Encoding Intelligent Agents for Uncertain, Unknown, and Dynamic Tasks:

From Programming to Interactive Artificial Learning

Jacob W. Crandall

Michael A. Goodrich
and Lanny Lin

AAAI Spring Symposium 2009
Agents that Learn from Human Teachers

March 24, 2009

Research Problem

- Goal:
  End-users can “program” their own machines to do “intelligent” things

Research Problem

- Goal:
  End-users can "program" their own machines to do "intelligent" things

- Challenges
  - System designers have **limited *a priori* knowledge**
  - End-users may **not** be **technology experts**

Methods

1. Traditional AI Programming

2. Classical Artificial Learning

3. Interactive Artificial Learning (**IAL**)

Methods

1. Traditional AI Programming

2. Classical Artificial Learning

3. Interactive Artificial Learning (**IAL**)

Return on User Investment

$$\frac{\text{Agent Competence}}{\text{User Input}}$$

Methods

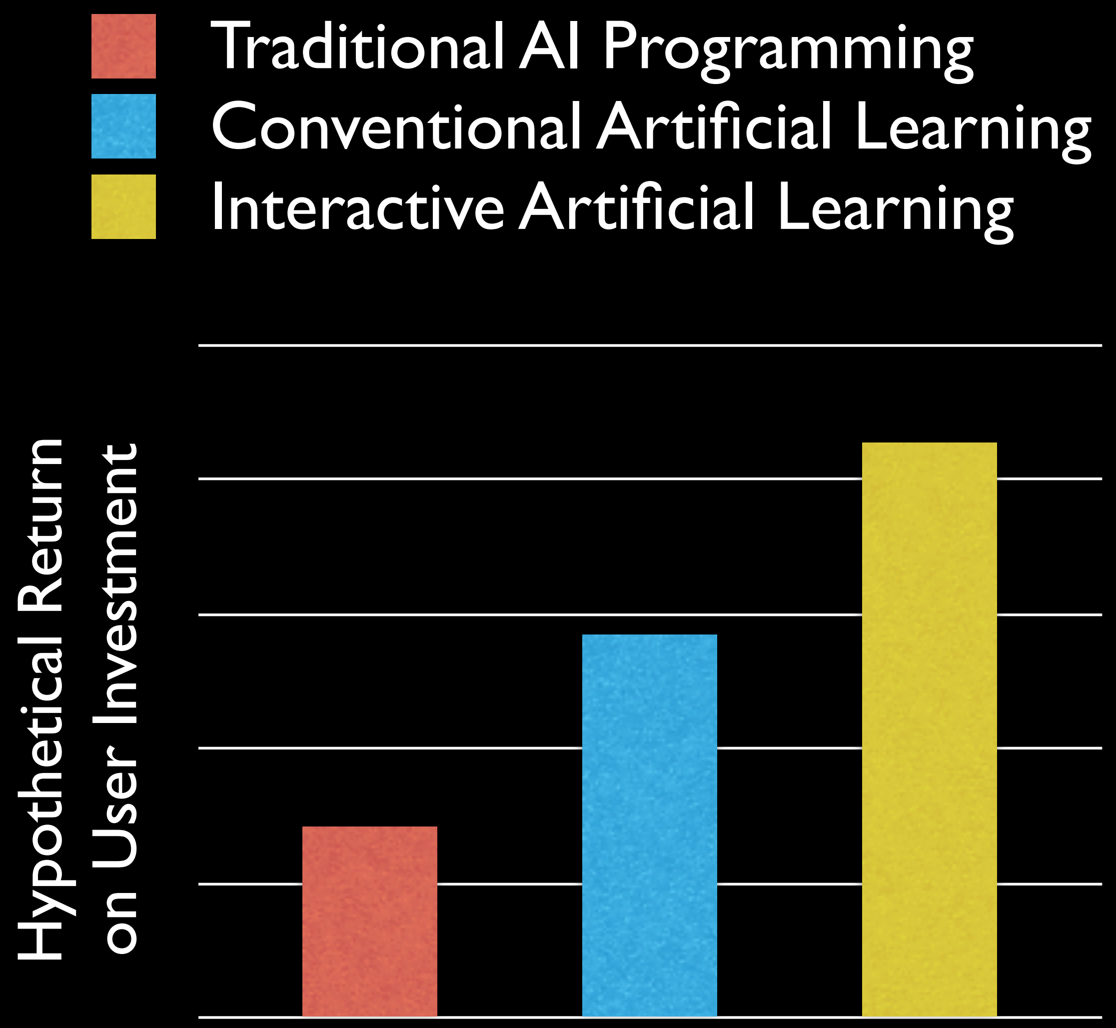
1. Traditional AI Programming

2. Classical Artificial Learning

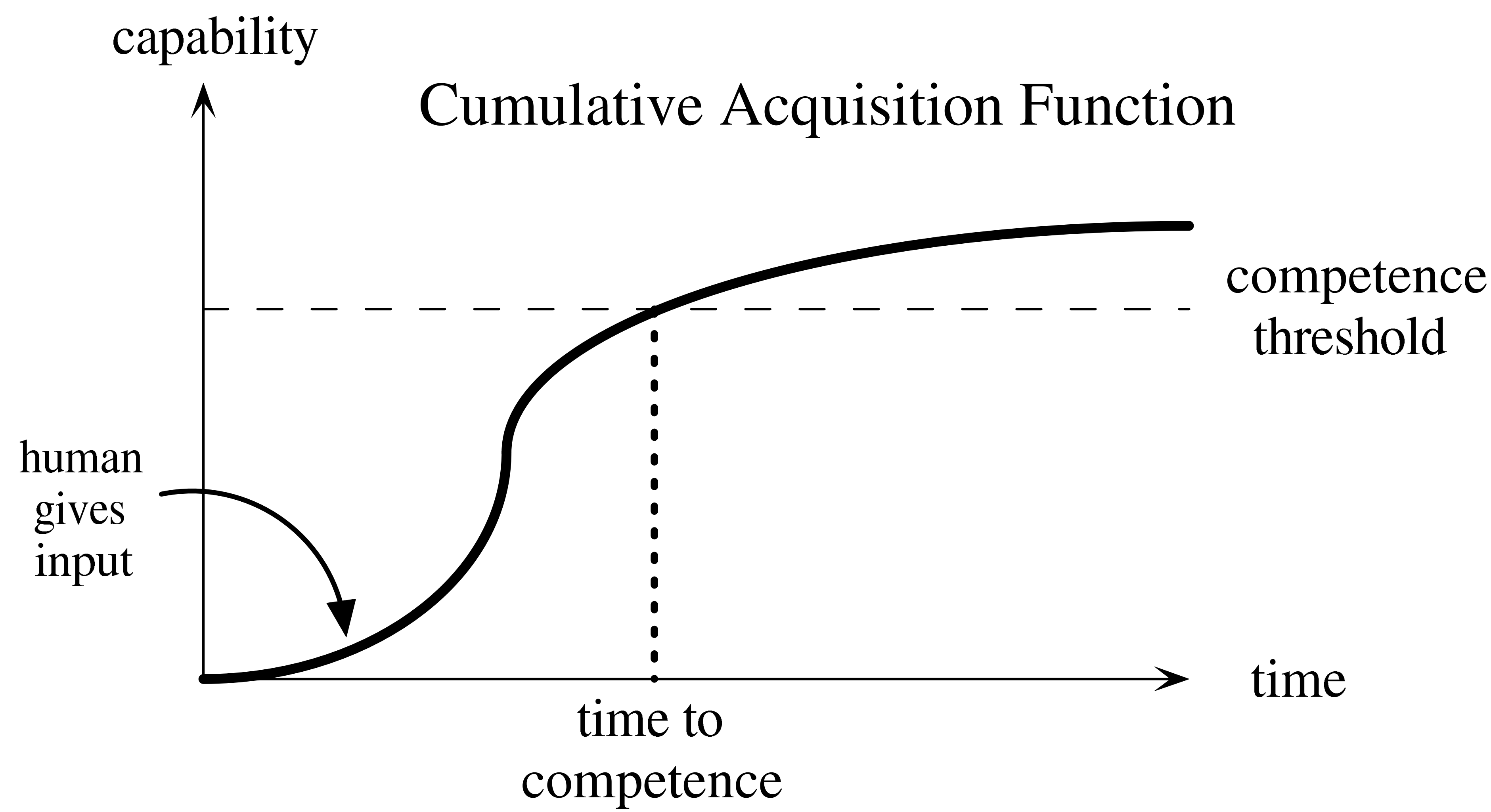
3. Interactive Artificial Learning (**IAL**)

Return on User Investment

$$\frac{\text{Agent Competence}}{\text{User Input}}$$

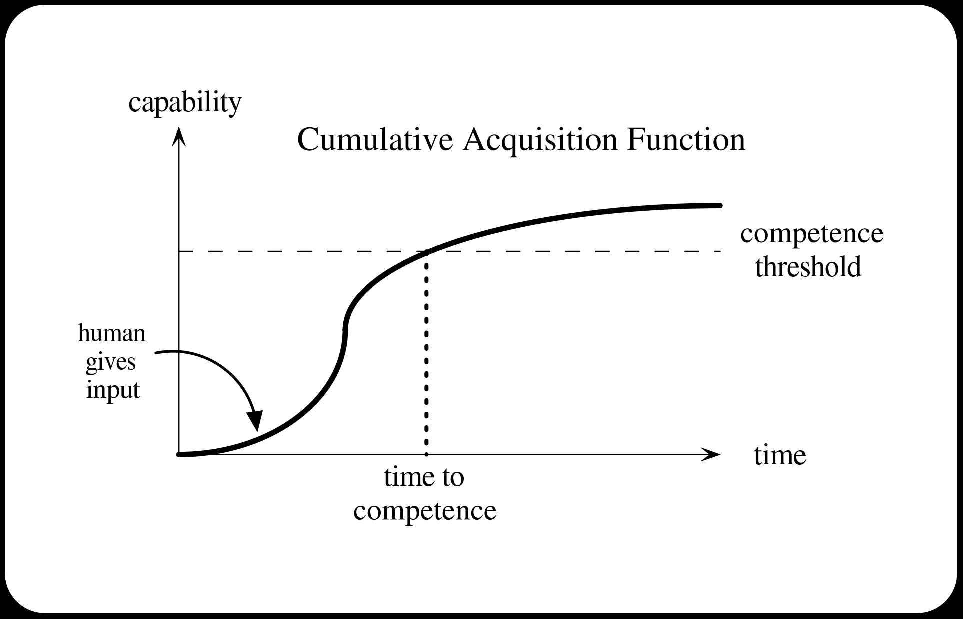


Metrics for IAL



3 Questions for IAL

- What **roles** should humans play and **when** do they get involved?
- How should systems be designed to support these **human-machine interactions**?
- What **learning algorithms** are most appropriate?

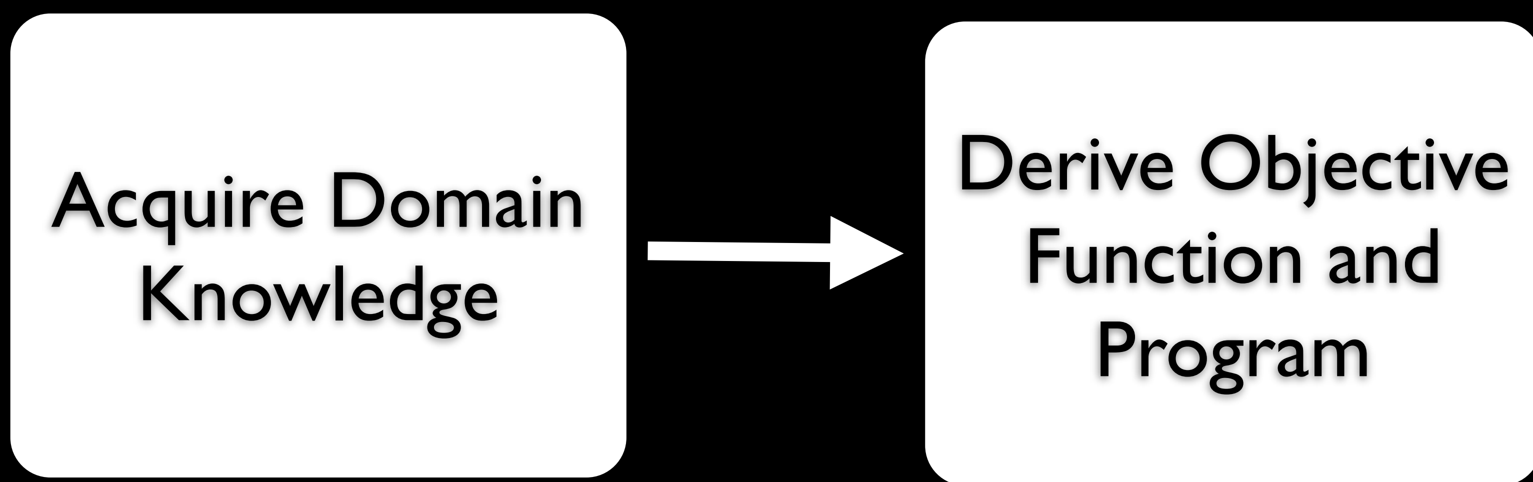


Traditional AI Programming

Traditional AI Programming

Acquire Domain Knowledge

Traditional AI Programming

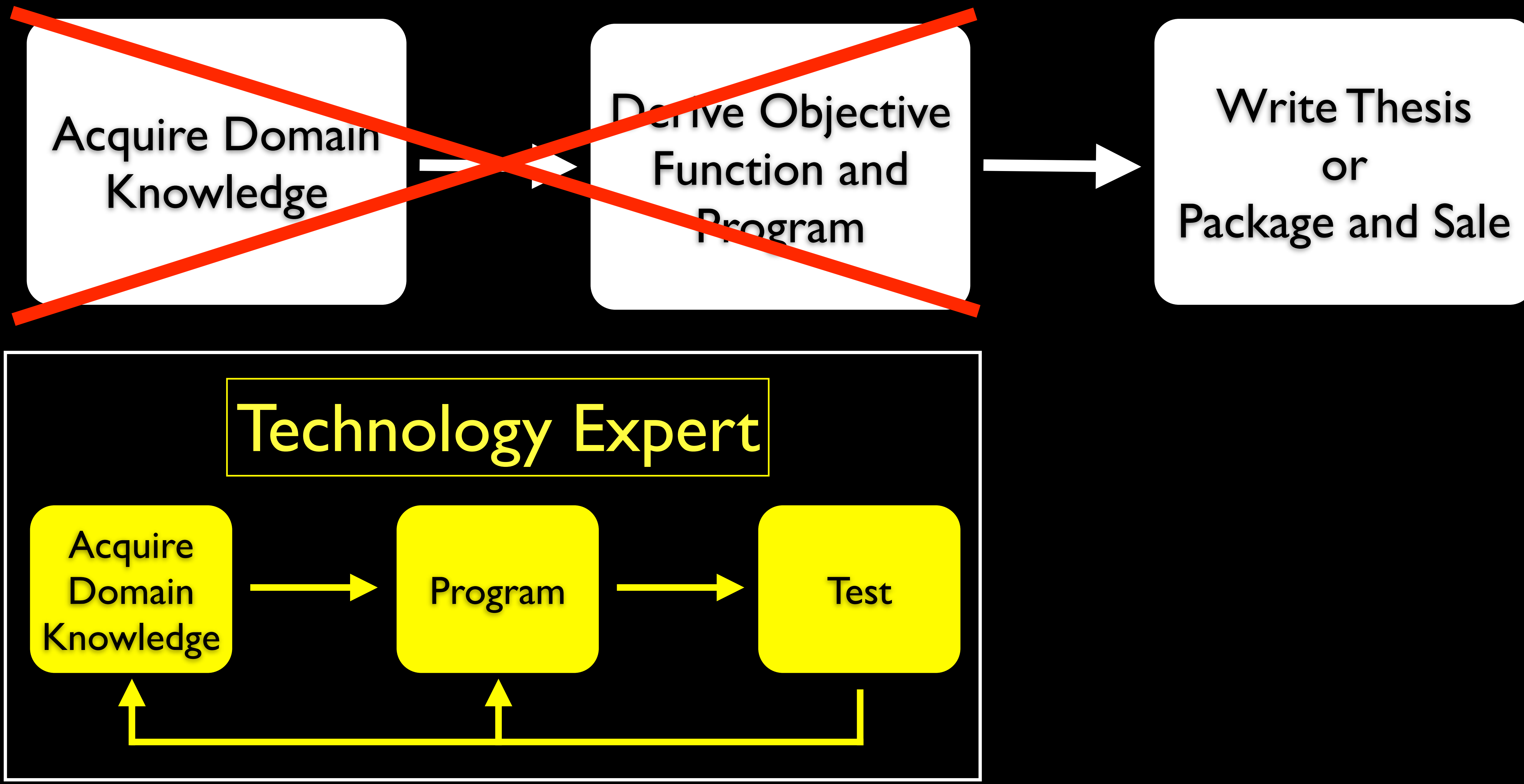


Traditional AI Programming

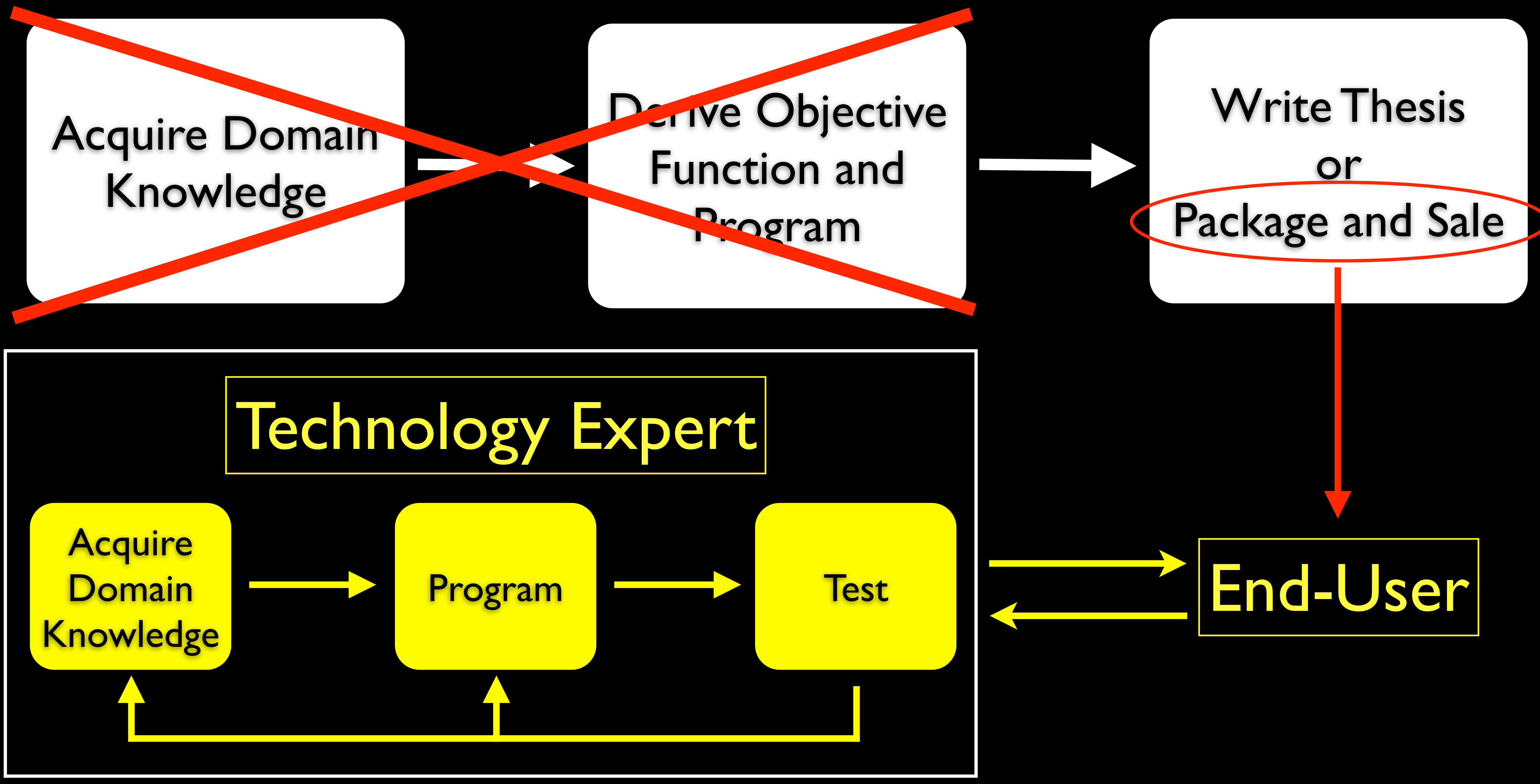
Acquire Domain Knowledge → Derive Objective Function and Program → Write Thesis or Package and Sale

Traditional AI Programming

Traditional AI Programming



Traditional AI Programming



Classical Artificial Learning

Classical Artificial Learning

Pre-configure algorithm

Classical Artificial Learning

Pre-configure algorithm

Technology expert specifies:
- States and Features
- Reward representation
- Learning representations
- Parameter values
- Etc.

Classical Artificial Learning

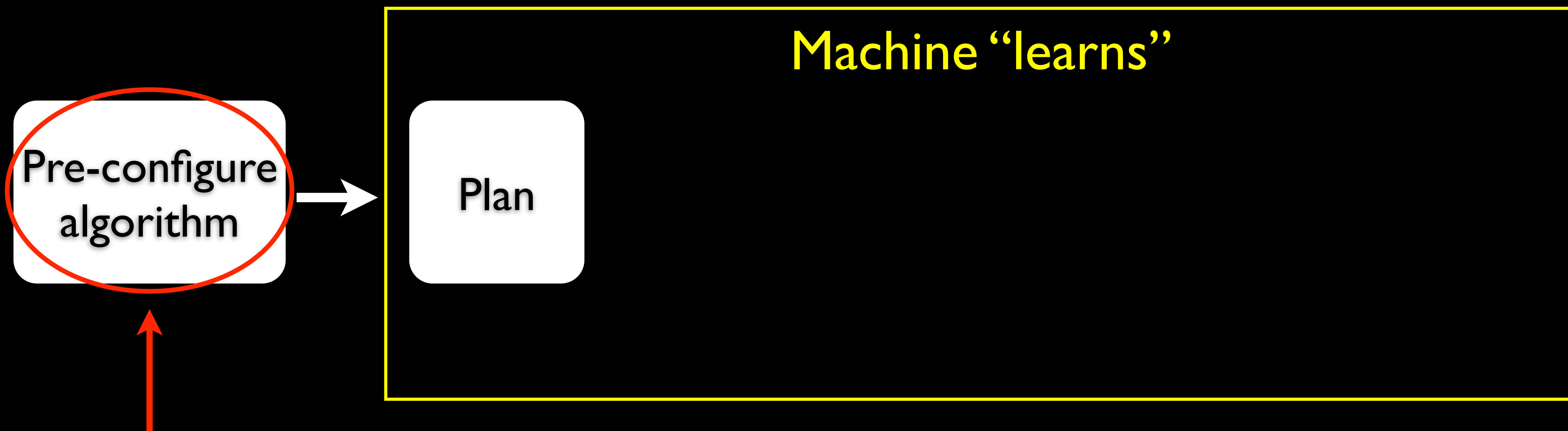
Pre-configure algorithm

Machine "learns"

Technology expert specifies:

- States and Features
- Reward representation
- Learning representations
- Parameter values
- Etc.

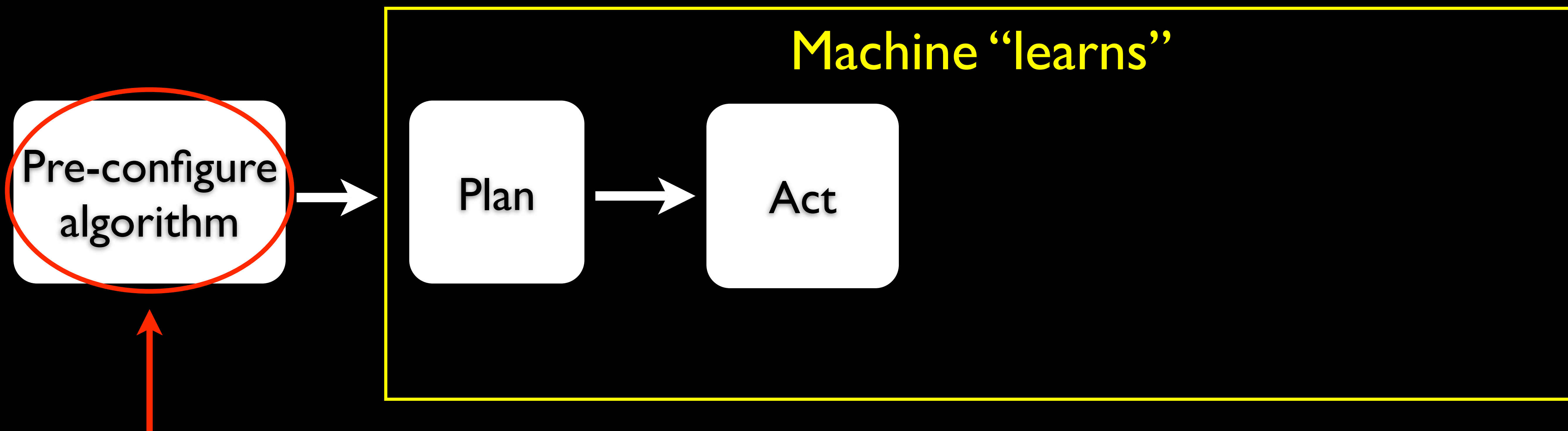
Classical Artificial Learning



Technology expert specifies:

- States and Features
- Reward representation
- Learning representations
- Parameter values
- Etc.

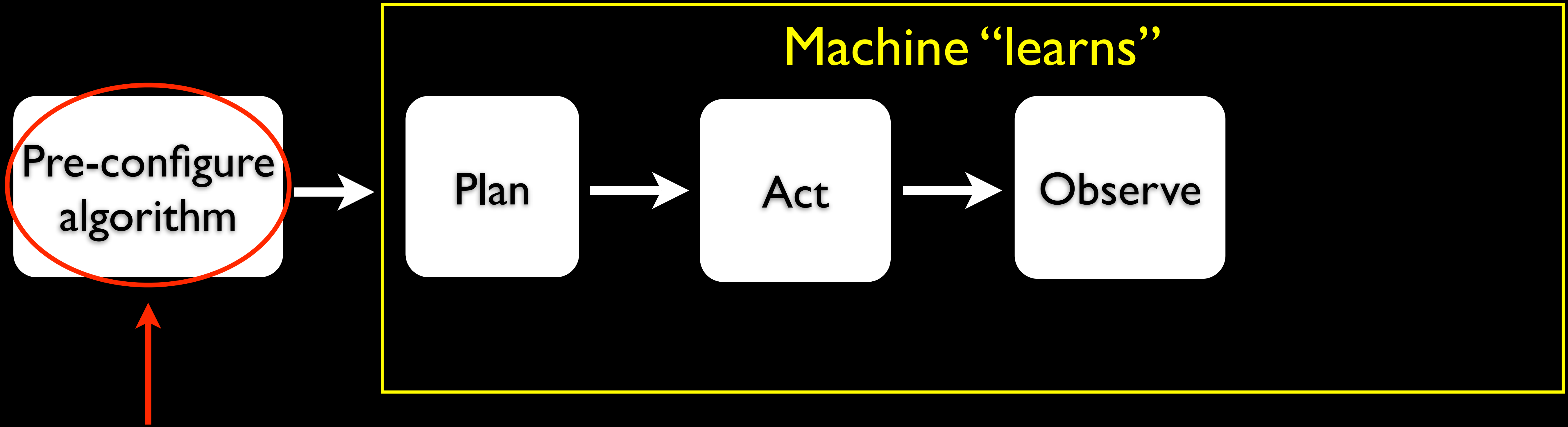
Classical Artificial Learning



Technology expert specifies:

- States and Features
- Reward representation
- Learning representations
- Parameter values
- Etc.

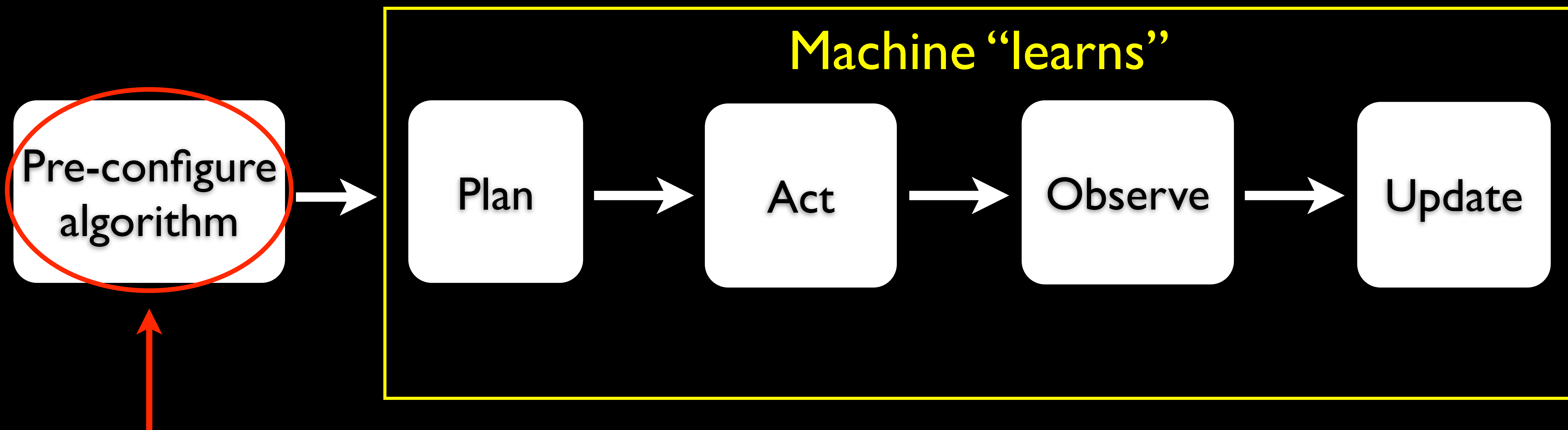
Classical Artificial Learning



Technology expert specifies:

- States and Features
- Reward representation
- Learning representations
- Parameter values
- Etc.

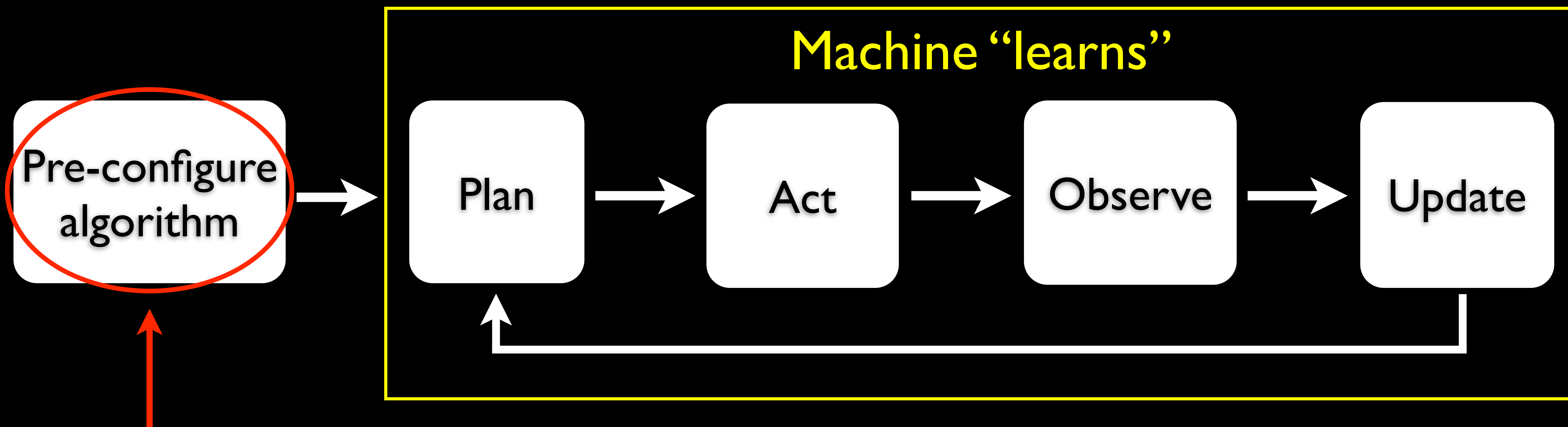
Classical Artificial Learning



Technology expert specifies:

- States and Features
- Reward representation
- Learning representations
- Parameter values
- Etc.

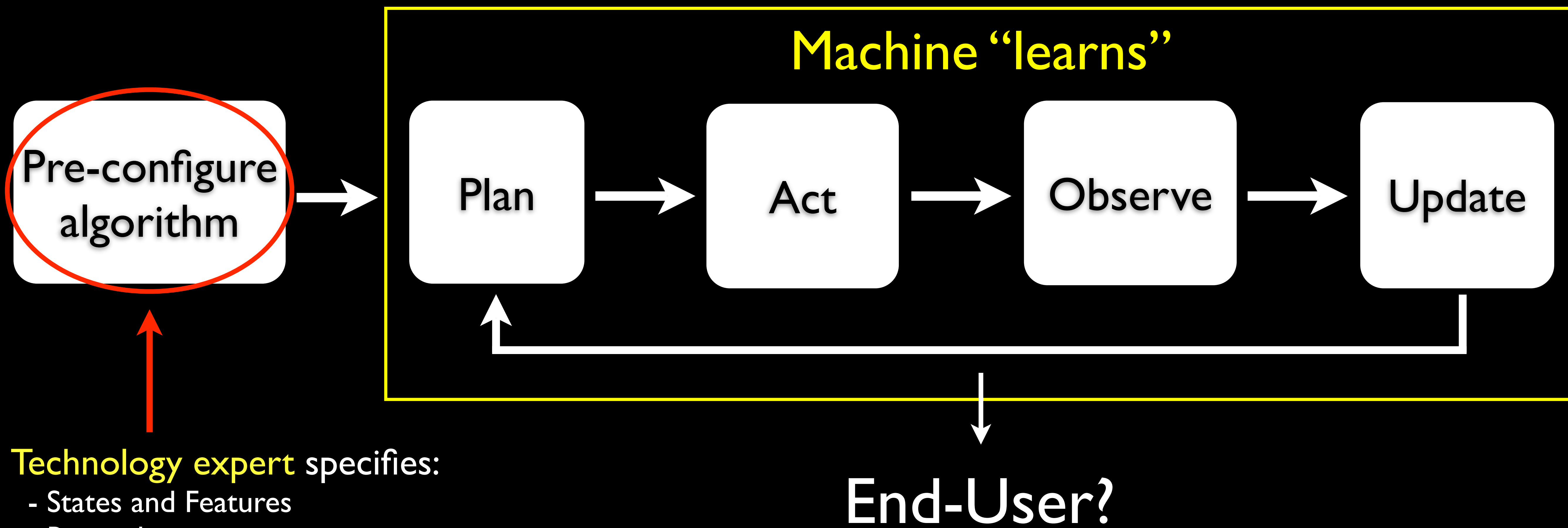
Classical Artificial Learning



Technology expert specifies:

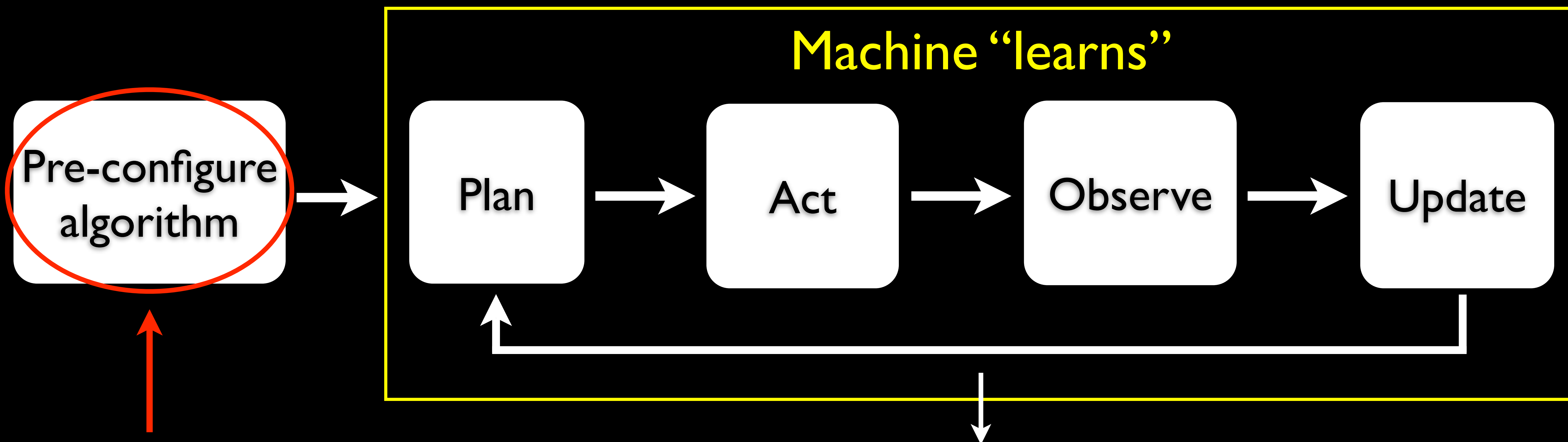
- States and Features
- Reward representation
- Learning representations
- Parameter values
- Etc.

Classical Artificial Learning



Technology expert specifies:
- States and Features
- Reward representation
- Learning representations
- Parameter values
- Etc.

Classical Artificial Learning

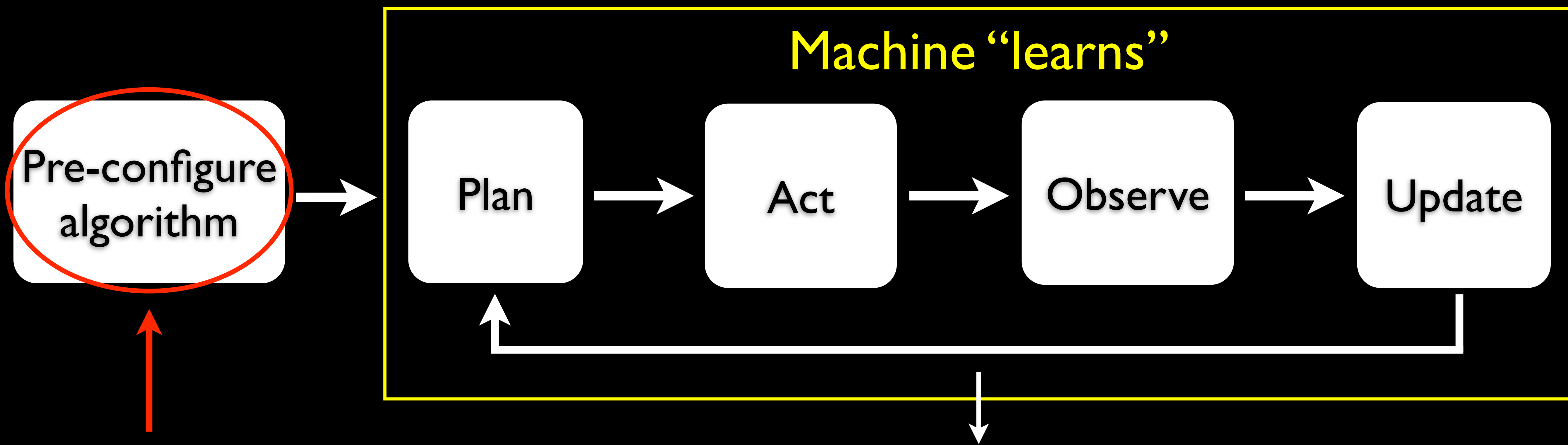


Technology expert specifies:
- States and Features
- Reward representation
- Learning representations
- Parameter values
- Etc.

End-User?

- Learning process is too slow and dangerous

Classical Artificial Learning



Technology expert specifies:

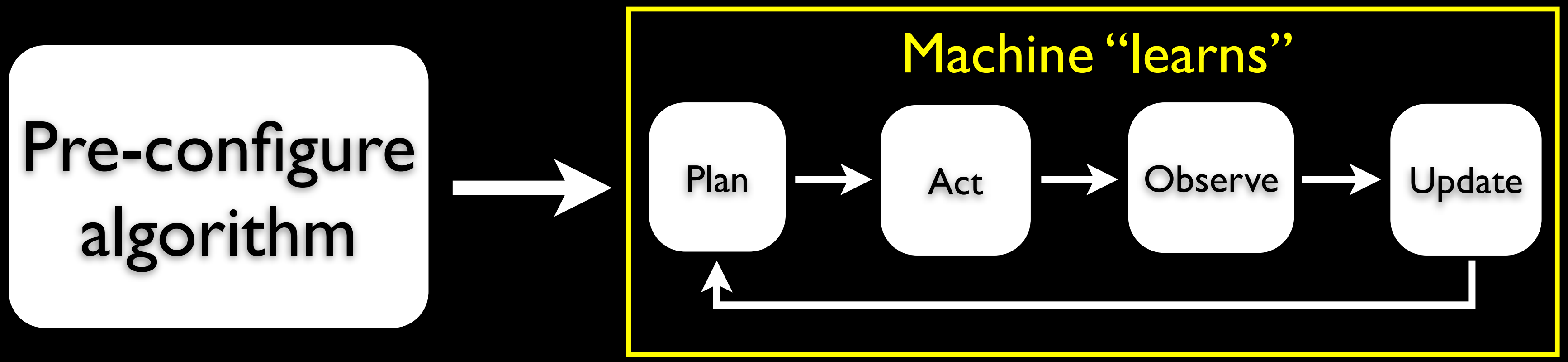
- States and Features
- Reward representation
- Learning representations
- Parameter values
- Etc.

End-User?

- Learning process is too slow and dangerous
- Plus ....

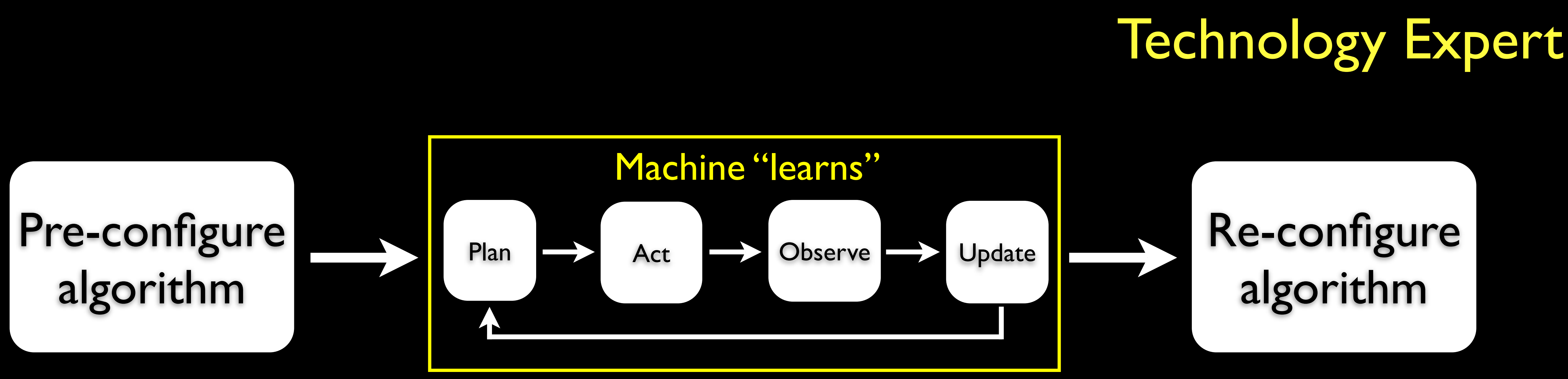
What really happens

- Designer doesn't know *a priori* what the end-user wants
  - **learning algorithm fails**



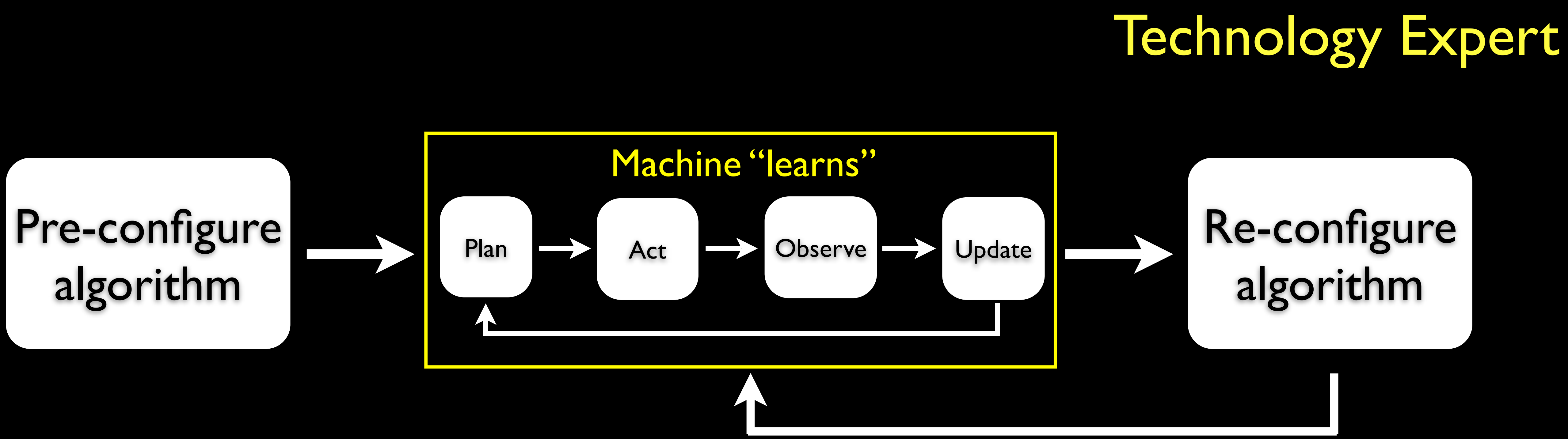
What really happens

- Designer doesn't know *a priori* what the end-user wants
  - **learning algorithm fails**

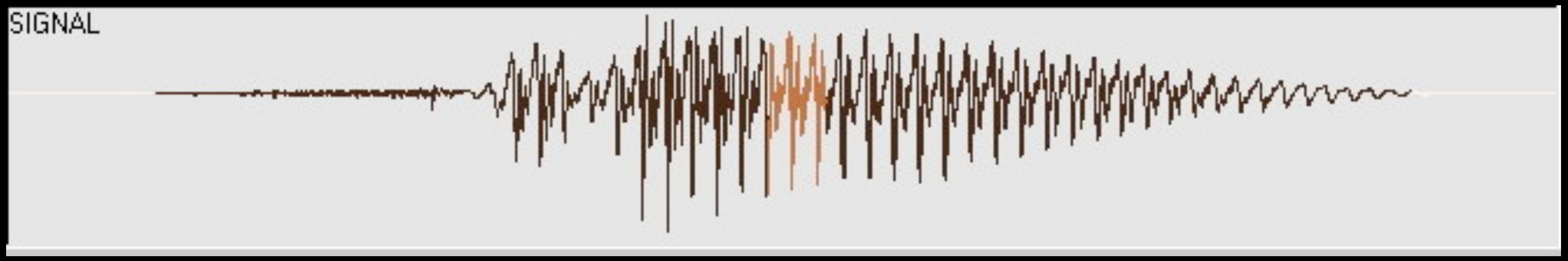


What really happens

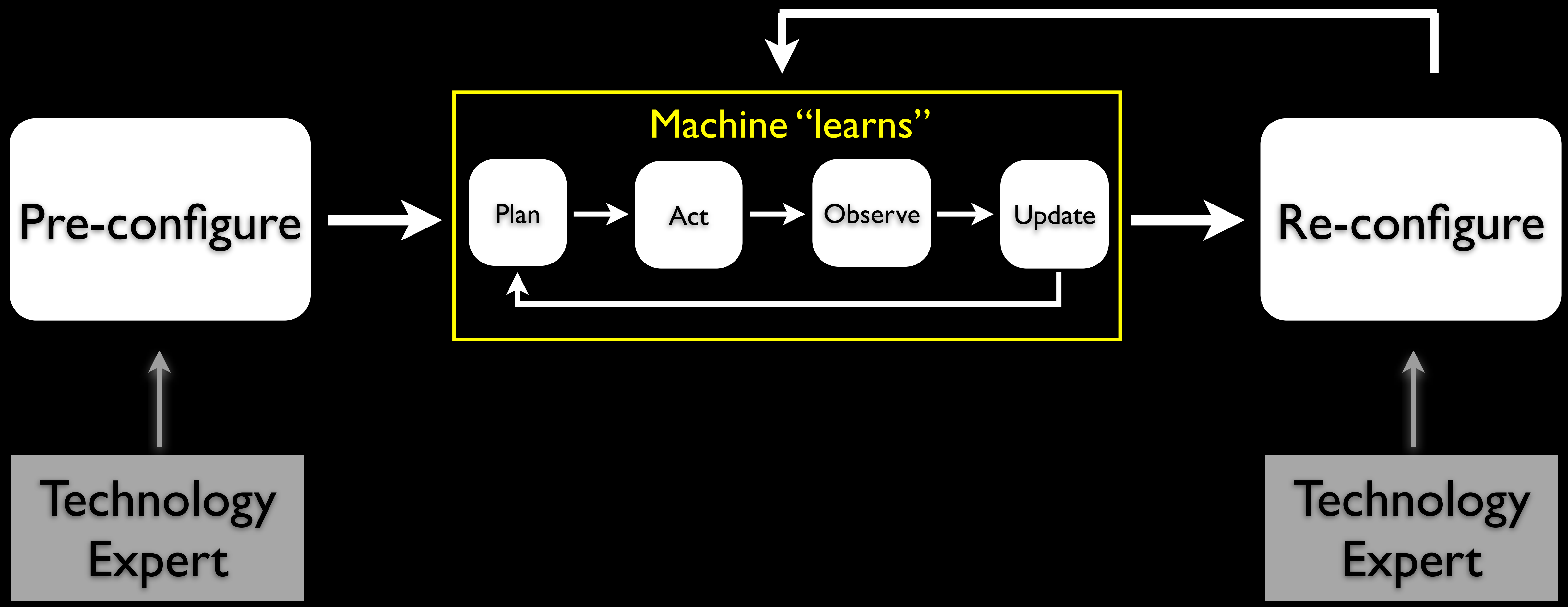
- Designer doesn't know *a priori* what the end-user wants
  - **learning algorithm fails**



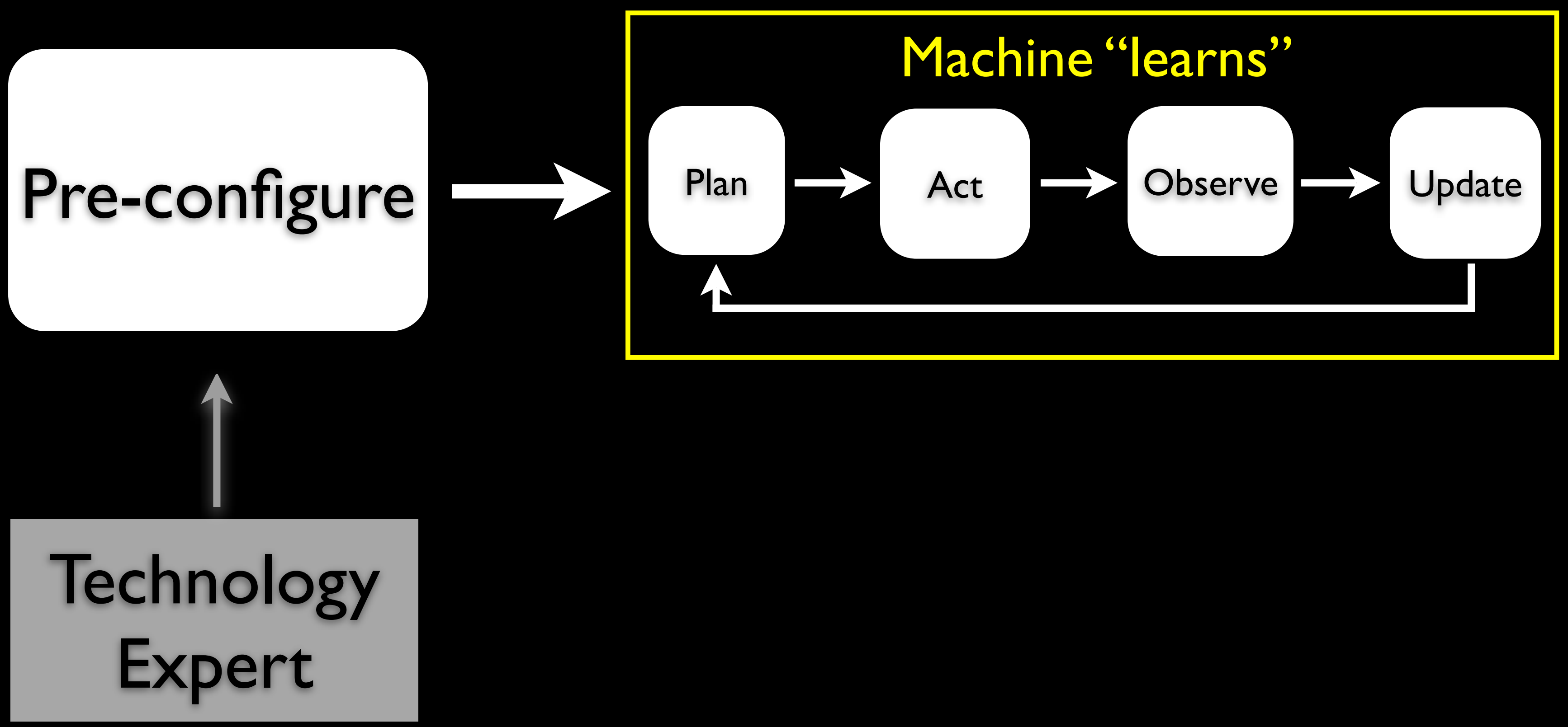
Insane Researcher Skills



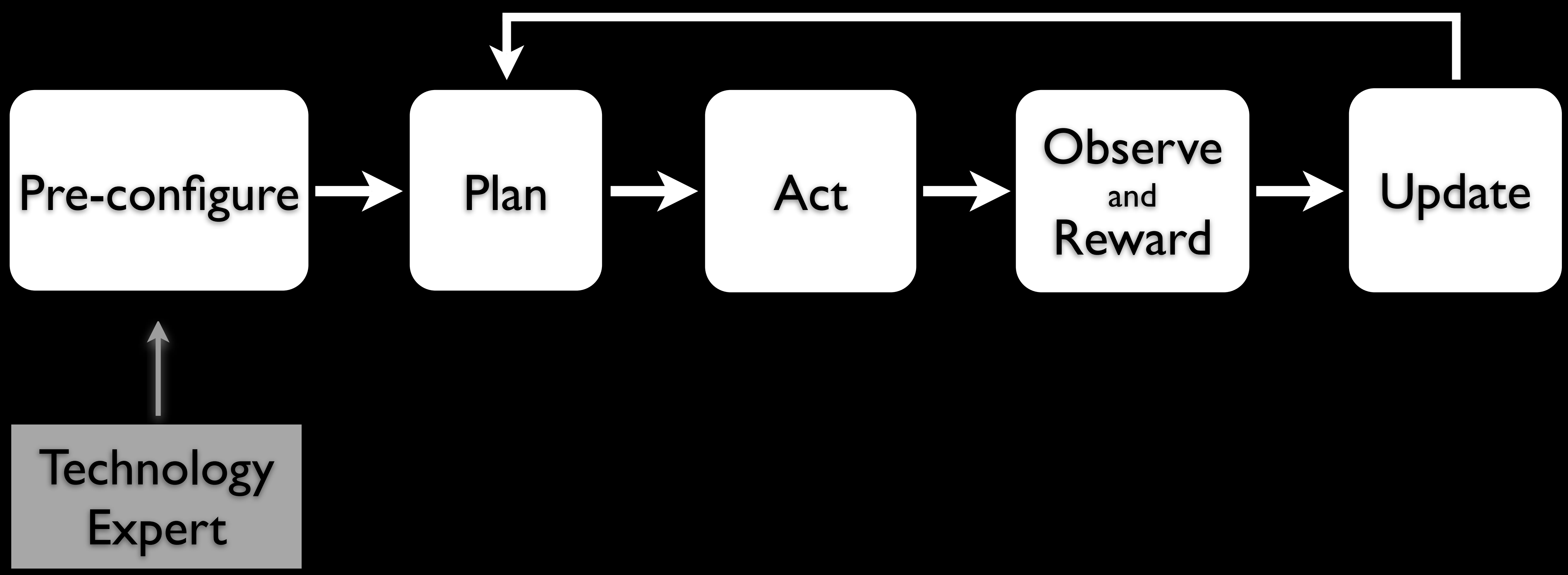
Interactive Artificial Learning



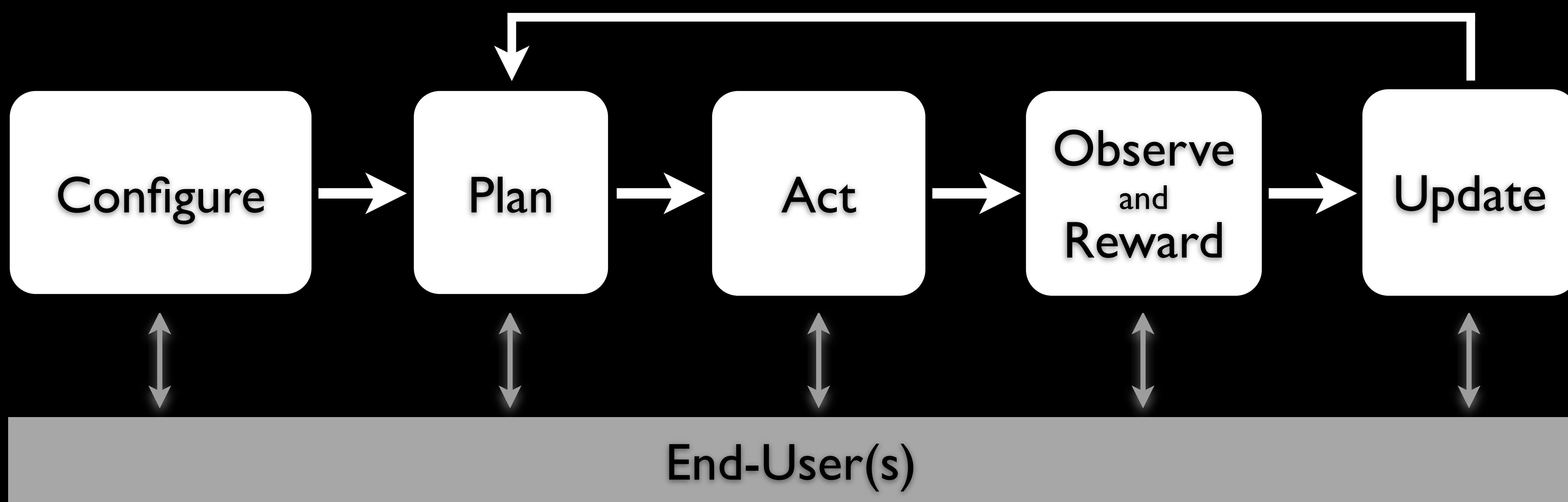
Interactive Artificial Learning



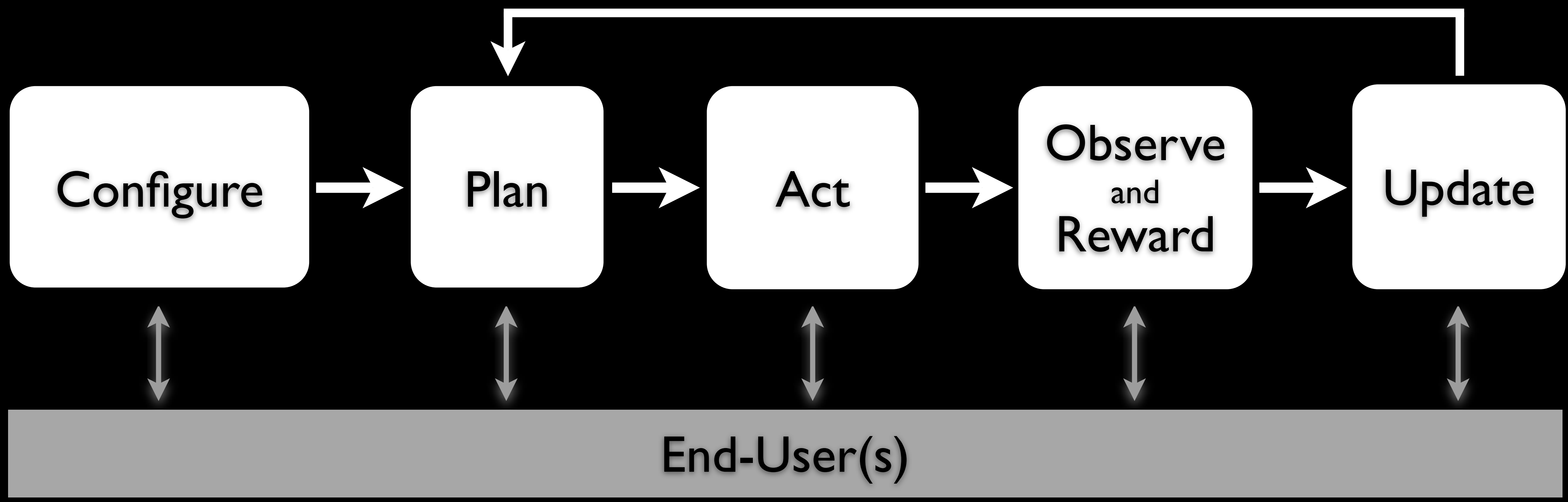
Interactive Artificial Learning



Interactive Artificial Learning



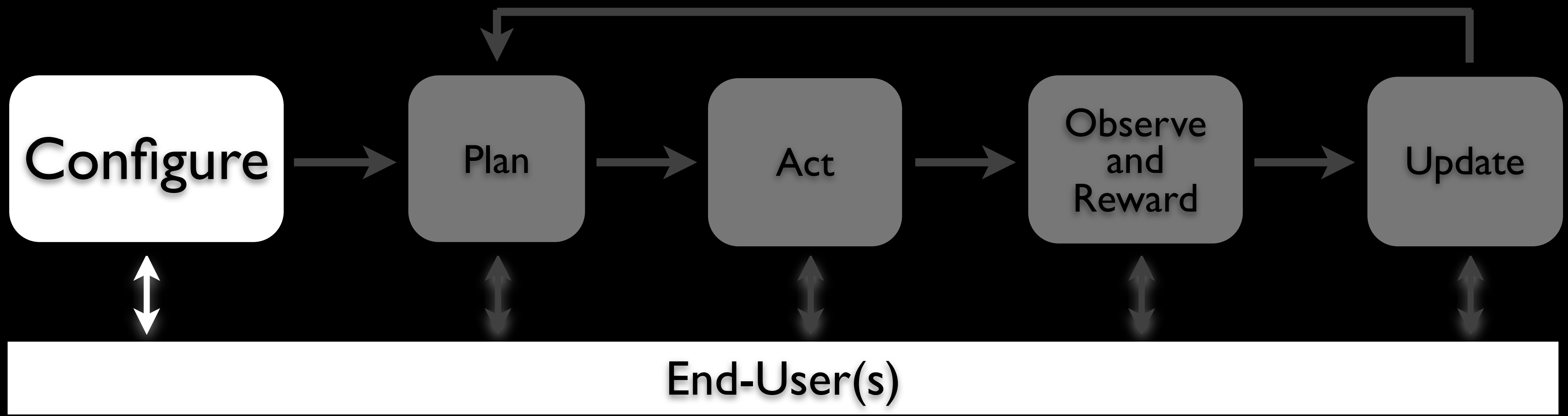
Interactive Artificial Learning



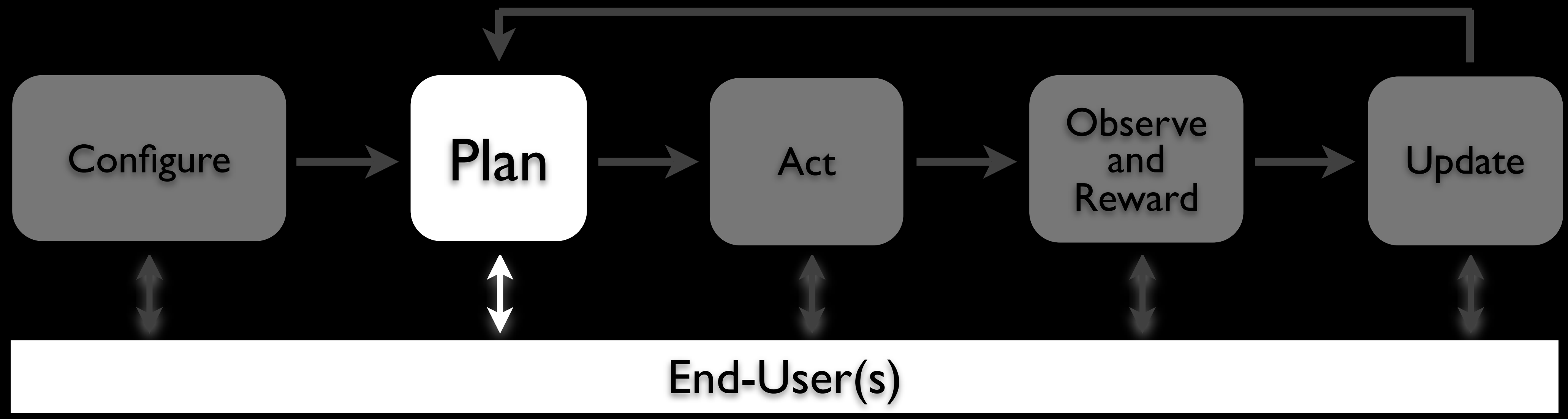
Which interactions are most productive?

What should the interactions be like?

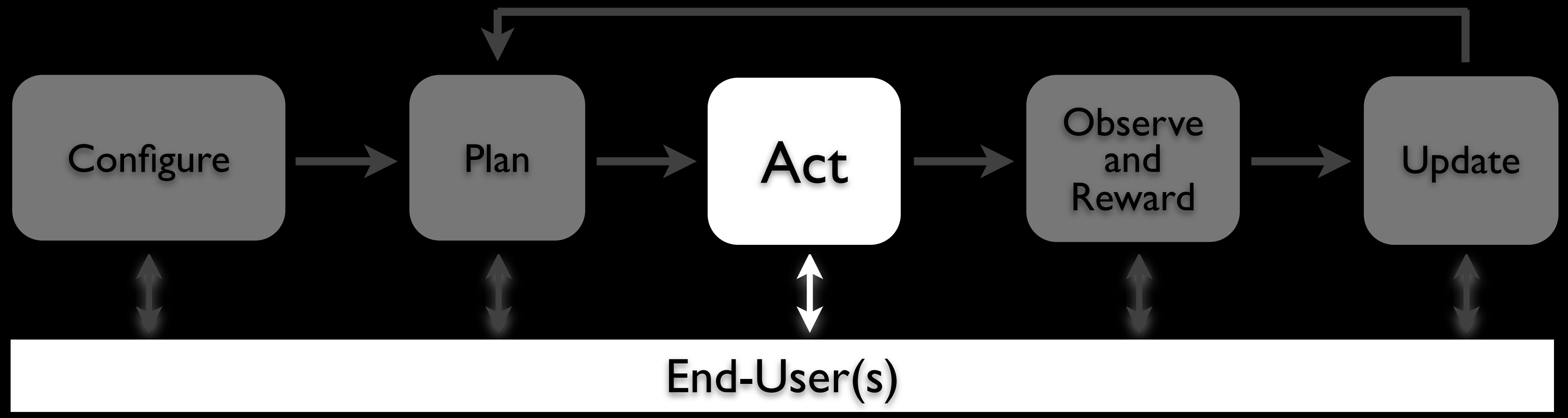
What learning algorithms facilitate these interactions?



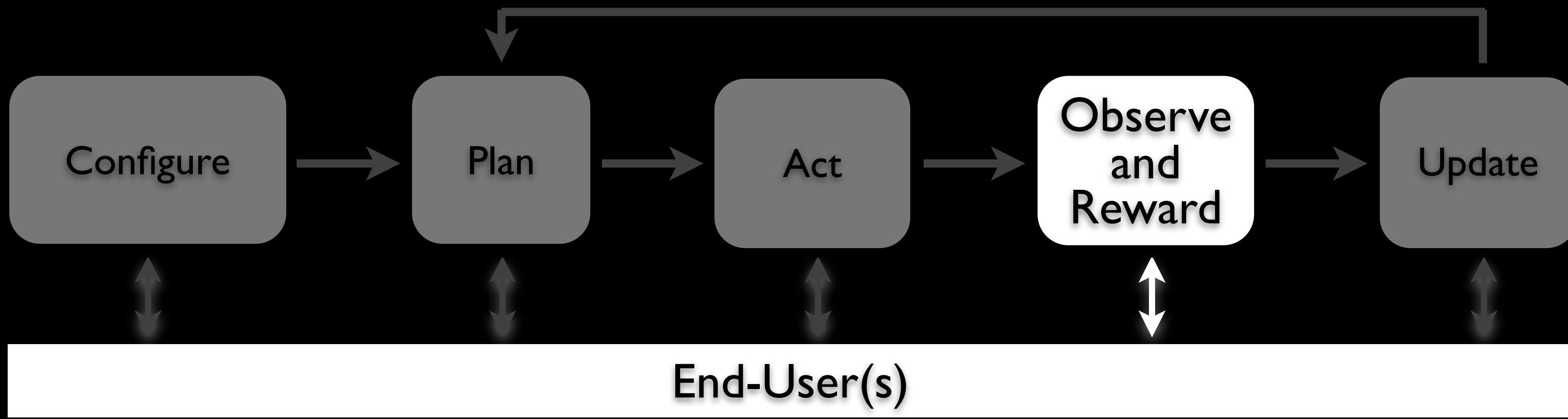
- Configuration vs. pre-configuration
- Select from a toolbox of algorithms, representations, etc.
- Possibilities for reward scaffolding



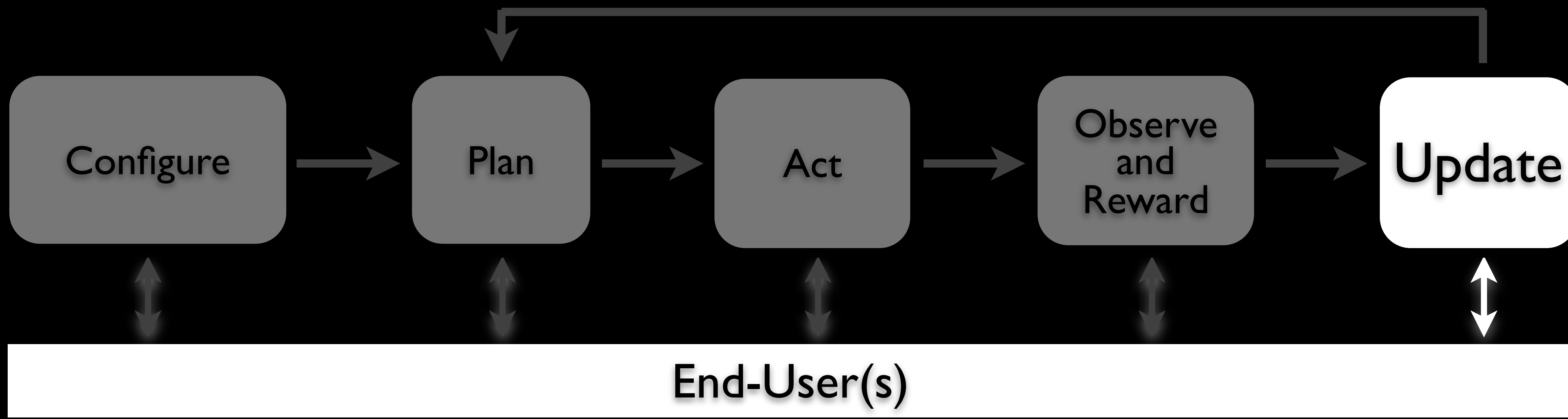
- **Remove the black-box**
- Replace with collaborative process
  - human and machine ask **questions**, give **answers**, make **hypothesis**, etc.
  - Improve utility estimates and human understanding
- **Explore** estimated effects of parameters and values
  - play with internal state and parameters



- Teaching by demonstration/imitation learning
- Additional benefit: improve "situation awareness"
  - (Kaber & Endsley 2004)
- Who takes control?



- Specify or "discuss" utility of outcome
  - User provides part of the reward vector
- Scaffolding
- Identify and annotate outcome



- **Remove black-box**, have a **conversation**
  - leverage human knowledge
  - improve human knowledge
- **Visualization** of how machine sees the world
  - Fails & Olsen, 2003
  - help user understand what machine learned

Conclusions

- A lot has been done
  - teaching by demonstration, imitation learning, reward specification
- But let’s **open up the box**
  - Interactively determine representations, features, etc.
  - Interaction in planning, updating, configuring