```
s-sends-rule  THM

⊢  ∀i:Id, k:Knd, l:IdLnk, ds:x:Id fp-> Type, da:k:Knd fp-> Type,
    f:(tg:Id
       ×State(ds)→ma-valtype(da; k)→
       (da(rcv(l; tg))?Void List)) List.
   source(l) = i
   ⇒
   @i: ma-single-sends(ds; da; k; l; f) ∈ Dsys
   & (∀D:Dsys.
      @i: ma-single-sends(ds; da; k; l; f) ⊆ D
      ⇒
      D
      realizes es.(∀x:Id. vartype(i;x) ⊆r ds(x)?Top)
                  & (∀e:E.
                     loc(e) = i
                     ⇒
                     (valtype(e) ⊆r ma-valtype(da; kind(e))))
                  & (∀e:E.
                     isrcv(e)
                     ⇒
                     lnk(e) = l
                     ⇒
                     (valtype(e) ⊆r ma-valtype(da; kind(e))))
                  & (∀e:E.
                     loc(e) = i
                     ⇒
                     kind(e) = k
                     ⇒
                     (∃L:{e':E| isrcv(e') & lnk(e') = l } List.
                      (∀e':E.
                       (e' ∈ L)
                       ⟺
                       isrcv(e') & lnk(e') = l & sender(e') = e)
                      & (∀e1,e2:E. e1 before e2 ∈ L ⇒ (e1 <loc e2))
                      & map(λe'.<tag(e'),val(e')>;L)
                        =
                        tagged-list-messages(λz.
                        (z when e);val(e);f))))
  by NewSpecializeEsLemmaOn Thm: better_DASH_sends_DASH_rule
     ma-single-sends(ds; da; k; l; f)
     THEN
     Try (Unfold `ma-valtype` 0 THEN Complete Auto)
     THEN
     Try
     (RevHypSubst' -2 0
      THEN
      AllHyps (λh.BackThru h THEN Complete Auto))
     THEN
     Try (Analyze -1 THEN Unhide THEN ExRepD)
     THEN
     Subst' (rcv(l; tag(e')) = kind(e')) 0
     THEN
     Try BackThruSomeHyp
     THEN
```

```
    RevHypSubst' -1 0
    THEN
    MoveToConcl -2
    THEN
    Unfolds [`es-isrcv`;`es-lnk`;`es-tag`] 0
    THEN
    GenConclAtAddr [2;3]
    THEN
    Analyze -2
    THEN
    Analyze -2
    THEN
    Unfolds [`isrcv`;`lnk`;`tagof`] 0
    THEN
    Reduce 0
    THEN
    Fold `rcv` 0
END
```