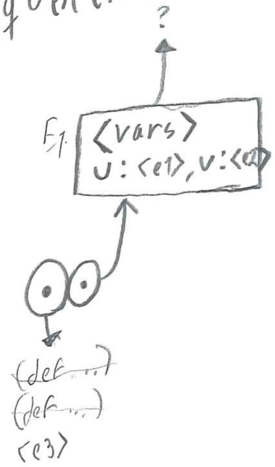
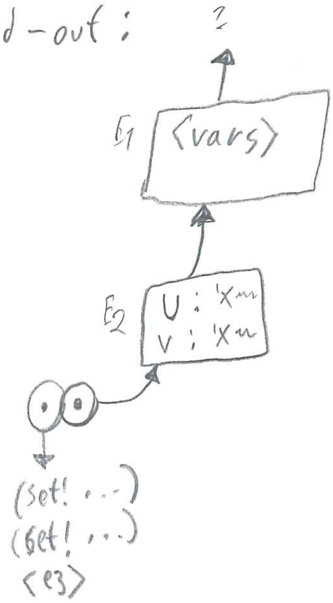
4.17 Draw diagrams for evaluating <e3> when done sequentially, and when scanned out as in 4.16.

Sequential:



Scanned-out:



Extra frame since procedures have their own frames, and let → λ, so really two nested lambdas, whereas defines (and sets) just update the bounding frame.

It can't make a difference in behaviour of a correct program because of variable look-up rules which gradually "reach up"! (Also "confinement" of the new environ.)

Simultaneous without extra frame:

```
(λ <vars>                          (λ <vars>
  (define u <e1>)        →          (set! u '*unassigned*)
  (define v <e2>)                   (set! v '*unassigned*)
  <e3>)                             (set! u <e1>)
                                    (set! v <e2>)
                                    <e3>)
```

(Or just have defines at the start of the function body?)