EPIC Tutorial - Exercise 1
A. Hornof - 7/27/04

The goal of this exercise is to get familiar with the basic controls and functionality of EpicApp, the front end to EPIC. Check off each exercise after you complete it. Fill in the blanks.

__ 1. Run the Williams model using the Random_ID_Search.prs production rules. Run it ten steps at a time until objects appear. Then run it one step at a time. Watch for a trial or two.

__ 2. Run all trials with a single command. (Hint: Set Cycles/Run to a big number.)

__ 3. Set the model to run until it hits any of these three rules:
    Start_random_search
    Continue_random_search
    Make_response

__ 4. Running the model, note that the gaze moves in a somewhat random order, stopping on the target. You may need to zoom in a little on the layouts, or adjust the displayed size of the visual objects, to read all the ID label number of each object.

__ 5. Study the three visual windows. Note how the "Physical Vision" shows all objects and features, the "Sensory Vision" window shows just the information available to the eyes, and how object features linger in "Perceptual Vision" even after the eyes have moved away.

Questions: After the eyes move to a new location, how many cycles does it take for each of these features to become available in each visual window? Fill in the blanks with the number of cycles. (Hint: This will require you to single-step through the model.)

Feature	In Sensory Vision	In Perceptual Vision
Color	_____	_____
Shape	_____	_____
Text	_____	_____

What parameter determines the number of cycles to get from Sensory to Perceptual Vision?

_____

In which retinal zone is each feature available? Put a check box in each blank that applies. Assume each zone has the radius given here.

Feature	Fovea (1°)	Parafovea (7.5°)	Periphery (90°)
Visibility	__	__	__
Position	__	__	__
Color	__	__	__
Shape	__	__	__
Text	__	__	__

Why is the color sometimes not available within the big circle in Perceptual Vision?

_____

_____

__ 6. Set up the PPS Memory and Run messages side by side with the "Perceptual Vision" window. (Perhaps zoom in on the "Perceptual Vision".) Run the model one cycle at a time.

Note how the contents of Visual WM in the PPS execution correspond to the contents of the "Perceptual Vision" window on every cycle.

If you just study the PPS WM contents, how many cycles are required after the eyes move for you to figure out which Vpsychobj is the newly-fixated object?

_____

__ 7. Observe how three rules get instantiated and fire, starting with the firing of "Trigger_the_display_appearance". For each rule that fires, figure out what changed most recently in the PPS WM to allow it to fire. To do this, set EpicApp to show the PPS memory contents and run messages, and to pause every time that one of the four rules listed below rules fires. Open the PPS trace in the "Normal Output" window side by side with the production rules (which you open in a text editor). Run EpicApp until the next rule fires. Note that the LHS (left hand side, the IF side) of every rule that fires is satisfied by a clause in the PPS WM. Note any variable bindings--the values assigned to the variables.

Rule   *What changed to allow it to fire*

Trigger_the_display_appearance _____

Start_random_search _____

Continue_random_search _____
     OR
Target_found_so_stop_search _____

__ 8. Turn on the trace for the Ocular and Device processors. Note that the device effectively has an eye tracker running, and reports when the eyes move. Find a eye movement that occurs in the middle of a search. How much time elapses from when the Eye reports the start of a movement to when the Device reports the start of the eye movement? _____

How long did the eye movement last? _____

Going backward in time, how much time was needed to prepare the eye movement? Go back to when the Ocular processor received the command and started to prepare.

_____

__ 9. Change the simulated device so that 100 objects appear on the screen, as in the original Williams '67 task. The Williams '67 parameter string sets the number of trials in an experiment and the number of items that appear in the display. For example, the string "10 5" indicates ten trials and five objects. (Hint: Open "Run Controls ...") Run the model for a few trials. What do you notice?