Mobiles Ambients

- Locations represented by a topology of boundaries. Not identified with globally unique names.

- Process mobility represented as crossing of boundaries. Not as communication of processes or process names over channels.

- Security represented as inability/ability to cross boundaries.

- Interaction between processes is by shared location within a common boundary.
Mobile Ambients

n
names

P, Q ::= processes
  (\forall m) P
  O
  P \mid Q
  !P
  n[P] ambient
  M \cdot P action

M ::= capabilities
  in n
  out n
  open n

subjective moves

n[in m \cdot P] \mid n[R] \rightarrow m[n[P] \mid R]
m[n[\text{out } m \cdot P] \mid L] \rightarrow m[L] \mid n[P] R]
open m \cdot P \mid n[R] \rightarrow P \mid Q
ENTRY CAPABILITY

\[
n [\text{in} \, m, P | Q] \mid m[R] \\
\rightarrow m [n[P|Q] | R]
\]
Exit Capability

\[ m\{n[\text{out m.P/Q}]\} R \] \rightarrow \n \{P/Q\} \mid m \{R\} \]
Open Capability

\[ \text{open m.P | m[Q]} \rightarrow P \upharpoonright Q \]

\[ \text{open m.P | } \begin{array}{c} \text{m} \\ \text{Q} \end{array} \rightarrow P \upharpoonright Q \]
Objective Moves

\[ mv \text{ in } m.P \{ m[R] \} \rightarrow m.P[m[R]] \]
\[ m[\text{mv out } m.P[R]] \rightarrow P \mid m[R] \]

Ambient I/O

\[ (x).P \quad \text{input} \quad [\text{synchronous}] \]
\[ \langle x \rangle \quad \text{output} \quad [\text{asynchronous}] \]

\[ (x).P \mid \langle m \rangle \rightarrow P\{m/x\} \]
\[ P \{ x \leftarrow m \} \]
Encoding Sub-Active Moves w/ Sub-Active Moves

allow n ≡ !open n

mv in n . P ≡ (∀k) k ∈ n . enter [out k . open k . P]

n[k^][P] ≡ n[P | allow enter]

mv in n . P \ n[k^][Q] → !n[k^][P|Q]

Proof:

(∀k) \ k ∈ n . enter [out k . open k . P]

n[k^][Q] | open enter)

→ (∀k) \ n[k^][enter [out k . open k . P]] | Q | !open enter

(∀k) \ n[k^][enter [open k . P] | k[ ] | Q | !open enter]

(∀k) \ n[k^][open k . P | k[ ] | Q | !open enter]

n[k^][Q] = !n[k^][Q]
\[ \text{mv out n. P} \equiv \bot \]
\[(\forall k) K [\text{out n. exit [out k. open k. P]}] \]
\[ n^k [\text{P}] \equiv n [\text{P}] \text{ allow exit} \]

\[ n^k [\text{mv out n. P} \mid Q] \rightarrow n^k [\text{Q}] \mid P \]

Exercise:
prove
\[ n^k [\text{mv out n. P} \mid Q] \rightarrow n^k [\text{Q}] \mid P \]

\[ n^k [\text{P}] \equiv n [\text{P allow enter}] \mid \text{allow exit} \]
**REFERENCE CELL IN AMBIGUOUS CALVINUS**

\[
\begin{align*}
\text{cell } c \omega & \triangleq c''[\langle w \rangle] \\
\text{get } c \langle x \rangle. P & \triangleq \\
& \text{mv in } c \langle x \rangle. (\langle x \rangle \text{mv out } c. P) \\
\text{set } c \langle w \rangle. P & \triangleq \\
& \text{mv in } c \langle x \rangle. (\langle w \rangle \text{mv out } c. P)
\end{align*}
\]
Loks Example

\[
\text{acquire } n.P \equiv \text{open } n.P
\]
\[
\text{release } n.P \equiv n[\emptyset]P
\]

\[
\text{acquire } l.P \mid \text{release } l.Q = \text{open } l.P \mid l[\emptyset]Q \to
\]
\[
P \mid \emptyset Q \equiv P \mid Q
\]
phil l r \equiv ! (\text{acquire l, acquire r.} \\
\text{release l, release r}) \\
\text{fork f} \equiv \text{release f} \\
2\text{-table} \equiv \text{phil f1 f2} | \\
\text{phil f2 f1} | \\
\text{fork f1 | fork f2} \\
2\text{-table-nd} \equiv \text{phil f1 f2} | \\
\text{phil f1 f2} | \\
\text{fork f1 | fork f2} \\
3\text{-table} \equiv \text{phil f1 f2 | phil f2 f3 | phil f3 f1} | \\
\text{fork f1 | fork f2 | fork f3}