## Soar (A. Newell et al.)

**Overview:**

Soar is an architecture for human cognition expressed in the form of a production system. It involves the collaboration of a number of researchers including Allen Newell, John Laird and Paul Rosenbloom and others at different institutions. The theory builds upon earlier efforts involving Newell such as GPS (Newell & Simon) and GOMS (Card, Moran & Newell). Like the latter model, Soar is capable of simulating actual responses and response times.

The principal element in Soar is the idea of a problem space: all cognitive acts are some form of search task. Memory is unitary and procedural; there is no distinction between procedural and declarative memory. Chunking is the primary mechanism for learning and represents the conversion of problem-solving acts into long-term memory. The occasion for chunking is an impasse and its resolution in the problem solving process (i.e., satisfying production rules). Newell states that Soar suggests a reconstructive view of memory (c.f. Bartlett).

Soar exhibits a variety of different types or levels of learning: operators (e.g., create, call), search control (e.g., operator selection, plans), declarative data (e.g., recognition/recall), and tasks (e.g., identify problem spaces, initial/goal states). Soar is capable of transfer within or across trials or tasks.

**Scope/Application**:

Newell (1990) has positioned Soar as the basis for a unified theory of cognition and attempts to show how it explains a wide range of past results and phenomena. For example, he provides interpretations for response time data, verbal learning tasks, reasoning tasks, mental models and skill acquisition. In addition, versions of Soar have been developed that perform as intelligent systems for configuring computer systems and formulating algorithms.

To learn more about recent work on Soar, see the Univ Mich Soar project site.

**Example**:

Newell (1990; pp 335-336) provides the following description of how Soar would handle a simple recognition task:

Study trial: given item, become familiar (recognition problem space)

Operator: recognize-next element

Fails if subitem is not recognized; get an impasse

Subgoal - learn to recognize subitem

Assign a name (recognize item)

Chunk is created to assign name if item is recognized

Test trail: given an item

If chunk fires, name is assigned and item is recognized

If chunk fails to fire, item is not recognized

The key aspect of this example, is that Soar treats recognition as a problem solving activity in which it tries to recursively identify the components of the item and creates an impasse when it fails.

**Principles**:

As a theory of learning, Soar specifies (or confirms) a number of principles:

1. All learning arises from goal-directed activities; specific knowledge is acquired in order to satisify goals (needs)

2. Learning occurs at a constant rate -- the rate at which impasses occur while problem solving (average of 0.5 chunk/second)

3. Transfer occurs by identical elements and is highly specific (c.f. Thorndike). Transfer can be general if the productions are abstract.

4. Rehearsal helps learning provided it involves active processing (i.e., creation of chunks)

5. Chunking is the basis for the organization of memory

**References**:

Laird, J.E., Newell, A., & P.S. Rosenbloom. (1987). Soar: An architecture for general intelligence. Artificial Intelligence, 33, 1-64.

Newell, A. (1990). Unified Theories of Cognition. Cambridge, MA: Harvard University Press.

ที่มา : http://www.suphet.com/index.php?lay=show&ac=article&Id=412505&Ntype=2