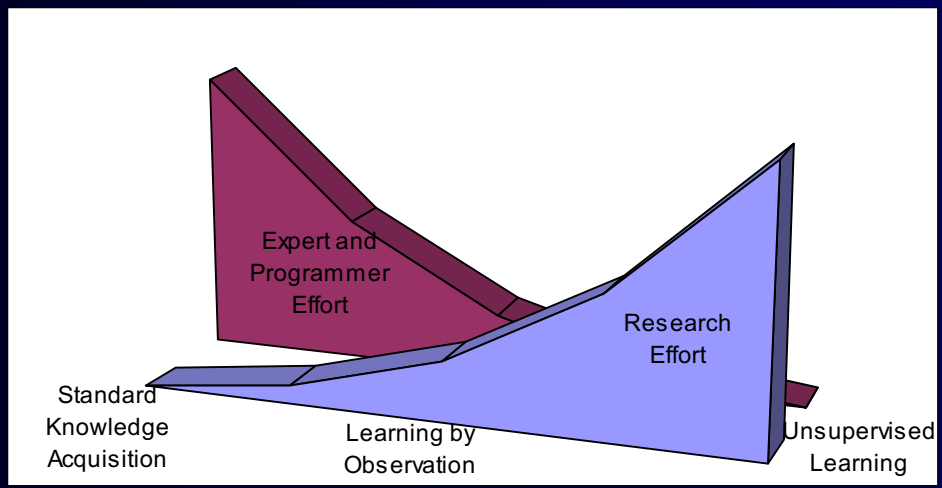
Learning Task Performance Knowledge by Observation

Mike van Lent

Marina del Rey
Saturday, May 13, 2000

Motivation

- Task performance agents are becoming common
  - Training simulations, Computer games
- Knowledge acquisition for these agents is expensive
  - 10 man/years for TacAir-Soar
- There is a continuum of KA approaches



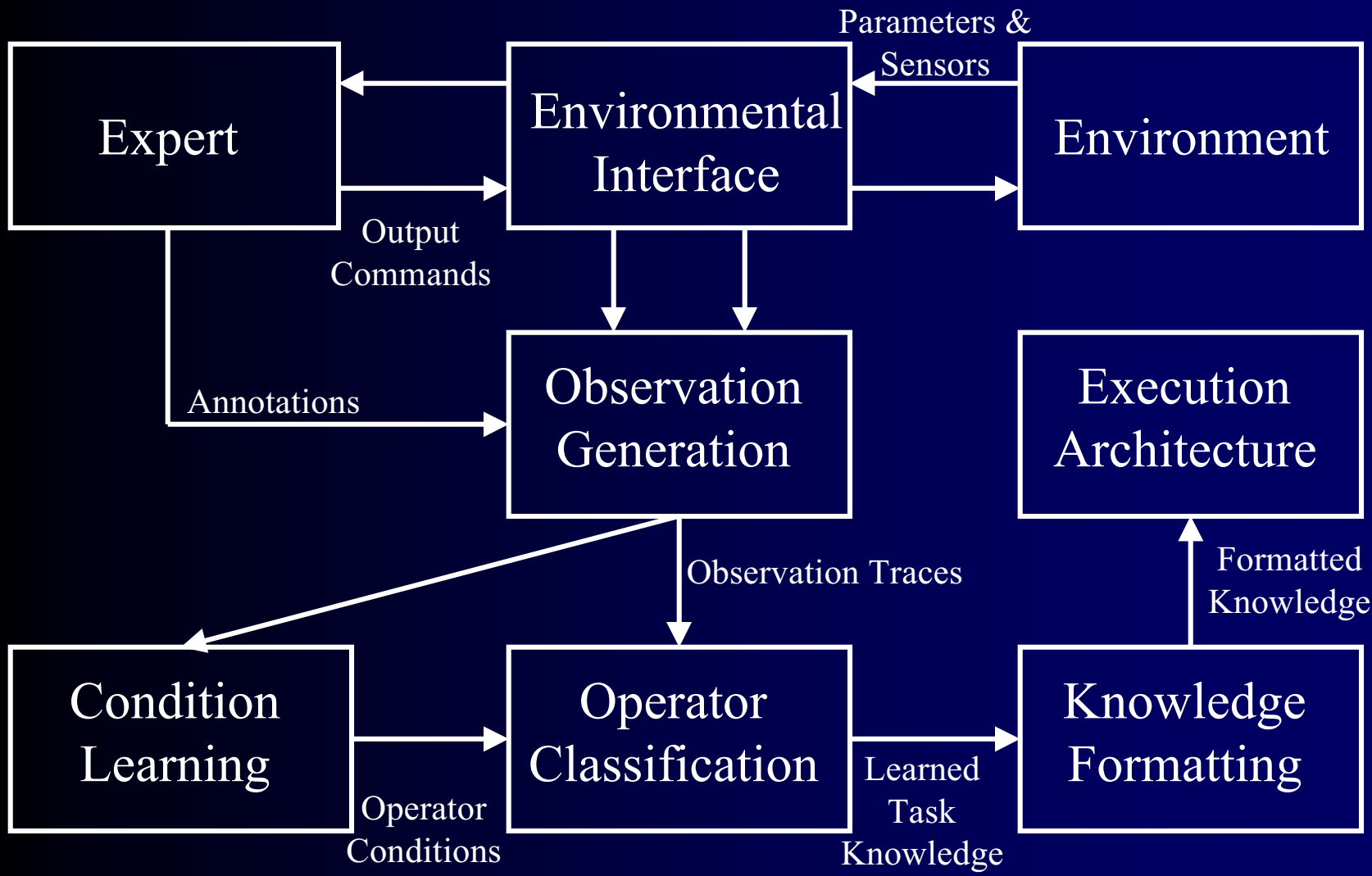
Observation as a knowledge source

- Doesn't require expert to make knowledge explicit
  - Unlike standard KA, automated KA and instruction
- Doesn't require expert to learn new tools
  - Unlike automated KA
- Doesn't require knowledge engineer to learn task
  - Unlike standard KA
- Results in knowledge that matches human behavior
  - Unlike unsupervised learning
- Doesn't require any initial knowledge
  - Unlike instruction
- Rich and focused knowledge source

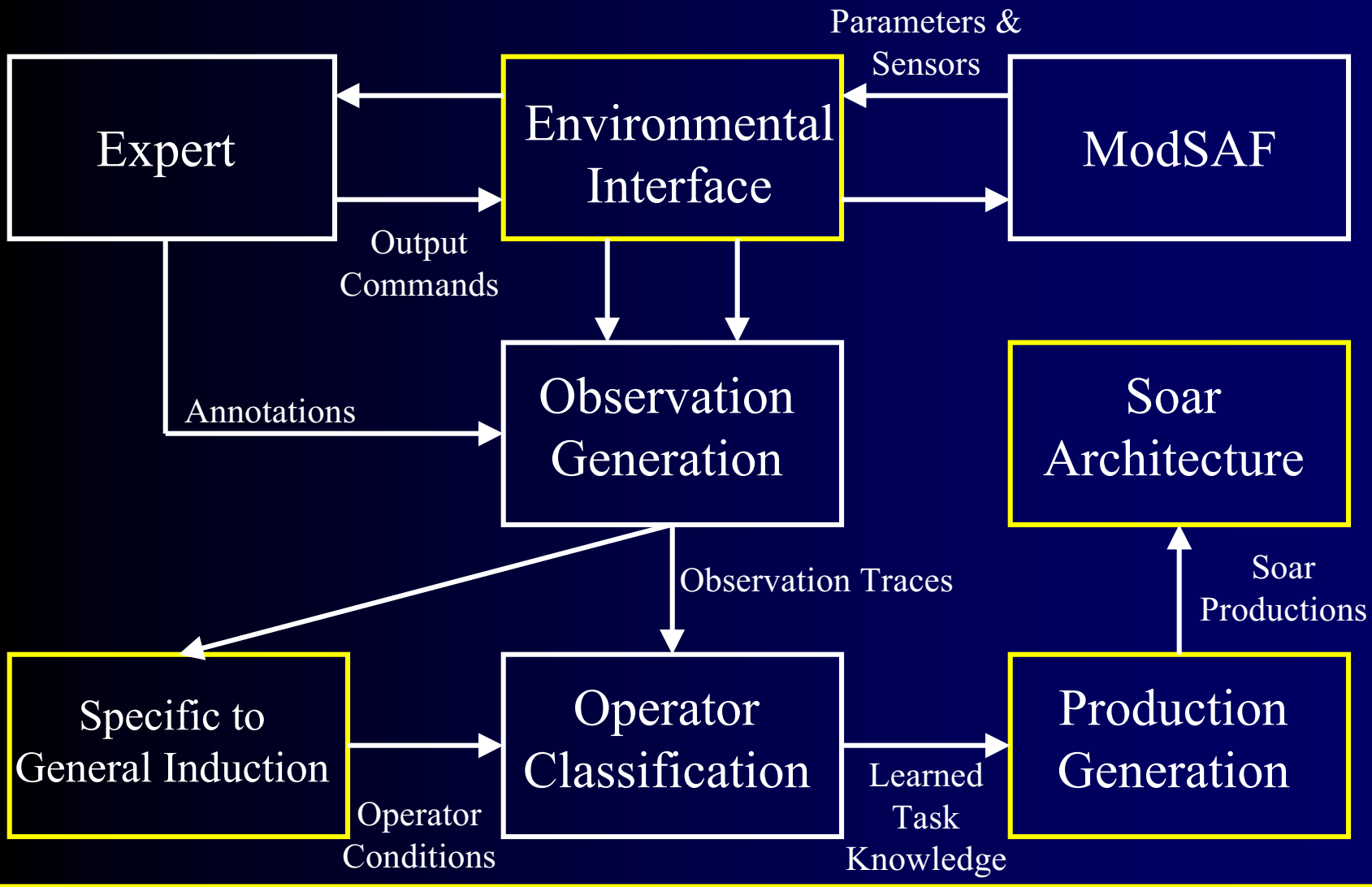
Task Knowledge Representation

- Based on Soar's knowledge representation
- Operator Hierarchy
- Operator consists of:
  - Pre-conditions (potentially disjunctive)
    - Includes negated test for goal-achieved feature
  - Conditional Actions
    - Action attribute and value (pass-through action values)
  - Goal conditions (potentially disjunctive)
    - Create goal-achieved feature
    - Persistent and non-persistent goal-achieved features
- Task and Domain parameters are widely used to generalize the learned task knowledge

L-by-O Framework



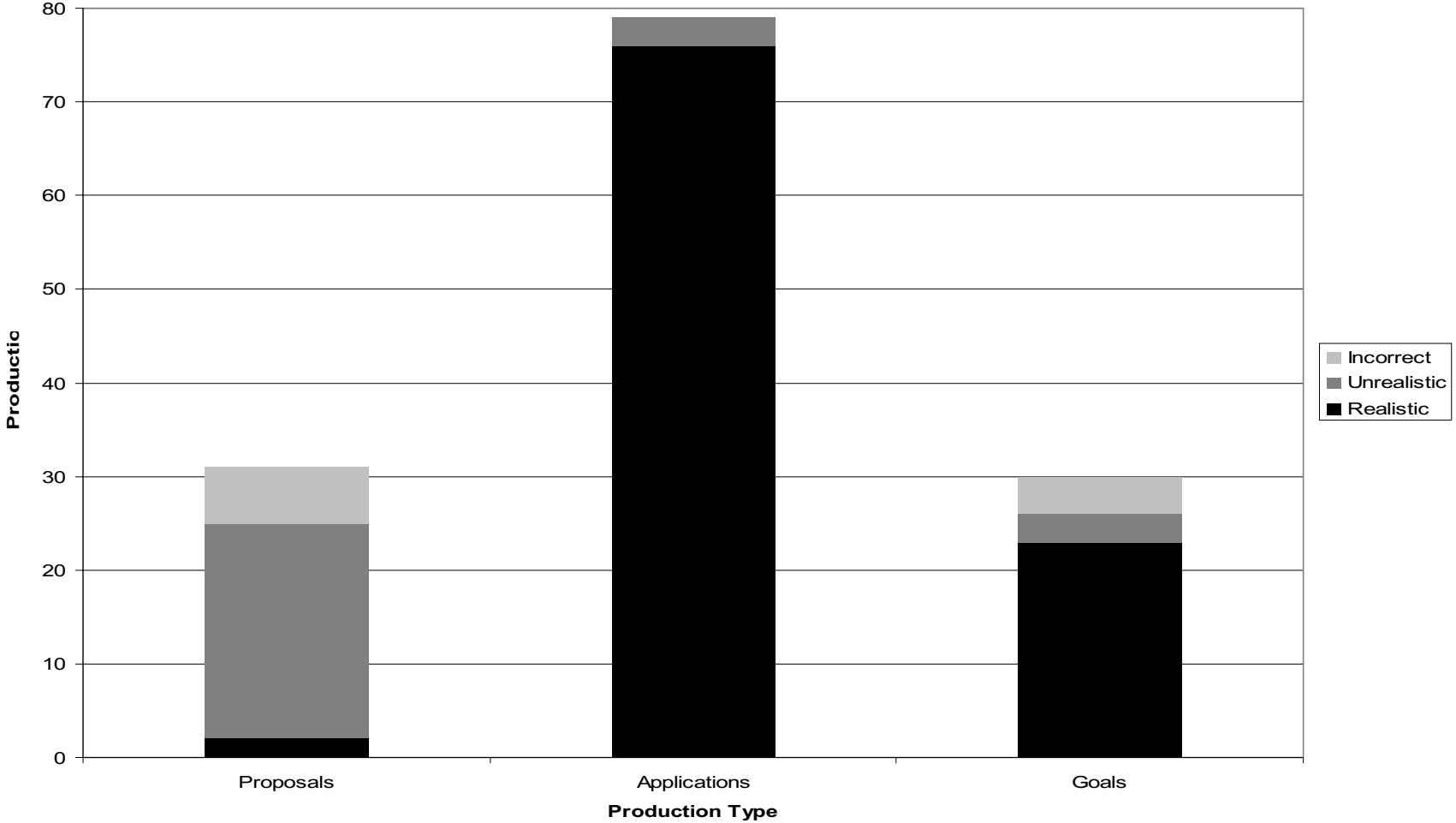
KnoMic


Parameters &
Sensors
Expert
Environmental
Interface
ModSAF
Output
Commands
Annotations
Observation
Generation
Soar
Architecture
Observation Traces
Soar
Productions
Specific to
General Induction
Operator
Classification
Production
Generation
Operator
Conditions
Learned
Task
Knowledge

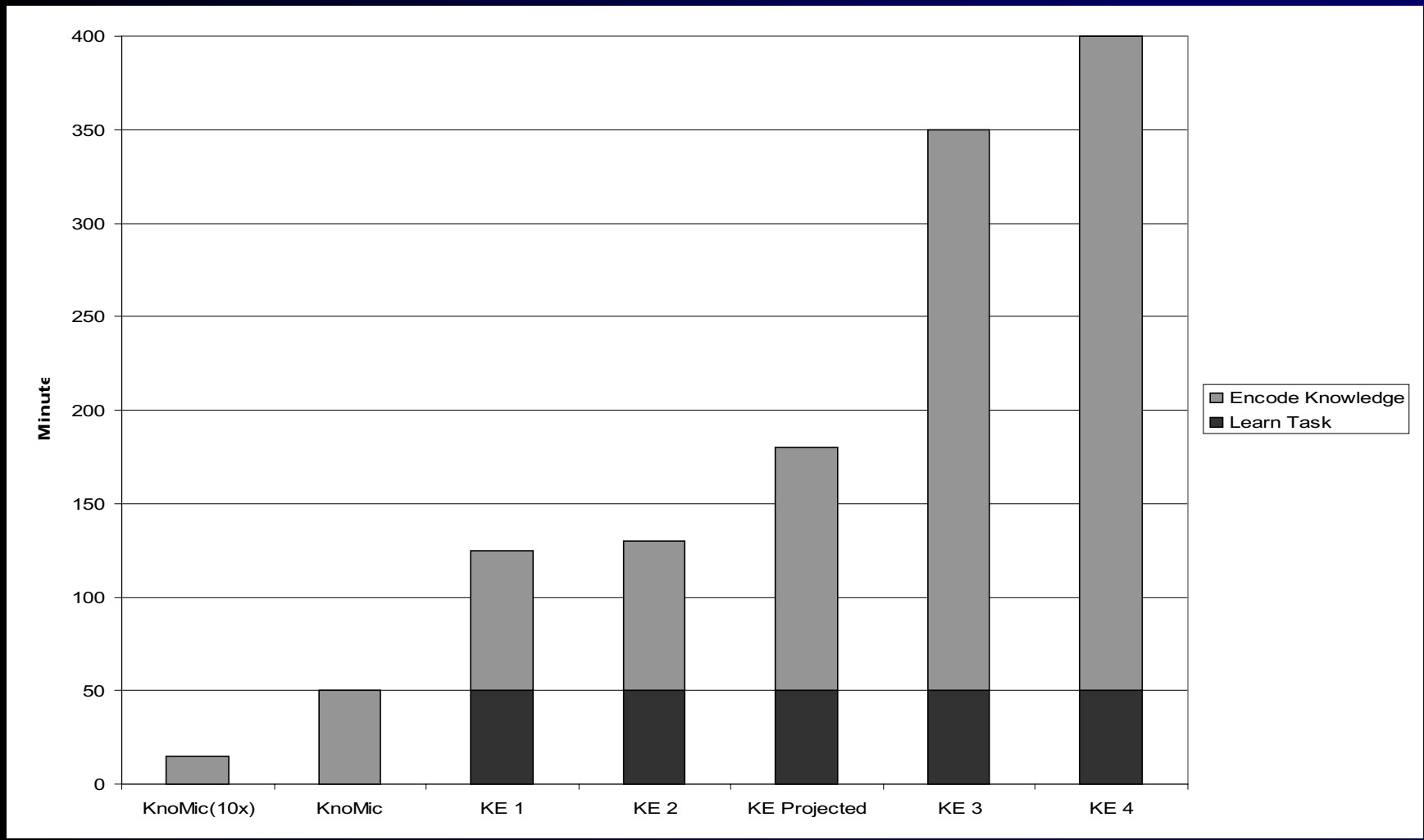
Experiments

- Evaluation Criteria
  - Coverage of learned task knowledge
  - Realism of learned task knowledge
  - Efficiency of learning by observation vs. standard KA
- Experiment 1: TacAir-Soar
  - Main domain for evaluating KnoMic
- Experiment 2: Quake II-Soar
  - Framework approach makes supporting new domains easy
- Task Analysis: Towers of Hanoi
  - Example of an internal feature that can't be learned
  - Solutions: Instruction, Historical sensors,...

Realism


80
70
60
50
40
30
20
10
0
Productio
Proposals
Applications
Goals
Production Type
Incorrect
Unrealistic
Realistic

Efficiency


400
350
300
250
200
150
100
50
0
Minute
KnoMic(10x)
KnoMic
KE 1
KE 2
KE Projected
KE 3
KE 4
Encode Knowledge
Learn Task

Contributions

- Learning-by-observation framework
  - Defines data structures and flow of information
  - Allows many different learning-by-observation systems
- Criteria for evaluation of KA systems
  - Coverage, Realism, Efficiency
- Evaluation of KnoMic (a learning-by-observation sys.)
  - Three domains
- Taxonomy of possible errors in an observation trace
- Exploration of one approach to combining observation and instruction

Future Work

- Explore the efficiency of different learning algorithms
  - Negative Instances
- Explore approaches to learning internal features
  - Initial domain knowledge
  - Historical sensors
- Combine instruction and observation
  - Two stage learning process
- Eliminate the need for operator annotations
  - Automatic observation trace segmentation
- Knowledge acquisition tool for the Quake II domain
  - Based on the learning by observation framework

Nuggets and Coal

- Nuggets
  - Thesis Defense: May 22 at 10:00am
  - KnoMic is able to learn most of the task knowledge for a complex domain
  - What can and can't be learned is well understood
  - Learning-by-observation framework creates lots of exciting ideas for future research
  - Observation and Instruction complement each other nicely
- Coal
  - Internal features are necessary for some domains
  - Observation isn't a magic bullet
    - A useful knowledge source to use in conjunction with other sources