

DINING PHILOSOPHERS IN ANST ACTOR LANGUAGE

```
 $P_{phi} = rec(\lambda b. \lambda l. \lambda r. \lambda self. \lambda acks. \lambda m.$   
  if (eq?(m, self),  
    if (eq?(acks, 0),  
      become (b(l, r, self, acks + 1)),  
      seq (send (l, mkrelease (self)),  
          send (r, mkrelease (self)),  
          become (b (l, r, self, 0)),  
          send (l, mkpickup (self)),  
          send (r, mkpickup (self))))),  
  become (b (l, r, self, acks))))
```

DINING PHILOSOPHERS IN AMST (2)

$B_{\text{fork}} = \text{rec } (\lambda b. \lambda h. \lambda w. \lambda m.$

if (pickup?(m),

if (eq?(h, nil),

seq(send(phil(m), phil(n)),

become(b(phil(m), nil))),

become(b(h, phil(m))))),

if (release?(m),

if (eq?(w, nil),

become(b(nil, nil)),

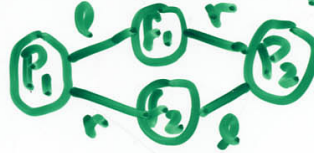
seq(send(w, w),

become(b(w, nil))))),

become(b(h, w))))))

DINING PHILOSOPHERS IN AMST (3)

Using the definitions to set up a 2-phil's dining table:



defactor { $f_1 := \text{Bfork}(\text{nil}, \text{nil}),$
 $f_2 := \text{Bfork}(\text{nil}, \text{nil}),$
 $P_1 := \text{Bphil}(f_1, f_2, P_1, \emptyset),$
 $P_2 := \text{Bphil}(f_2, f_1, P_2, \emptyset) \} e$

where e is defined as:

$e = \text{seq}(\text{send}(f_1, \text{mkpickup}(P_1)),$
 $\text{send}(f_2, \text{mkpickup}(P_1)),$
 $\text{send}(f_1, \text{mkpickup}(P_2)),$
 $\text{send}(f_2, \text{mkpickup}(P_2)))$

DINING PHILOSOPHERS IN ANST (4)

Auxiliary definitions:

$$\text{mkpickup} = \lambda p. \text{pr}(\text{"pickup"}, p)$$
$$\text{mkrelease} = \lambda p. \text{pr}(\text{"release"}, p)$$
$$\text{pickup?} = \lambda m. \text{if}(\text{ispr?}(m), \\ \text{eq?}(\text{1st}(m), \text{"pickup"}), \\ \text{nil})$$
$$\text{release?} = \lambda m. \text{if}(\text{ispr?}(m), \\ \text{eq?}(\text{1st}(m), \text{"release"}), \\ \text{nil})$$
$$\text{phil} = \lambda m. \text{if}(\text{pickup?}(m), \\ \text{2nd}(m), \\ \text{nil})$$

REFERENCE CELL IN ACTOR LANGUAGE

$$B_{\text{cell}} = \text{rec}(\lambda b. \lambda c. \lambda m. \\ \text{if}(\text{get?}(m), \\ \text{seq}(\text{become}(b(c)), \\ \text{send}(\text{cvt}(m), c)) \\ \text{if}(\text{set?}(m), \\ \text{become}(b(\text{contents}(m))), \\ \text{become}(b(c))))))$$

Using the cell:

$$\text{letfactor } \{a := B_{\text{cell}}(0)\} e$$
$$e = \text{seq}(\text{send}(a, \text{mkset}(3)), \\ \text{send}(a, \text{mkset}(5)), \\ \text{send}(a, \text{mkget}(e)))$$