es-kind-sends-iff[11,40]

state ds
k:A sends [tg, $e.f(e)$:B] on l
$\equiv_{\text{def}}$ ((∀e:E. (kind(e) = rcv(l,tg)) ⇒ (valtype(e) ⊆r B))
& (∀x:Id. vartype(source(l);x) ⊆r $ds(x)$?Top)
& (∀e:E. (kind(e) = k) ⇒ (loc(e) = source(l)) ⇒ (valtype(e) ⊆r A)))
c∧ (∀e@source(l).
(kind(e) = k) ⇒ (↑isl($f(e)$)) ⇒ (∃e':E. (kind(e') = rcv(l,tg) & sender(e') = e))
& (∀e':E.
(kind(e') = rcv(l,tg))
⇒ ((kind(sender(e')) = k)
c∧ (↑isl(f(sender(e'))))
c∧ (val(e') = outl(f(sender(e'))))))
& (∀e':E.
(kind(e') = rcv(l,tg))
⇒ (∀e'':E.
(kind(e'') = rcv(l,tg)) ⇒ (sender(e'') = sender(e')) ⇒ (e'' = e'))))

clarification:

es-kind-sends-iff(es;k;A;l;tg;B;ds;$e.f(e)$)
$\equiv_{\text{def}}$ ((∀e:es-E(es). (es-kind(es; e) = rcv(l,tg) ∈ Knd) ⇒ (es-valtype(es; e) ⊆r B))
& (∀x:Id. es-vartype(es; source(l); x) ⊆r fpf-cap(ds;IdDeq;x;Top))
& (∀e:es-E(es).
(es-kind(es; e) = k ∈ Knd)
⇒ (es-loc(es; e) = source(l) ∈ Id)
⇒ (es-valtype(es; e) ⊆r A)))
c∧ (alle-at(es;source(l);e.(es-kind(es; e) = k ∈ Knd)
⇒ (↑isl($f(e)$))
⇒ (∃e':es-E(es)
(es-kind(es; e') = rcv(l,tg) ∈ Knd & es-sender(es; e') = e ∈ es-E(es))))
& (∀e':es-E(es).
(es-kind(es; e') = rcv(l,tg) ∈ Knd)
⇒ ((es-kind(es; es-sender(es; e')) = k ∈ Knd)
c∧ (↑isl(f(es-sender(es; e'))))
c∧ (es-val(es; e') = outl(f(es-sender(es; e'))) ∈ B)))
& (∀e':es-E(es).
(es-kind(es; e') = rcv(l,tg) ∈ Knd)
⇒ (∀e'':es-E(es).
(es-kind(es; e'') = rcv(l,tg) ∈ Knd)
⇒ (es-sender(es; e'') = es-sender(es; e') ∈ es-E(es))
⇒ (e'' = e' ∈ es-E(es)))))

http://www.nuprl.org/FDLcontent/p0_963683_/p58_320423_{es-kind-sends-iff}.html