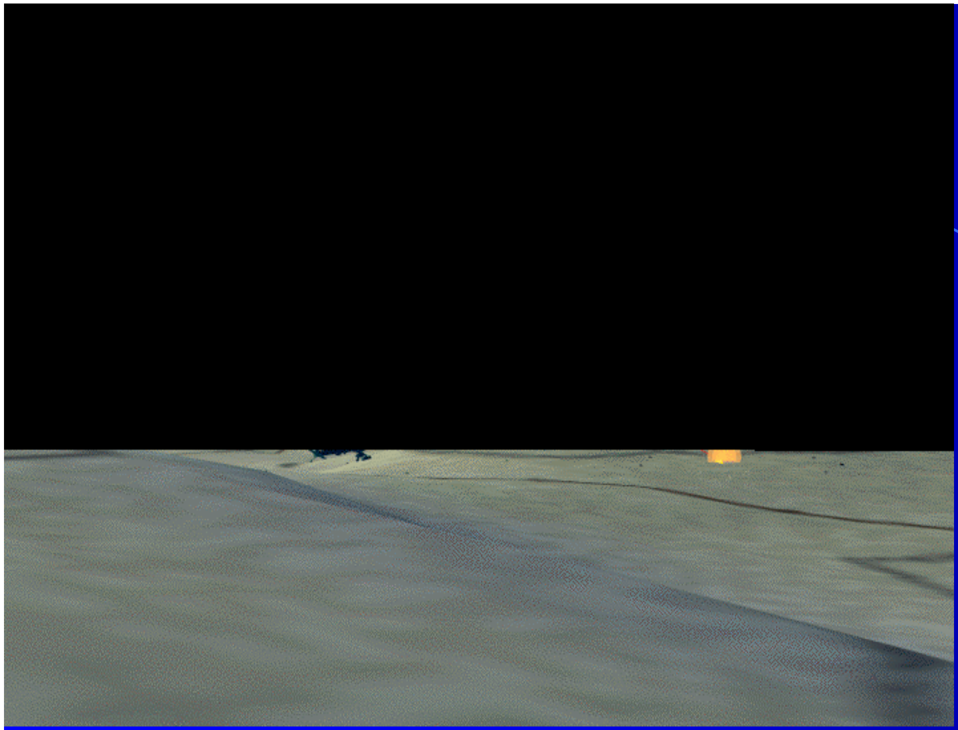
Goal Driven Behavior

- Complex operators are decomposed to simpler ones
  - Occurs whenever rules are insufficient to apply operator
  - Decomposition is dynamic and situation dependent
  - Over 90 operators in RWA-Soar



Coordination of Behavior & Action

- Combines goal-driven and reactive behaviors
  - Suggest new operators anywhere in goal hierarchy
  - Generate preferences for operators
  - Terminate operators
- Provides limited multi-task capability
  - Constrained by single goal hierarchy in Soar
- Other possible approaches
  - Multiple goal hierarchies
  - Flush and re-build goal hierarchies when needed



Modeling Perceptual Attention

Problem

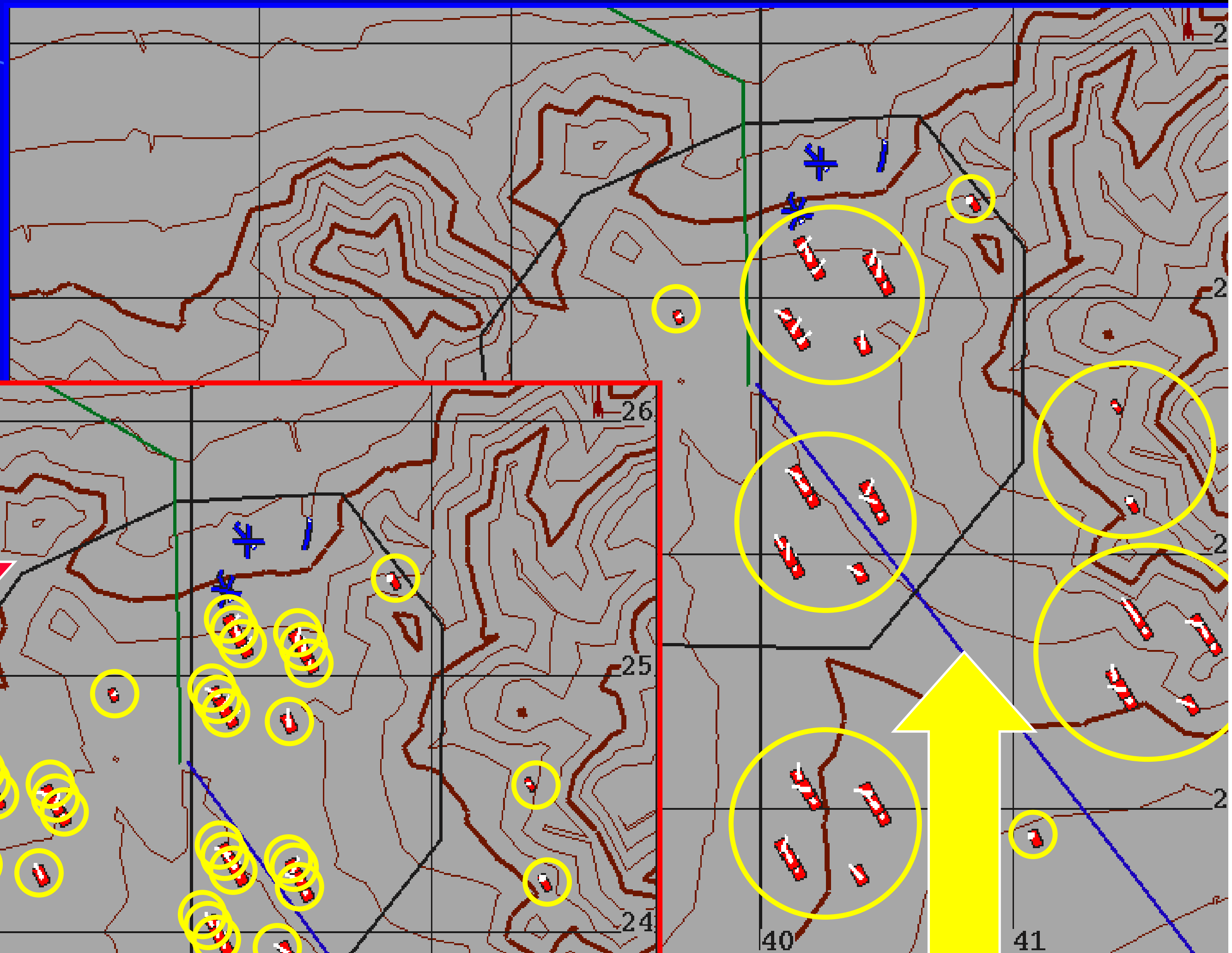
- Naïve vision model
  - — Entity-level resolution
  - — Unrealistic field of view (360^{o}, 7 km)
- No focus of attention
  - — Perceptual overload often occurs
  - — Pilot crashes helicopter

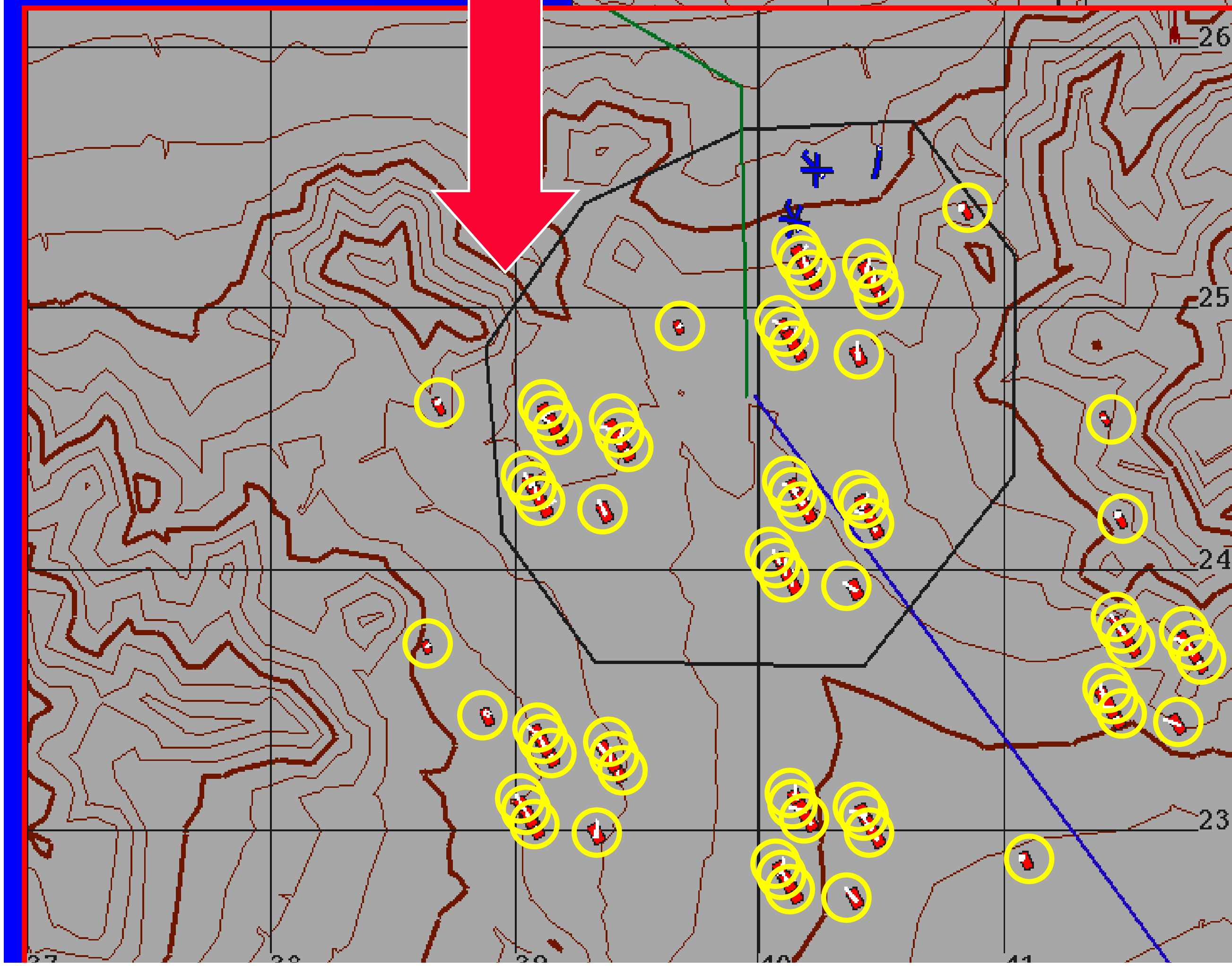
Approach

- Zoom lens model of attention
  - — Gestalt grouping in pre-attentive stage
  - — Multi-resolution focus
- Control of attention
  - — Goal-driven: task-based, group-oriented
  - — Stimulus-driven: abrupt onset, contrast

Naïve Vision Model

- Entity-oriented
- Stimulus-driven
- No perceptual control





Model of Attention

- Gestalt grouping
- Zoom lens effect
- Goal and stimulus driven

Underlying Technologies/Algorithms

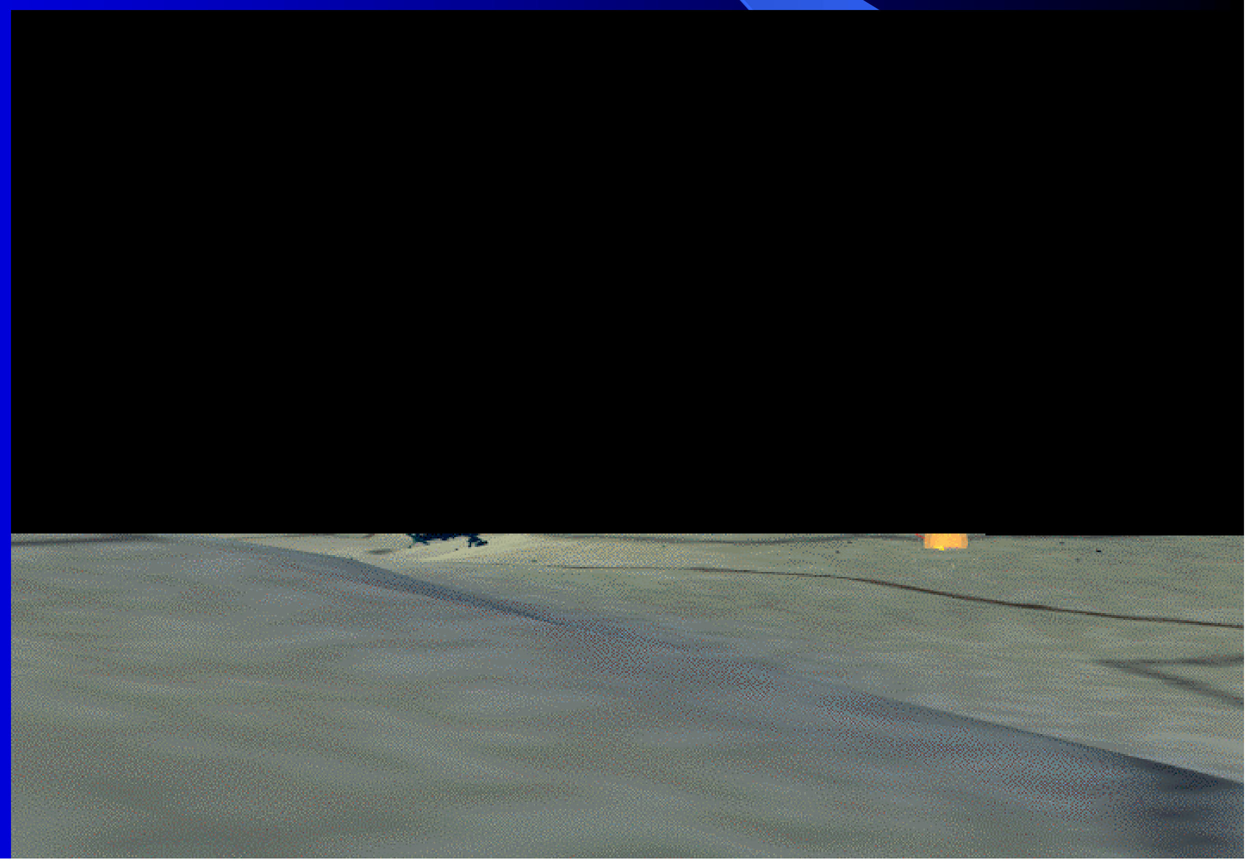
- Optimized RETE algorithm
  - Enables efficient matching in large rule sets
- Universal subgoaling
  - Operator-based architecture
  - Truth Maintenance System (TMS)
- Learning algorithm
  - Chunking mechanism

Soar Applications

- Agents for Synthetic Battlespaces
  - Commanders and Helicopter Pilots (USC)
  - Fixed Wing Aircraft Pilots (UM, Soar Technology)
- Animated, Pedagogical Agents
  - Steve (Rickel and Johnson, USC)
- Game Agents
  - Quake (Laird and van Lent, UM)

Intelligent Synthetic Forces

- Helicopter pilots
- Teamwork
- C3I Modeling
  - Planning
  - Execution
  - Re-planning
  - Collaboration

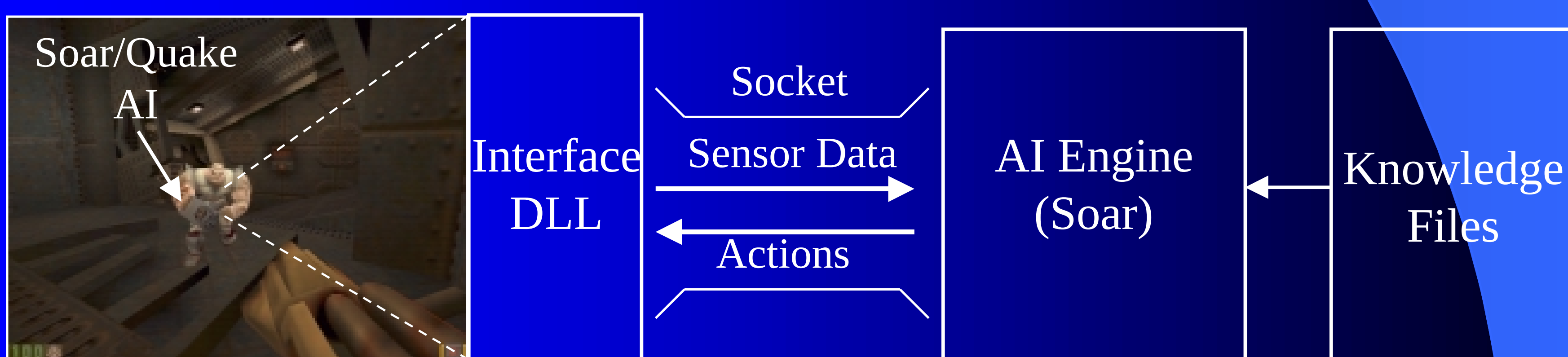


Steve: An Embodied Intelligent Agent for Virtual Environments

- 3D agent that interacts with students in virtual environments
- Can take different roles: teammate, teacher, guide, demonstrator
- Multiple trainees and agents work together in virtual teams
- Intelligent tutoring in the context of a shared team environment

Soar/Games Project

- Build an AI Engine around the Soar AI architecture
  - Soar/Quake II project
  - Soar/Descent 3 project
- U. of Michigan, Laird and van Lent



Validation Efforts

- Intelligent Synthetic Forces
  - Synthetic Theater of War '97 experience
  - Subject Matter Experts
- Human Factors / HCI studies
  - e.g., B. John (CMU) & R. Young (U.K.)
- Better models for validating integrated models of human behavior needed

Summary of Capabilities/Limitations

- Capabilities
  - Mixes goal-oriented and reactive behavior
  - Supports interaction with external world
  - Architecture lends itself to creating integrated models of human behavior
- Limitations
  - Learning mechanism not easily used

Future Development / Application Plans

- Integrate emotion with cognition
- Learn from experience
  - Incorporate inductive models of learning
- Continue work on models of collaboration in complex decision-making
  - Extend the current C3I models