The COGENT Tutorial

22nd Annual Conference of
the Cognitive Science Society

August 12th, 2000

Philadelphia, USA

Tutorial Overview

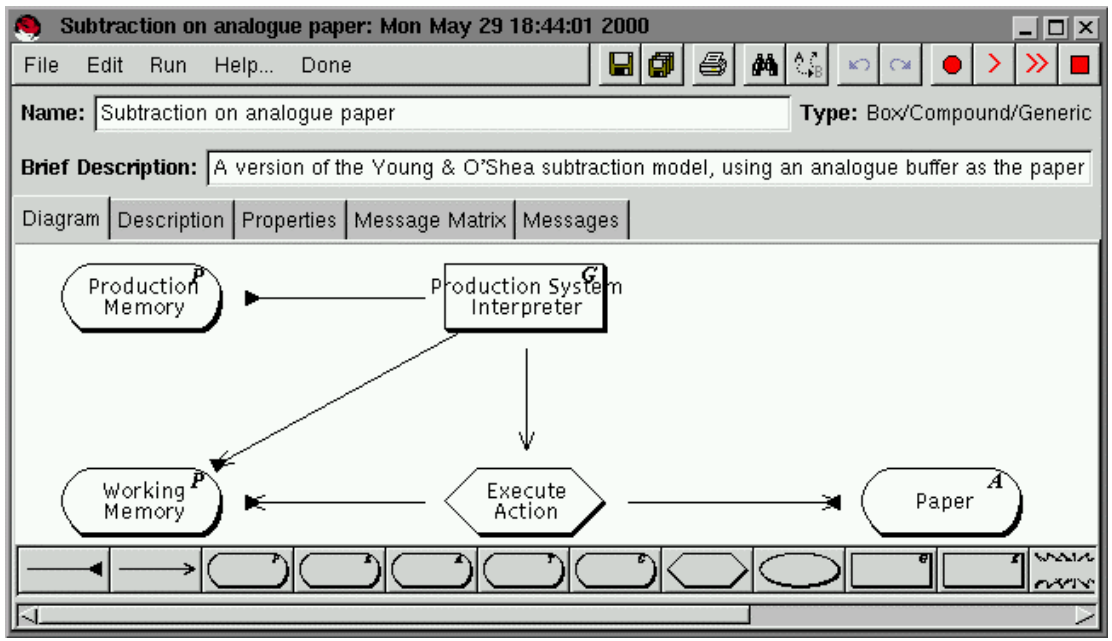
- COGENT: An Overview
- The Tutorial Task: The Modal Model of Memory
- The COGENT ‘Modal Model’ Model

- Building the Short-Term Store
- Adding the Long-Term Store
- Decay, Time and Rehearsal

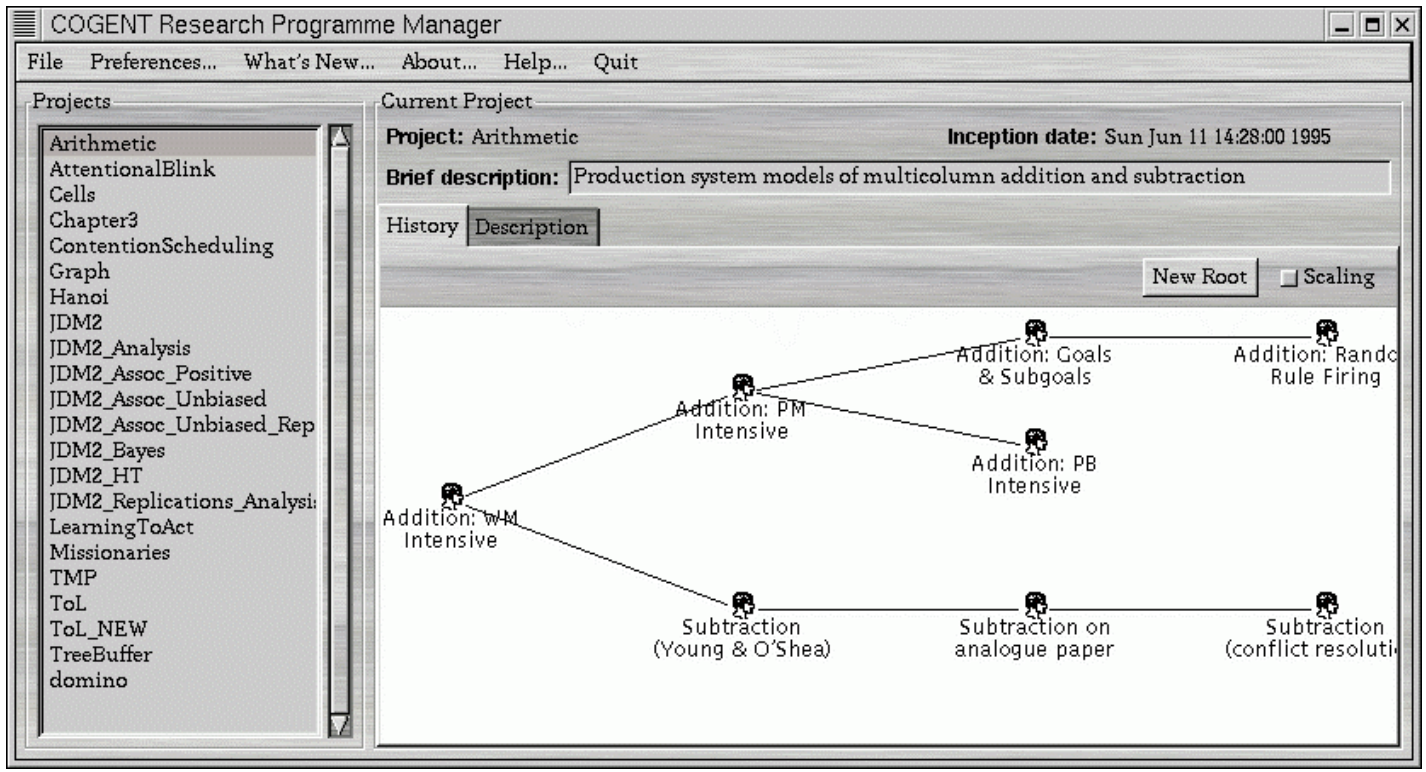
COGENT: Principal Features

- A visual programming environment;
- Research programme management tools;
- A range of standard functional components;
- An expressive rule-based modelling language and implementation system;
- Automated data visualisation tools; and
- A powerful model testing environment.

Visual Programming in COGENT



Research Programme Management



Standard Functional Components

- A library of components is supplied:
  - Memory buffers
  - Rule-based processes
  - Simple connectionist networks
  - Data input/output devices
- Components can be configured for different applications

Rule-Based Modelling Language: I

Processes may contain rules such as:

IF minuend(X) is in *Working Memory*
subtrahend(X) is in *Working Memory*

THEN add equal(minuend, subtrahend) to *Working Memory*
send difference(0) to *Write Answer*

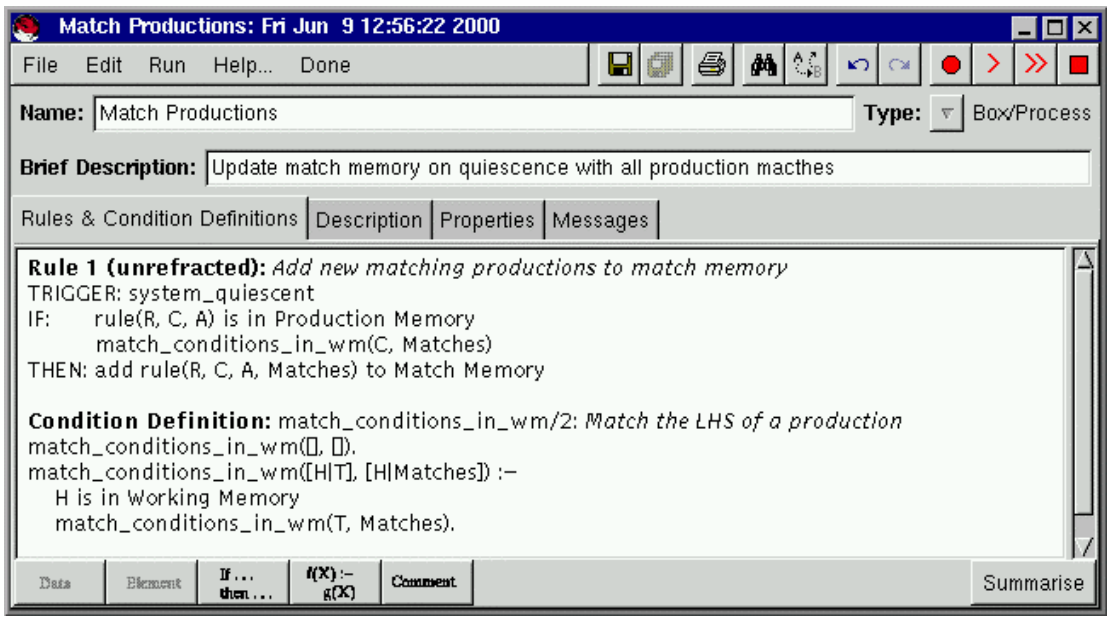
Rule-Based Modelling Language: II

COGENT's representation language is based on Prolog:

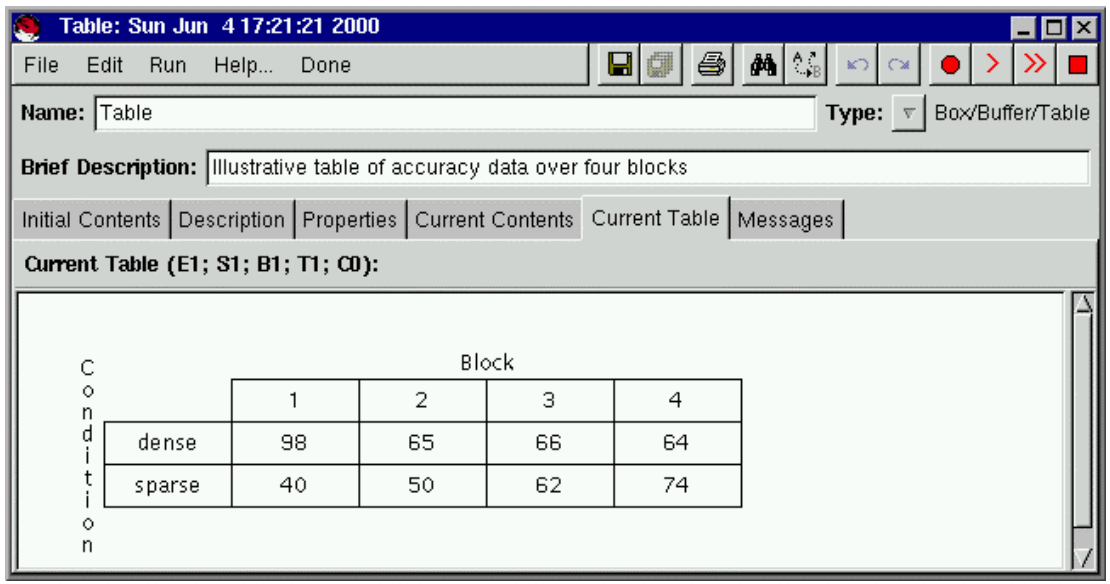
IF minuend(X) is in *Working Memory*
subtrahend(X) is in *Working Memory*

THEN add equal(minuend, subtrahend) to *Working Memory*
send difference(0) to *Write Answer*

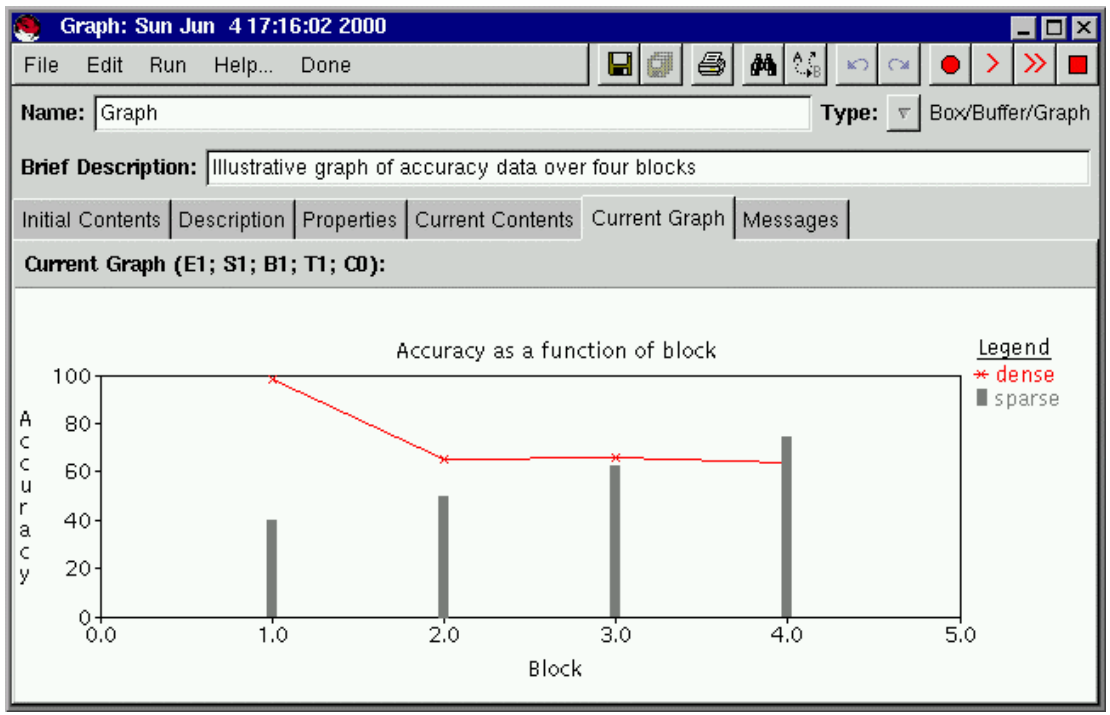
Rule-Based Modelling Language: III



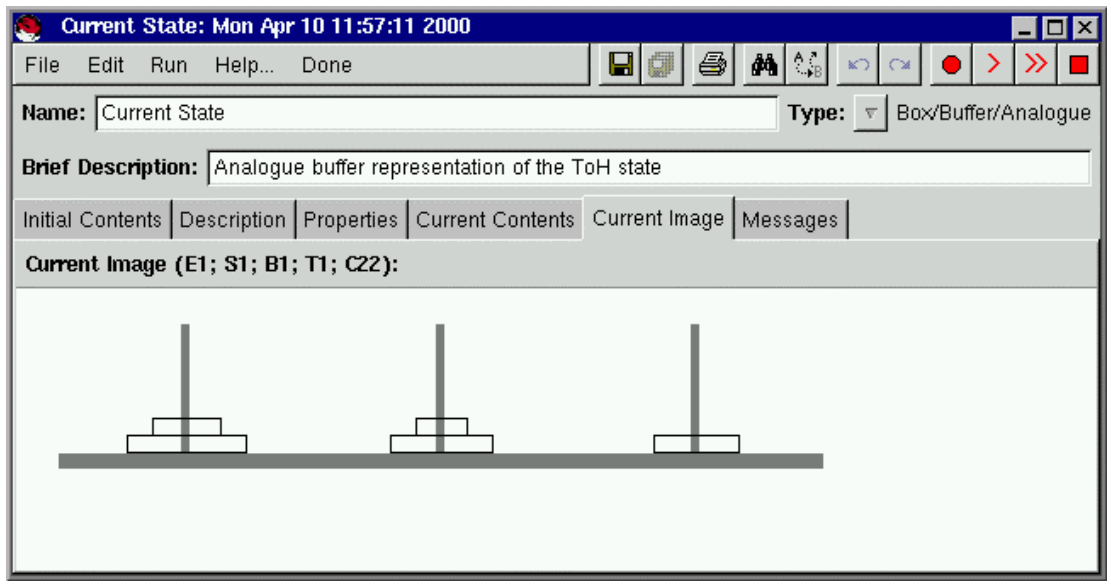
Data Visualisation Tools: Tables



Data Visualisation Tools: Graphs



Data Visualisation Tools: Pictures



The Model Testing Environment

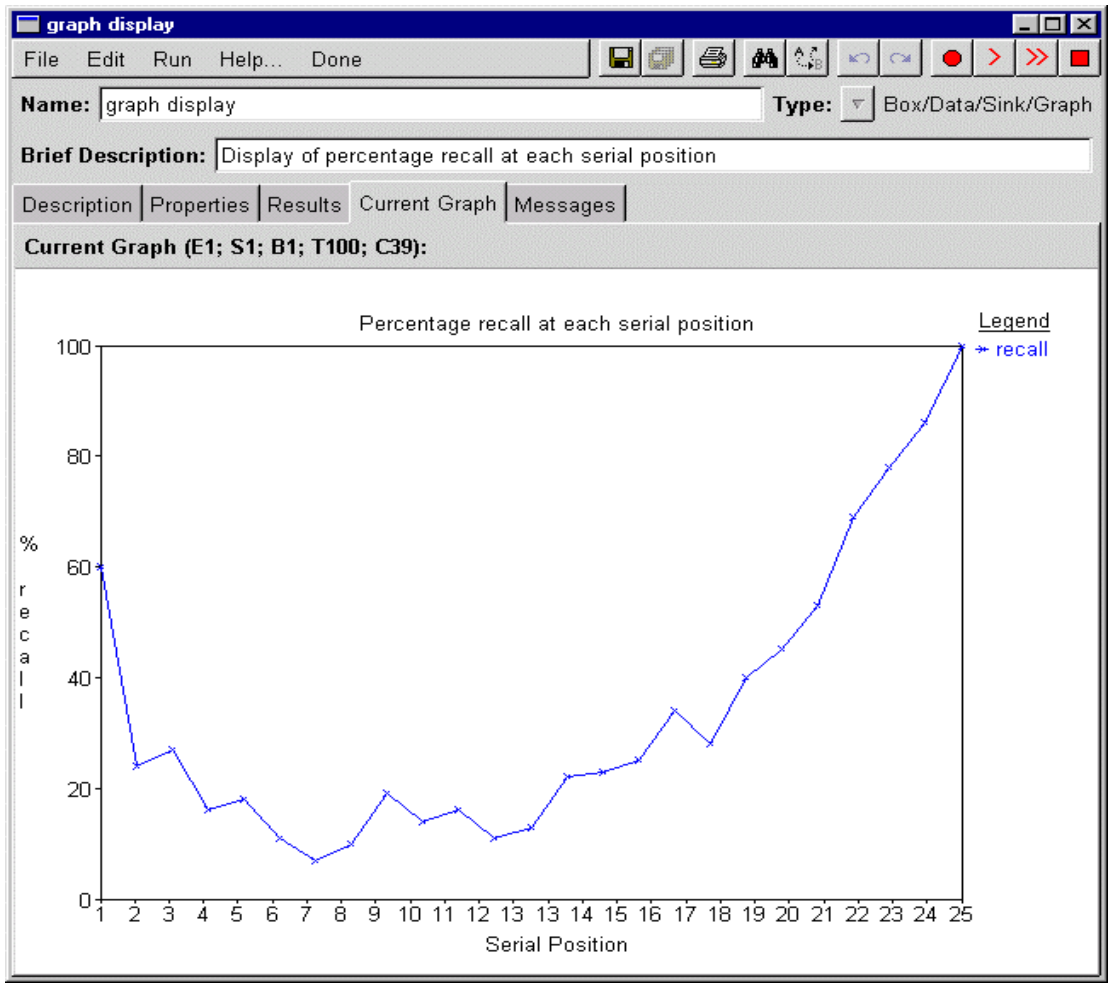
- Visualisation tools are dynamically updated
- Facilities are included to trace inter-component communication
- A flexible “scripting” environment allows:
  - models to be run over multiple blocks of trials;
  - multiple “subjects” to be run over multiple blocks;
  - automated parameter varying “meta-experiments”.

The Tutorial Task: Free Recall

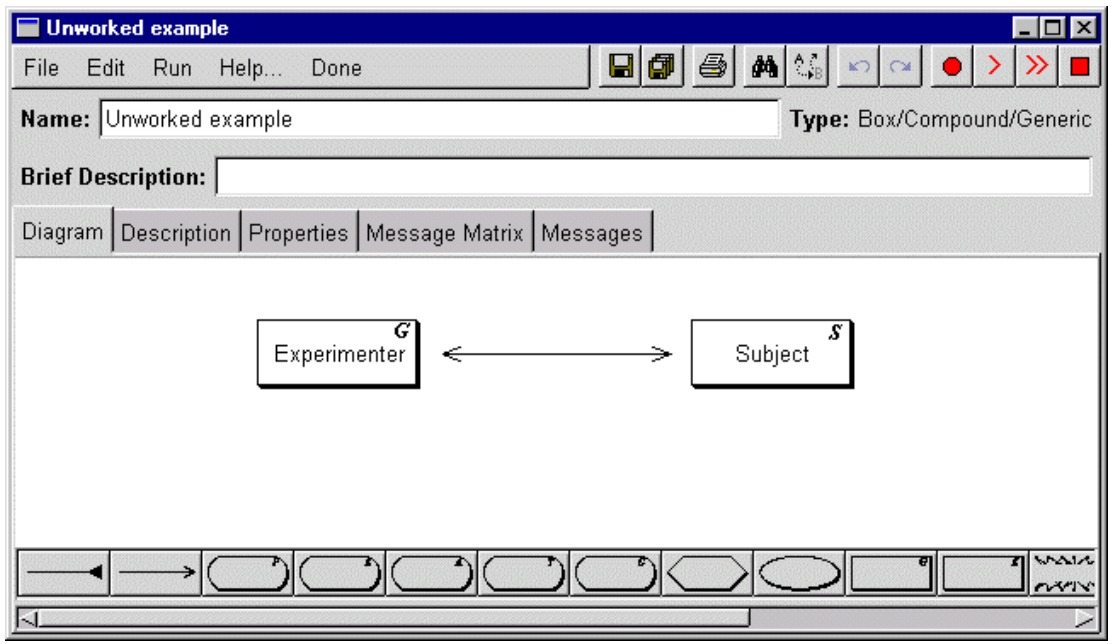
- On each trial, the subject is presented with a list of 25 words
- The subject is told to try to memorise the words
- After an interval, the subject must recall as many words as possible

(Glanzer & Cunitz, 1966)

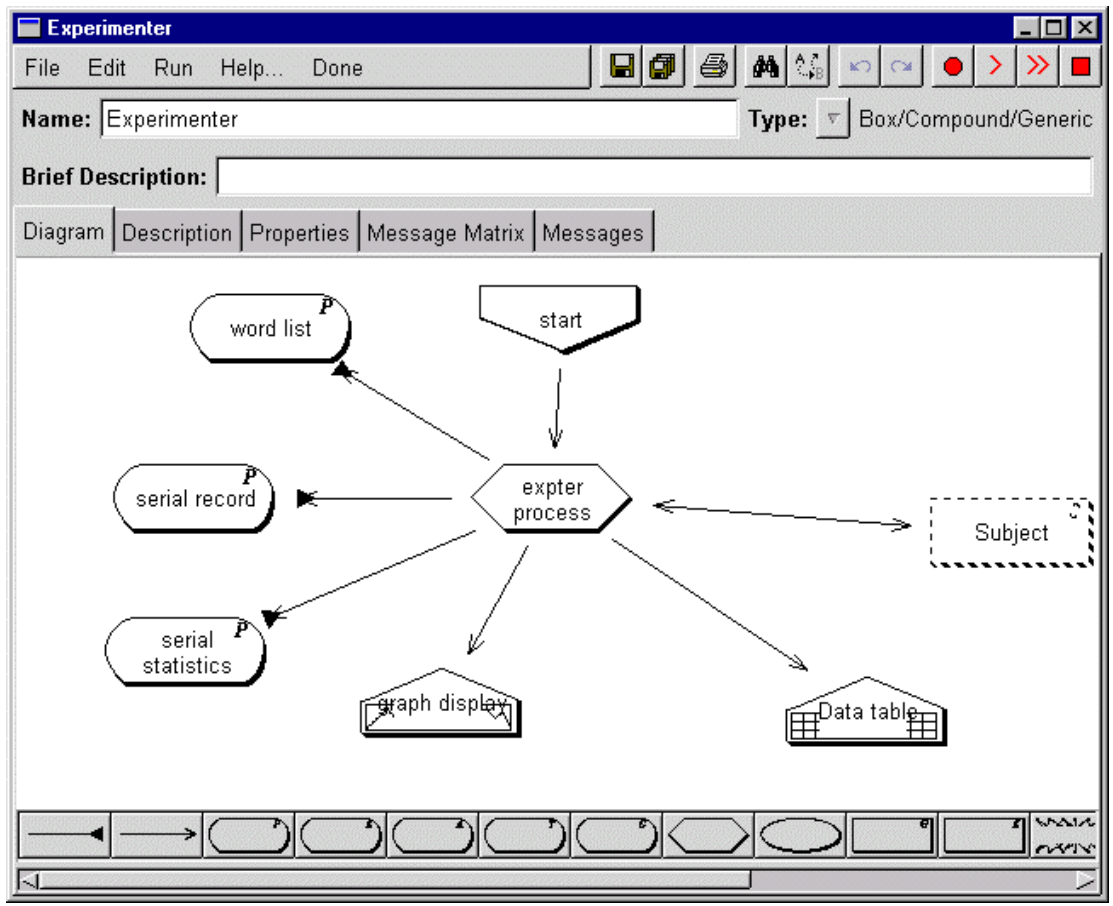
Free Recall: Empirical Findings



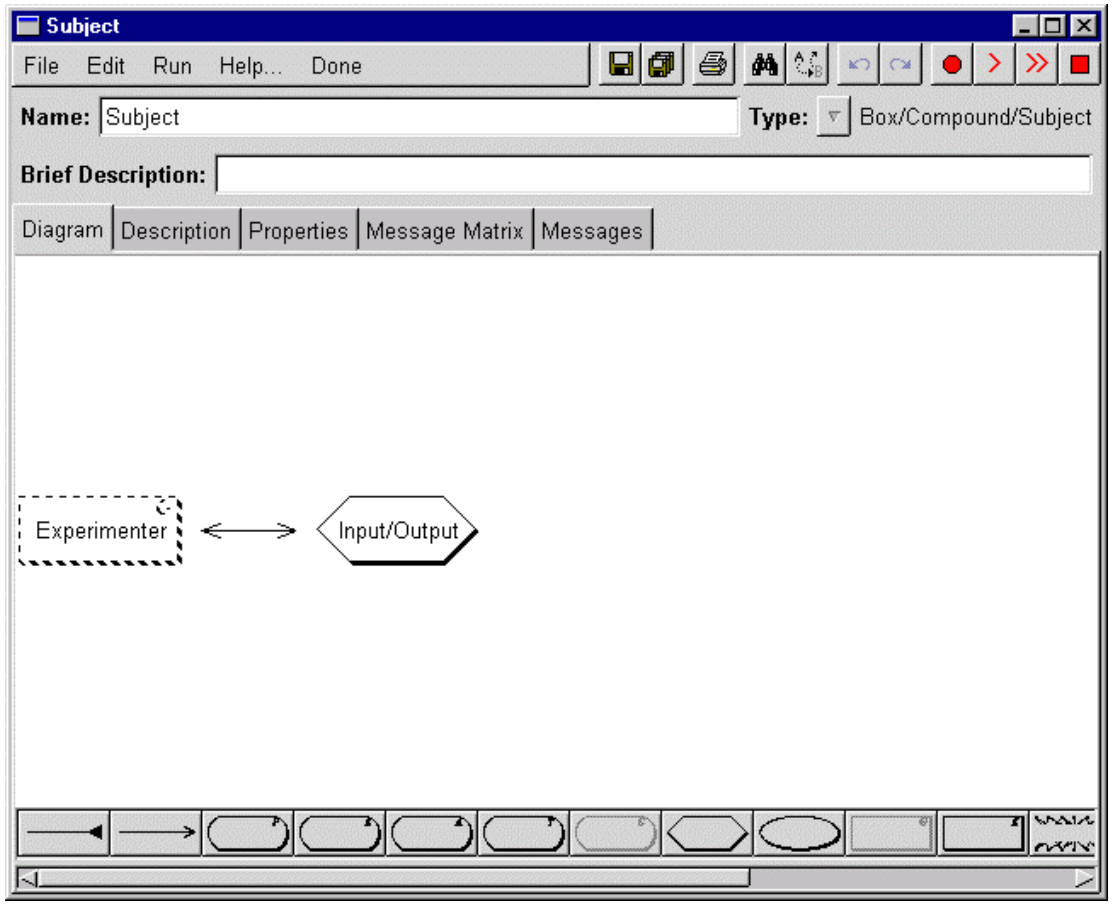
The Modal Model: Top Level



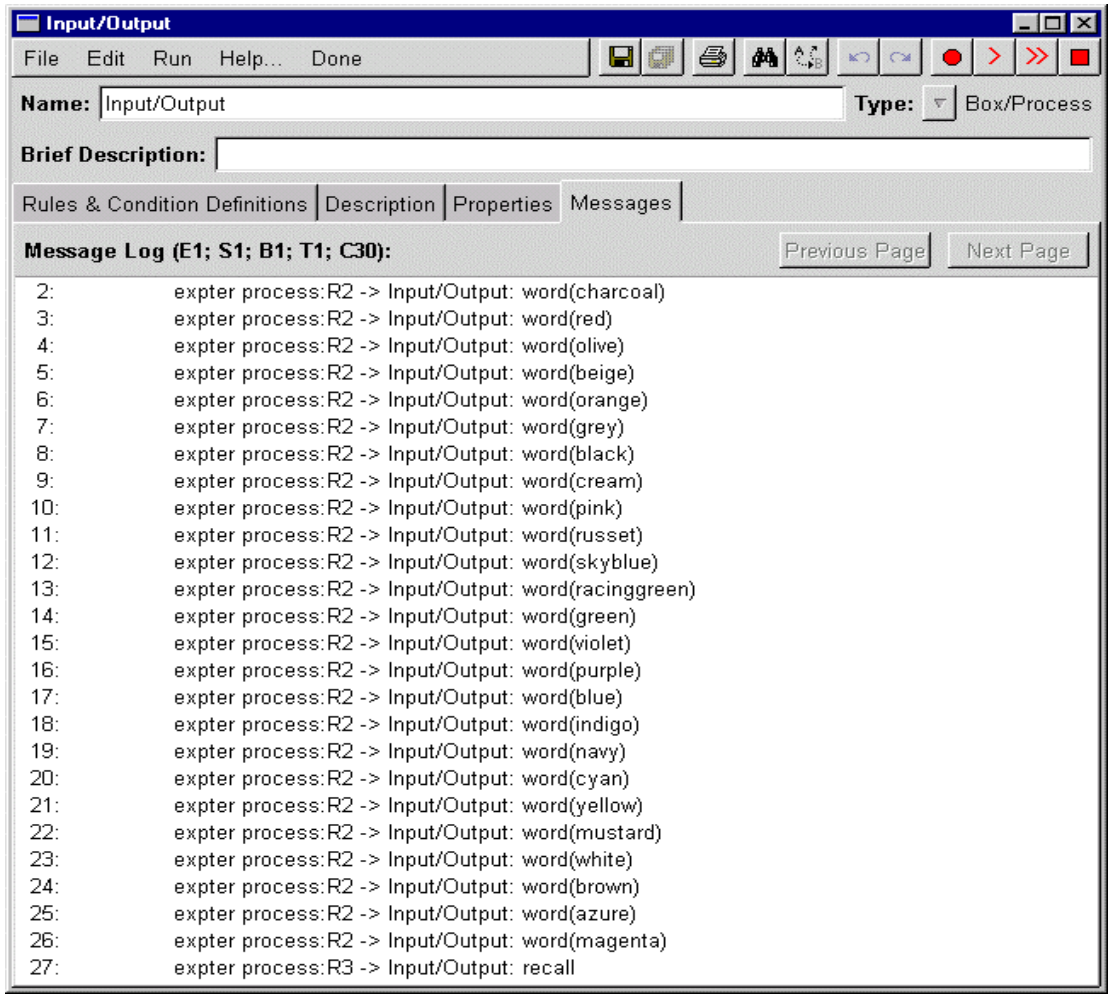
The Modal Model: *Experimenter*



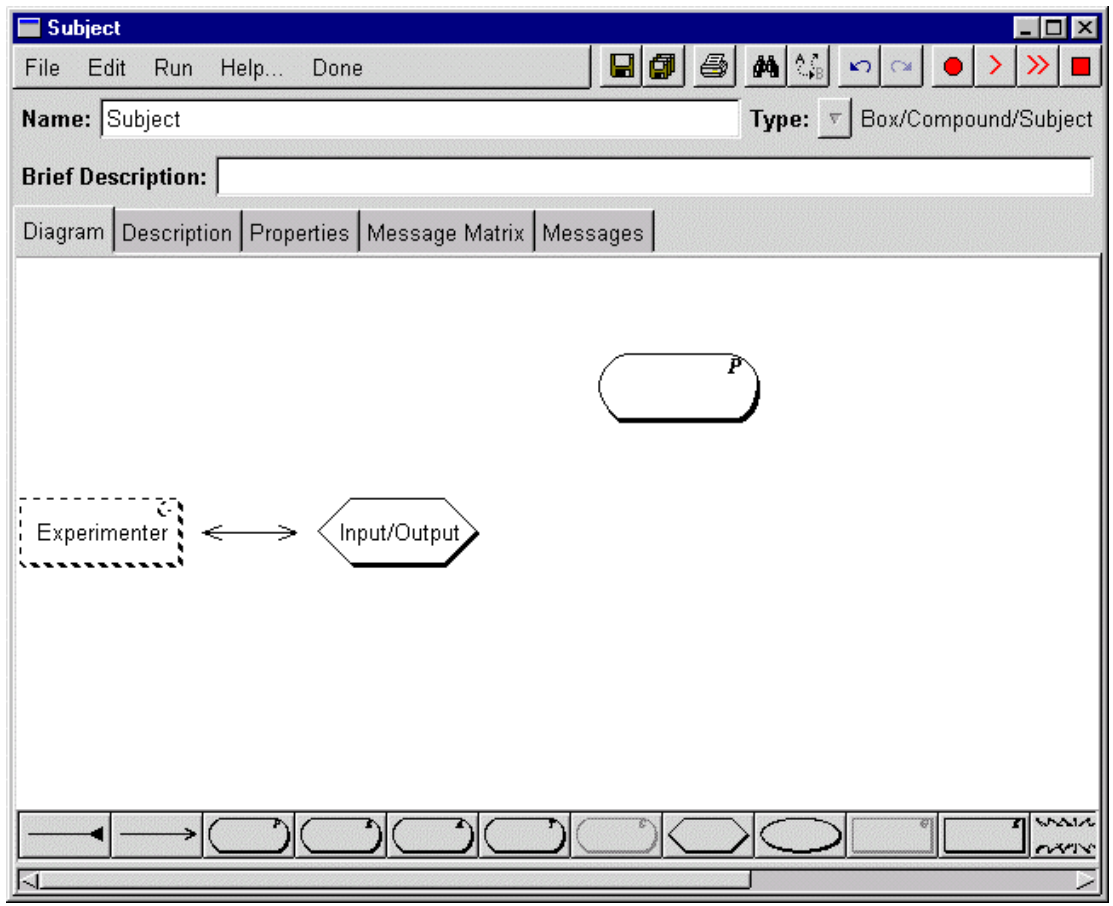
The Modal Model: *Subject*



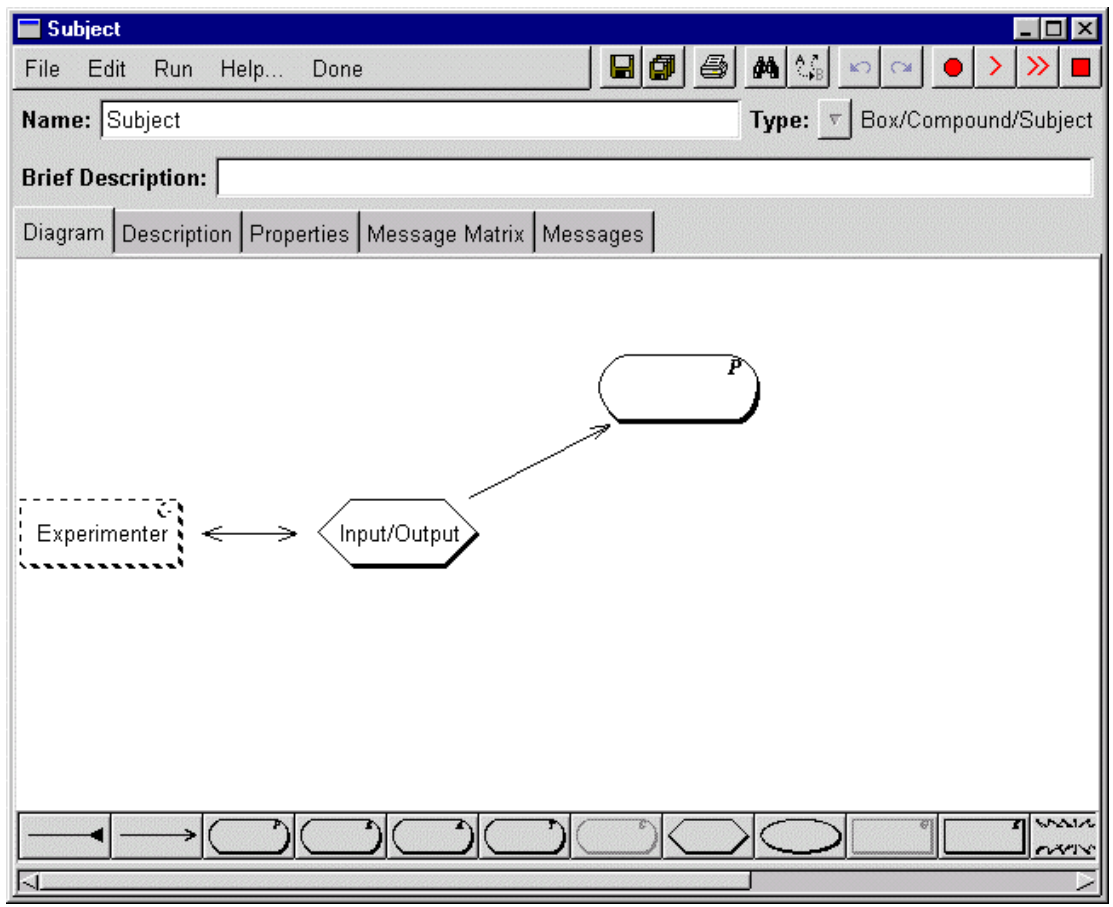
The Model Model: Messages



Building the Short Term Store: I

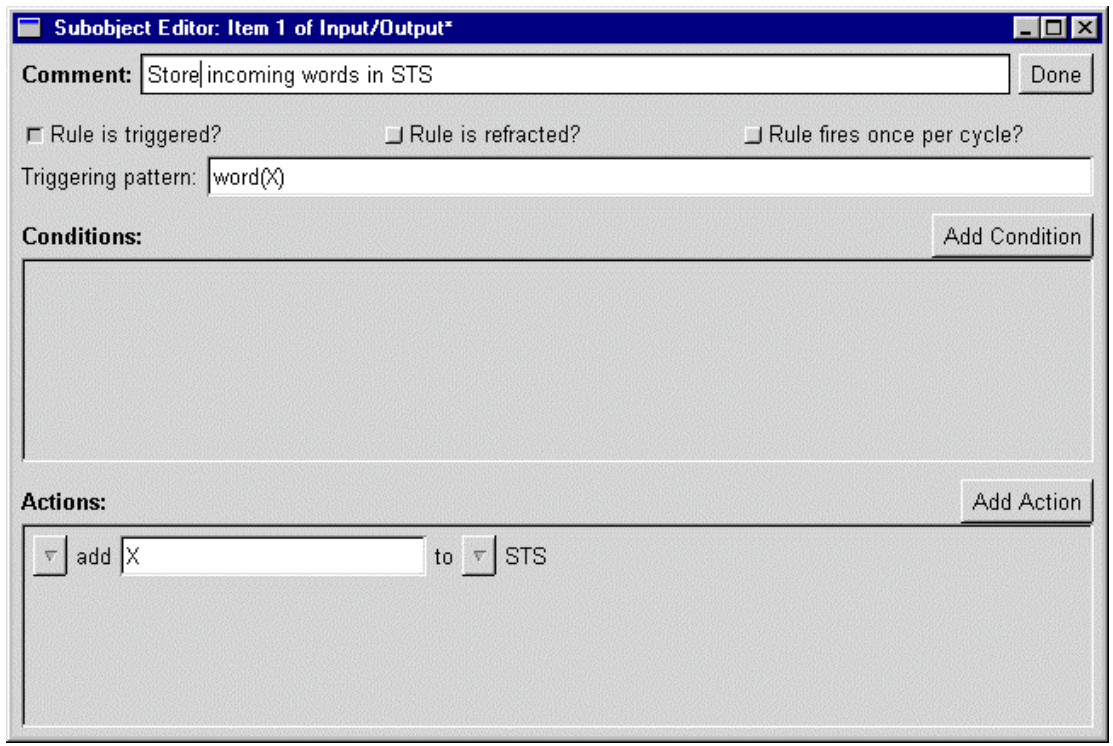


Building the Short Term Store: II

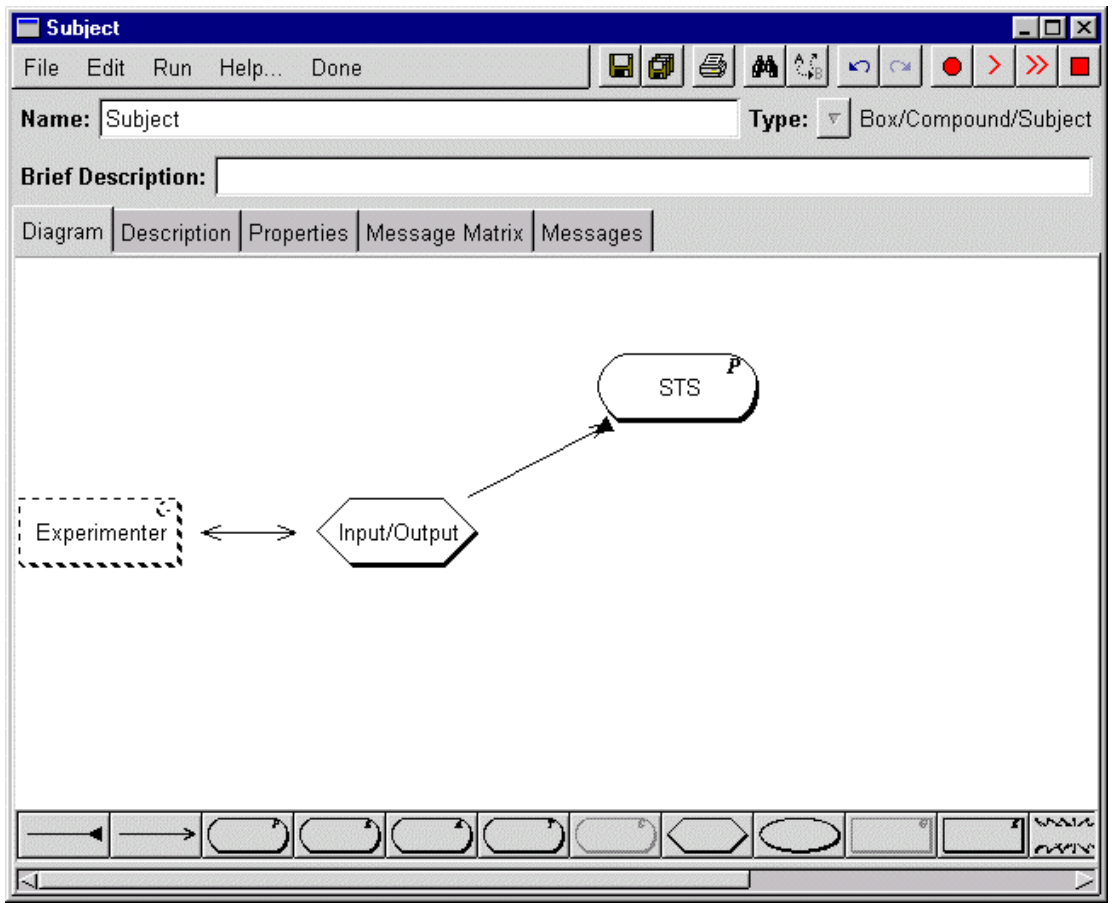


Building the Short Term Store: III

The rule to transfer words to *STS*:

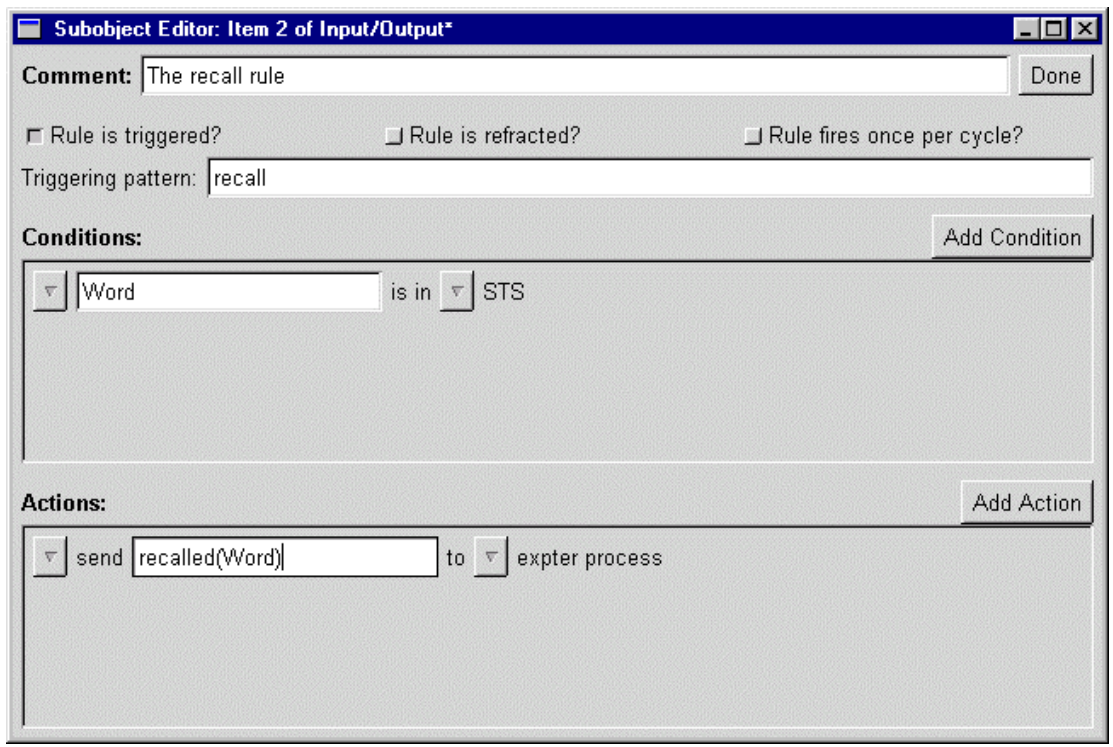


Building the Short Term Store: IV

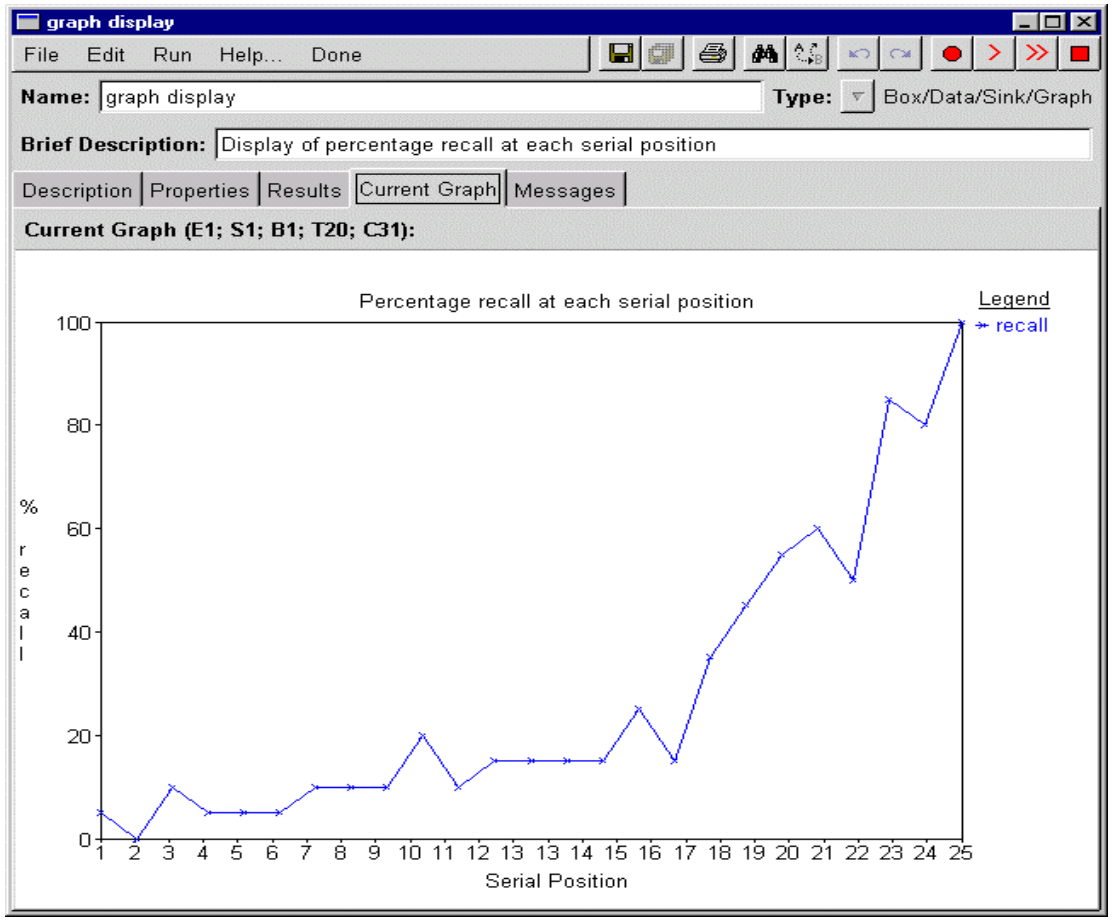


Building the Short Term Store: V

The rule to recall from *STS*:



Building the Short Term Store: VI



Building the Short Term Store: VII

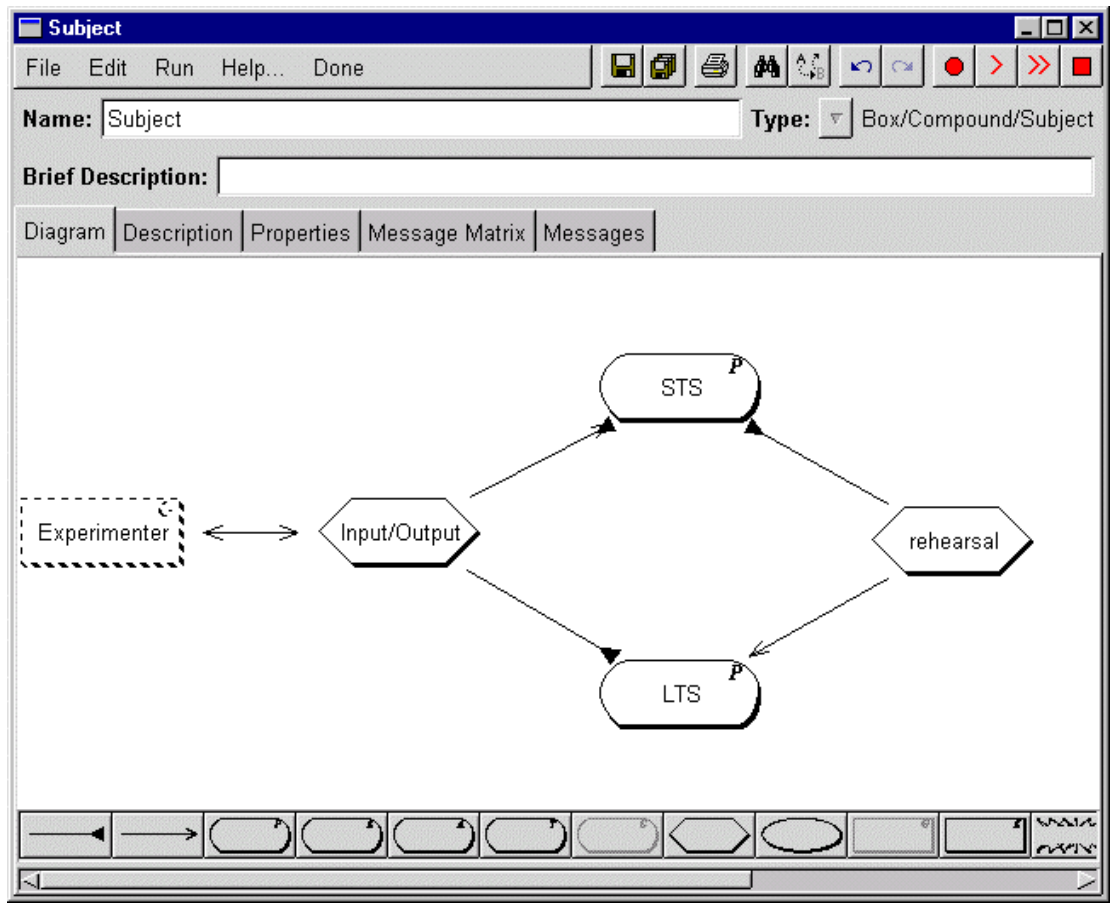
- Run more trials. What happens to the curve?
- Change the On Excess property of *STS*. What happens to the shape of the graph when you run a few trials?
- Watch the Messages view of *Input/Output*. What happens there now when you run (or single-step) through a trial?

Adding the Long Term Store: I

The modal model also includes:

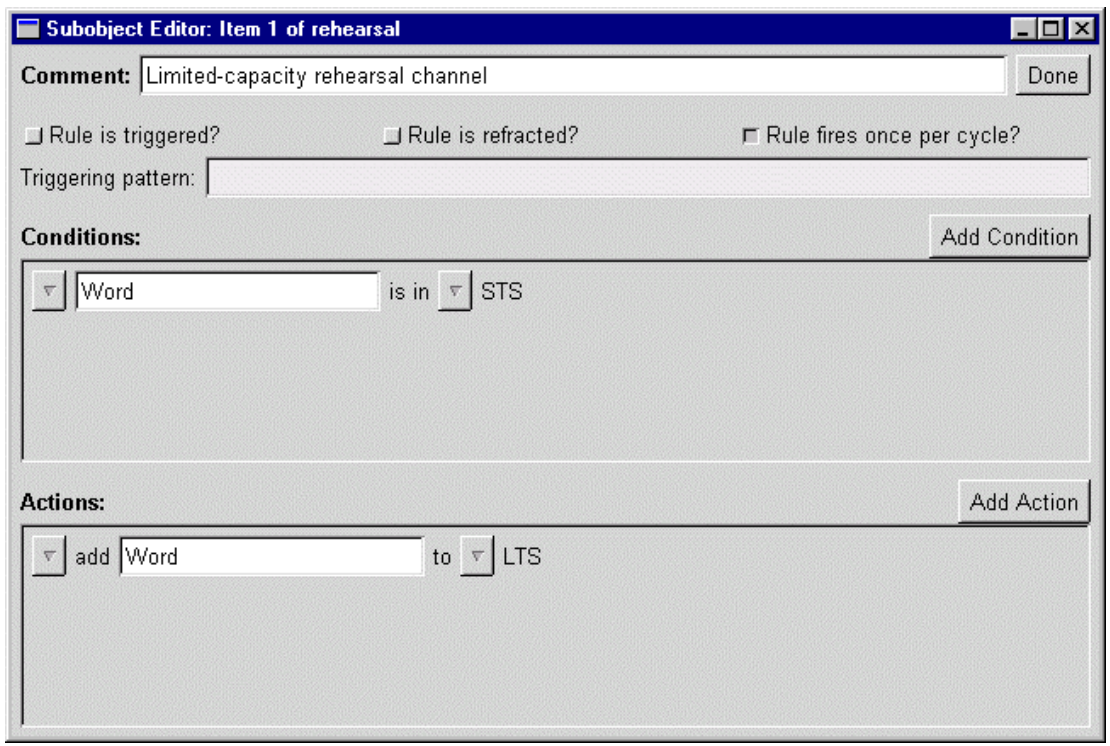
- a long term store (LTS);
- a rehearsal process to transfer information from STS to LTS; and
- the possibility to recall from either STS or LTS

Adding the Long Term Store: II



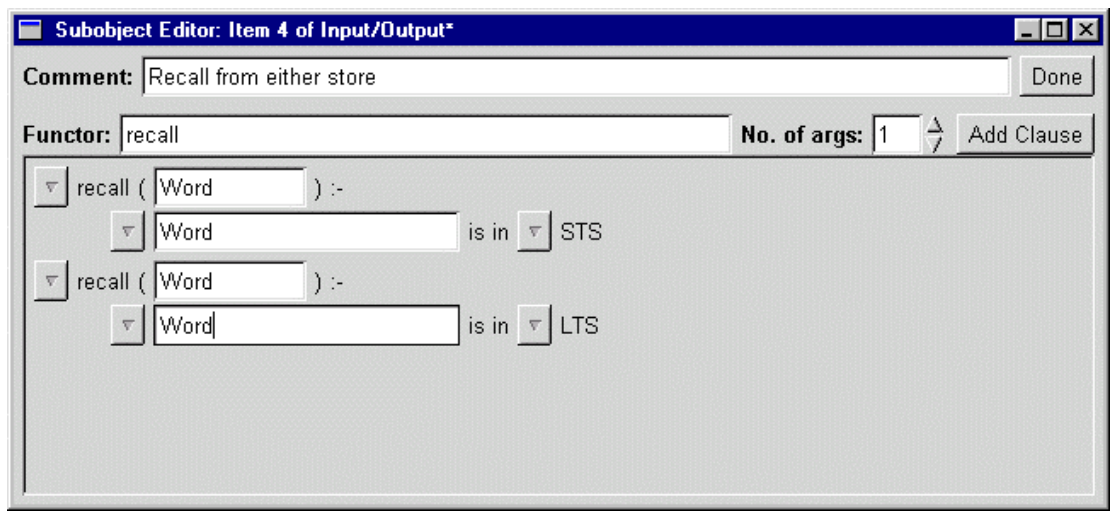
Adding the Long Term Store: III

The rehearsal rule:

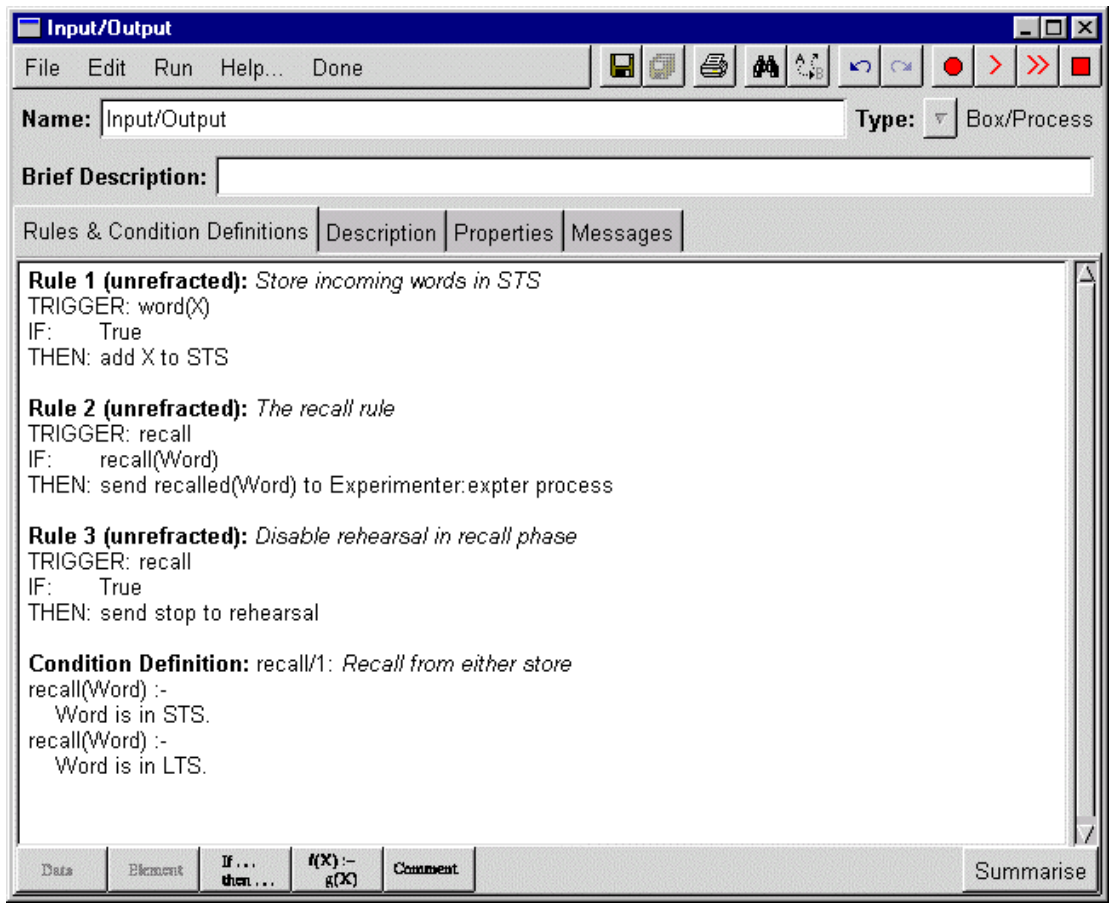


Adding the Long Term Store: IV

Recalling from either *STS* or *LTS*:



Adding the Long Term Store: V



Adding the Long Term Store: VI

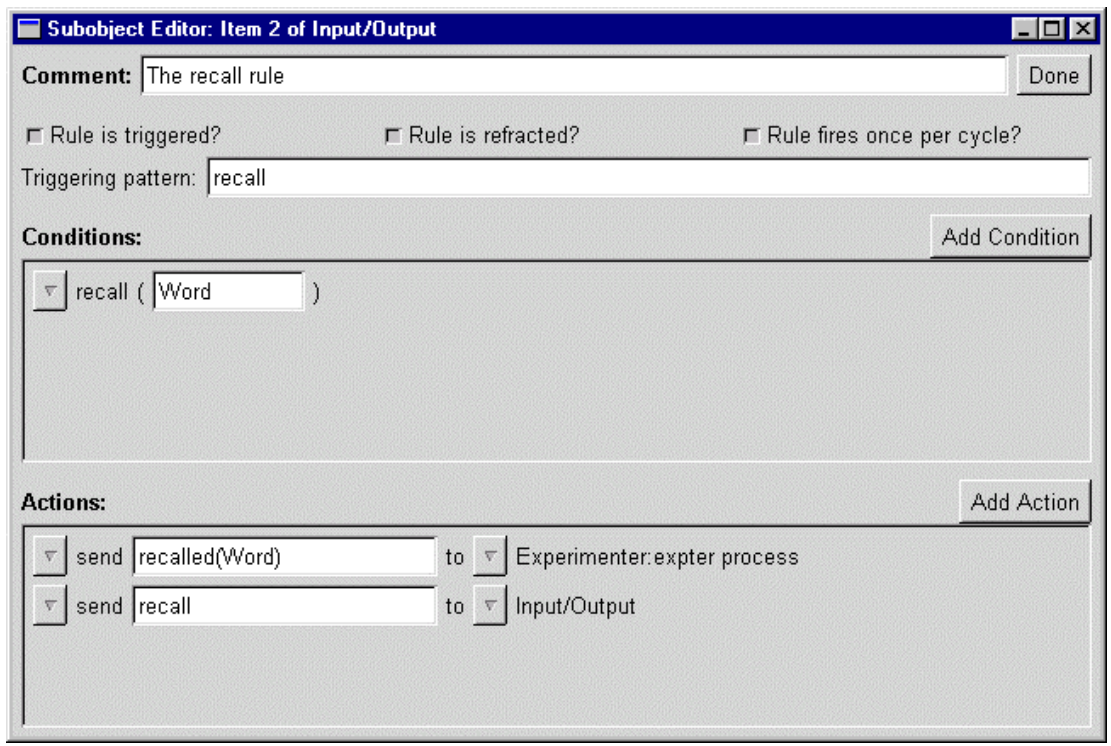
- What causes the Primacy Effect arise?
- Monitor the *Input/Output* box's Messages view. Why does the model sometimes recalls the same word twice in the same trial.
- The serial position curve still doesn't look like the one in the introduction. Characterise the difference. Can you account for it?

Decay, Time & Rehearsal: I

- Add decay to *LTS*. Explore different decay rates.
- Change the rehearsal rate by adding a copy of the rehearsal rule.
- All memorised words are currently recalled in parallel. Make the recall process serial.

Decay, Time & Rehearsal: II

The serial recall rule:



Decay, Time & Rehearsal: III

- Explore the effect of the Buffer Access property of each buffer. Play with these (and other) parameters to see how they affect the model's behaviour.
- The Experimenter system is written using standard COGENT. Try to discover how it works.
- Go on to develop the model into something substantial.

References

Atkinson, R. C., & Shiffrin, R. M. (1968). Human memory: A proposed system and its control processes. In Spence, K. W., & Spence, J. T. (Eds.), *The psychology of learning and motivation: Advances in research and theor*y. Academic Press, Orlando, FL.

Atkinson, R. C., & Shiffrin, R. M. (1971). The control of short term memory. *Scientific Americ*an, *22*5, 82–90.

Cooper, R., & Fox, J. (1998). COGENT: A visual design environment for cognitive modelling. *Behavior Research Methods, Instruments, & Computer*s, *3*0(4), 553–564.

Glanzer, M., & Cunitz, A. R. (1966). Two storage mechanisms in free recall. *Journal of Verbal Learning and Verbal Behavio*r, 5, 351–360.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Revie*w, *6*3, 81–97.

Young, R. M., & O'Shea, T. (1981). Errors in Children's Subtraction. *Cognitive Scienc*e, 5, 153–177.