NOMADIC PICT

DECLARATIONS

type T = T'	type abbreviation
new c:T P	new channel name
agent a=P and ... and a'=P' in Q	agent creation
migrate to s P	agent migration
def f[...]=P and ... and f'[...]=P' Q	process abstraction

PROCESSES

(P | Q)	parallel composition
(D P)	local declaration
()	null process

COMMUNICATION

	$c!v$	output v on channel c in current agent
	$c?p = P$	input
	$c?{*}p = P$	replicated input
	if v then P else Q	conditional
LOCATION DEPENDENT	iflocal $\langle a\rangle c!v$ then P else Q	test-and-send to agent a on this site.
	$\langle a\rangle c!v$	send to agent a on this site.
	$\langle a@s\rangle c!v$	send to agent a on site s.
	wait $c?p=P$ timout $t \to Q$	input with timeout (secs)
	terminate	kill agent
	$c@a!v$	location-independent output to channel c at agent a.

NOMADIC PICT EXAMPLE

```
getApplet?*[a s] →
      agent b =
          migrate to s →
               (<a@s>ack!b|B
     in O
```

LOCATION-INDEPENDENT COMMUNICATION

```
<a@?>c!v

c@a!v
```

MOBILE AGENT EXAMPLE

```
new answer: ^String
def spawn [s: Site prompt: String] =
    (agent b =
        (migrate to s
         answer@a! (sys.read prompt))
      in
        ())
( spawn! [s1 "How are you?"]
| spawn! [s2 "When do we start?"]
| answer?* s = print! s
   ...
```

This code (part of agent a) spawns two agents at sites s1 and s2, and prints answers coming on its "answer" channel.

REFERENCE CELL IN NOMADIC PICT

```
type RefInt =
   [
     set = /[Agent Int Sig]
     get = /[Agent /Int]
   ]

def refInt [s:Site r:/RefInt] =
 ( new set : ^[Agent Int Sig]
   new get : ^[Agent /Int]

   agent refIntAg =
   (
     new contents : ^Int
     run contents ! 0
     migrate to s
     ( set?* [a:Agent v:Int c:Sig] =
          contents?_ = (contents!v | c@a![])
     | get?* [a:Agent res:/Int] =
          contents?v = (contents!v | res@a!v))
   )
 r![ set = \[a:Agent v:Int c:Sig] =
              set@refIntAg![a v c]
     get = \[a:Agent res:\Int] = get@refIntAg[a res]]
```

NOMADIC PICT REFERENCE CELL USAGE

```
val cell1 = (refInt s1)
val cell2 = (refInt s2)

agent a =
(
   (cell2.set a 5);
   (prNL (int.toString (cell1.get a))));
   (prNL (int.toString (cell2.get a))));
    ()
)
```