```
es-locl-iff  THM

⊢  ∀the_es:ES, e,e':E.
|  (e <loc e') ⟺ ¬first(e') & e = pred(e') ∨ (e <loc pred(e'))
| by Use_ES_Axioms
|
1. the_es : ES
2. Trans e,e':E. (e <loc e')
3. SWellFounded((e <loc e'))
4. ∀e,e':E. loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
6. ∀e:E.
|  ¬first(e)
|  ⇒
|  (pred(e) <loc e) & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
7. ∀e:E. ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
8. Trans e,e':E. (e < e')
9. SWellFounded((e < e'))
10. ∀e:E.
|   isrcv(e)
|   ⇒
|   sends(lnk(e);sender(e))[index(e)] = msg(lnk(e);tag(e);val(e))
11. ∀e,e':E. (e <loc e') ⇒ (e < e')
12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
13. ∀e,e':E.
|   (e < e')
|   ⇒
|   ¬first(e') & (e < pred(e')) ∨ e = pred(e')
|   ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
16. ∀e,e':E.
|   isrcv(e)
|   ⇒
|   isrcv(e')
|   ⇒
|   lnk(e) = lnk(e')
|   ⇒
|   ((e <loc e')
|    ⟺
|    (sender(e) <loc sender(e'))
|    ∨ sender(e) = sender(e') & index(e)<index(e'))
17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
|   ∃e':E. isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
⊢  ∀e,e':E.
|  (e <loc e') ⟺ ¬first(e') & e = pred(e') ∨ (e <loc pred(e'))
| by (Unfold `and` 0 THEN Fold `cand` 0 THEN Analyze 0)
|    THEN
|    (Analyze 0)
|    THEN
|    (Analyze 0)
|    THEN
|    (Analyze 0)
|
|\
```

```
| 1. the_es : ES
| 2. Trans e,e':E. (e <loc e')
| 3. SWellFounded((e <loc e'))
| 4. ∀e,e':E. loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
| 5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
| 6. ∀e:E.
| |  ¬first(e)
| |  ⇒
| |  (pred(e) <loc e) & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
| 7. ∀e:E. ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
| 8. Trans e,e':E. (e < e')
| 9. SWellFounded((e < e'))
| 10. ∀e:E.
| |   isrcv(e)
| |   ⇒
| |   sends(lnk(e);sender(e))[index(e)] = msg(lnk(e);tag(e);val(e))
| 11. ∀e,e':E. (e <loc e') ⇒ (e < e')
| 12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
| 13. ∀e,e':E.
| |   (e < e')
| |   ⇒
| |   ¬first(e') & (e < pred(e')) ∨ e = pred(e')
| |   ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
| 14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
| 15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
| 16. ∀e,e':E.
| |   isrcv(e)
| |   ⇒
| |   isrcv(e')
| |   ⇒
| |   lnk(e) = lnk(e')
| |   ⇒
| |   ((e <loc e')
| |    ⟺
| |    (sender(e) <loc sender(e'))
| |    ∨ sender(e) = sender(e') & index(e)<index(e'))
| 17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
| |   ∃e':E. isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
| 18. e : E
| 19. e' : E
| 20. (e <loc e')
| ⊢  ¬first(e') & e = pred(e') ∨ (e <loc pred(e'))
| | by Assert (¬first(e'))
| |
| |\
| | .....assertion .....
| | 1. the_es : ES
| | 2. Trans e,e':E. (e <loc e')
| | 3. SWellFounded((e <loc e'))
| | 4. ∀e,e':E. loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
| | 5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
| | 6. ∀e:E.
| |    ¬first(e)
| |    ⇒
| |    (pred(e) <loc e) & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
```

```
| | 7. ∀e:E. ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
| | 8. Trans e,e':E. (e < e')
| | 9. SWellFounded((e < e'))
| | 10. ∀e:E.
| |     isrcv(e)
| |     ⇒
| |     sends(lnk(e);sender(e))[index(e)] = msg(lnk(e);tag(e);val(e))
| | 11. ∀e,e':E. (e <loc e') ⇒ (e < e')
| | 12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
| | 13. ∀e,e':E.
| |     (e < e')
| |     ⇒
| |     ¬first(e') & (e < pred(e')) ∨ e = pred(e')
| |     ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
| | 14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
| | 15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
| | 16. ∀e,e':E.
| |     isrcv(e)
| |     ⇒
| |     isrcv(e')
| |     ⇒
| |     lnk(e) = lnk(e')
| |     ⇒
| |     ((e <loc e')
| |      ⟺
| |      (sender(e) <loc sender(e'))
| |      ∨ sender(e) = sender(e') & index(e)<index(e'))
| | 17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
| |     ∃e':E.
| |     isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
| | 18. e : E
| | 19. e' : E
| | 20. (e <loc e')
| | ⊢  ¬first(e')
| |   by ((Analyze 0) THEN (InstHyp [e'] 5) THEN (Analyze -1)
| |       THEN
| |       (Analyze -2))
| |      THEN
| |      InstHyp [e] -1
| |
| .\
| . 1. the_es : ES
| . 2. Trans e,e':E. (e <loc e')
| . 3. SWellFounded((e <loc e'))
| . 4. ∀e,e':E. loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
| . 5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
| . 6. ∀e:E.
| . |  ¬first(e)
| . |  ⇒
| . |  (pred(e) <loc e) & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
| . 7. ∀e:E. ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
| . 8. Trans e,e':E. (e < e')
| . 9. SWellFounded((e < e'))
| . 10. ∀e:E.
| . |   isrcv(e)
```

```
| . |   ⇒
| . |   sends(lnk(e);sender(e))[index(e)] = msg(lnk(e);tag(e);val(e))
| . 11. ∀e,e':E. (e <loc e') ⇒ (e < e')
| . 12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
| . 13. ∀e,e':E.
| . |   (e < e')
| . |   ⇒
| . |   ¬first(e') & (e < pred(e')) ∨ e = pred(e')
| . |   ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
| . 14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
| . 15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
| . 16. ∀e,e':E.
| . |   isrcv(e)
| . |   ⇒
| . |   isrcv(e')
| . |   ⇒
| . |   lnk(e) = lnk(e')
| . |   ⇒
| . |   ((e <loc e')
| . |    ⟺
| . |    (sender(e) <loc sender(e'))
| . |    ∨ sender(e) = sender(e') & index(e)<index(e'))
| . 17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
| . |   ∃e':E.
| . |   isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
| . 18. e : E
| . 19. e' : E
| . 20. (e <loc e')
| . 21. ¬first(e')
| . ⊢  ¬first(e') & e = pred(e') ∨ (e <loc pred(e'))
| . | by InstHyp [e'] 6
| . |
| . 1. the_es : ES
| . 2. Trans e,e':E. (e <loc e')
| . 3. SWellFounded((e <loc e'))
| . 4. ∀e,e':E. loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
| . 5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
| . 6. ∀e:E.
| . |  ¬first(e)
| . |  ⇒
| . |  (pred(e) <loc e) & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
| . 7. ∀e:E. ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
| . 8. Trans e,e':E. (e < e')
| . 9. SWellFounded((e < e'))
| . 10. ∀e:E.
| . |   isrcv(e)
| . |   ⇒
| . |   sends(lnk(e);sender(e))[index(e)] = msg(lnk(e);tag(e);val(e))
| . 11. ∀e,e':E. (e <loc e') ⇒ (e < e')
| . 12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
| . 13. ∀e,e':E.
| . |   (e < e')
| . |   ⇒
| . |   ¬first(e') & (e < pred(e')) ∨ e = pred(e')
| . |   ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
```

```
| . 14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
| . 15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
| . 16. ∀e,e':E.
| . |   isrcv(e)
| . |   ⇒
| . |   isrcv(e')
| . |   ⇒
| . |   lnk(e) = lnk(e')
| . |   ⇒
| . |   ((e <loc e')
| . |    ⟺
| . |    (sender(e) <loc sender(e'))
| . |    ∨ sender(e) = sender(e') & index(e)<index(e'))
| . 17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
| . |   ∃e':E.
| . |   isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
| . 18. e : E
| . 19. e' : E
| . 20. (e <loc e')
| . 21. ¬first(e')
| . 22. (pred(e') <loc e')
| . |   & (∀e'@0:E. ¬((pred(e') <loc e'@0) & (e'@0 <loc e')))
| . ⊢  ¬first(e') & e = pred(e') ∨ (e <loc pred(e'))
| . | by Assert (¬(pred(e') <loc e))
| . |
| . |\
| . | .....assertion .....
| . | 1. the_es : ES
| . | 2. Trans e,e':E. (e <loc e')
| . | 3. SWellFounded((e <loc e'))
| . | 4. ∀e,e':E.
| . |    loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
| . | 5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
| . | 6. ∀e:E.
| . |    ¬first(e)
| . |    ⇒
| . |    (pred(e) <loc e)
| . |    & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
| . | 7. ∀e:E. ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
| . | 8. Trans e,e':E. (e < e')
| . | 9. SWellFounded((e < e'))
| . | 10. ∀e:E.
| . |     isrcv(e)
| . |     ⇒
| . |     sends(lnk(e);sender(e))[index(e)]
| . |     =
| . |     msg(lnk(e);tag(e);val(e))
| . | 11. ∀e,e':E. (e <loc e') ⇒ (e < e')
| . | 12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
| . | 13. ∀e,e':E.
| . |     (e < e')
| . |     ⇒
| . |     ¬first(e') & (e < pred(e')) ∨ e = pred(e')
| . |     ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
| . | 14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
```

```
| . | 15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
| . | 16. ∀e,e':E.
| . |     isrcv(e)
| . |     ⇒
| . |     isrcv(e')
| . |     ⇒
| . |     lnk(e) = lnk(e')
| . |     ⇒
| . |     ((e <loc e')
| . |      ⟺
| . |      (sender(e) <loc sender(e'))
| . |      ∨ sender(e) = sender(e') & index(e)<index(e'))
| . | 17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
| . |     ∃e':E.
| . |     isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
| . | 18. e : E
| . | 19. e' : E
| . | 20. (e <loc e')
| . | 21. ¬first(e')
| . | 22. (pred(e') <loc e')
| . |     & (∀e'@0:E. ¬((pred(e') <loc e'@0) & (e'@0 <loc e')))
| . | ⊢  ¬(pred(e') <loc e)
| . |   by Auto THEN InstHyp [e] -1 THEN ParallelOp -1
| . |
| . .\
| . . 1. the_es : ES
| . . 2. Trans e,e':E. (e <loc e')
| . . 3. SWellFounded((e <loc e'))
| . . 4. ∀e,e':E.
| . . |  loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
| . . 5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
| . . 6. ∀e:E.
| . . |  ¬first(e)
| . . |  ⇒
| . . |  (pred(e) <loc e)
| . . |  & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
| . . 7. ∀e:E. ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
| . . 8. Trans e,e':E. (e < e')
| . . 9. SWellFounded((e < e'))
| . . 10. ∀e:E.
| . . |   isrcv(e)
| . . |   ⇒
| . . |   sends(lnk(e);sender(e))[index(e)]
| . . |   =
| . . |   msg(lnk(e);tag(e);val(e))
| . . 11. ∀e,e':E. (e <loc e') ⇒ (e < e')
| . . 12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
| . . 13. ∀e,e':E.
| . . |   (e < e')
| . . |   ⇒
| . . |   ¬first(e') & (e < pred(e')) ∨ e = pred(e')
| . . |   ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
| . . 14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
| . . 15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
| . . 16. ∀e,e':E.
```

```
| . . |   isrcv(e)
| . . |   ⇒
| . . |   isrcv(e')
| . . |   ⇒
| . . |   lnk(e) = lnk(e')
| . . |   ⇒
| . . |   ((e <loc e')
| . . |    ⟺
| . . |    (sender(e) <loc sender(e'))
| . . |    ∨ sender(e) = sender(e') & index(e)<index(e'))
| . . 17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
| . . |   ∃e':E.
| . . |   isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
| . . 18. e : E
| . . 19. e' : E
| . . 20. (e <loc e')
| . . 21. ¬first(e')
| . . 22. (pred(e') <loc e')
| . . |   & (∀e'@0:E. ¬((pred(e') <loc e'@0) & (e'@0 <loc e')))
| . . 23. ¬(pred(e') <loc e)
| . . ⊢  ¬first(e') & e = pred(e') ∨ (e <loc pred(e'))
| . . | by InstHyp [e;pred(e')] 4 THEN Analyze -1 THEN Thin -1
| . . |    THEN
| . . |    Analyze -1
| . . |
| . . |\
| . . | .....antecedent .....
| . . | 1. the_es : ES
| . . | 2. Trans e,e':E. (e <loc e')
| . . | 3. SWellFounded((e <loc e'))
| . . | 4. ∀e,e':E.
| . . | |  loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
| . . | 5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
| . . | 6. ∀e:E.
| . . | |  ¬first(e)
| . . | |  ⇒
| . . | |  (pred(e) <loc e)
| . . | |  & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
| . . | 7. ∀e:E. ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
| . . | 8. Trans e,e':E. (e < e')
| . . | 9. SWellFounded((e < e'))
| . . | 10. ∀e:E.
| . . | |   isrcv(e)
| . . | |   ⇒
| . . | |   sends(lnk(e);sender(e))[index(e)]
| . . | |   =
| . . | |   msg(lnk(e);tag(e);val(e))
| . . | 11. ∀e,e':E. (e <loc e') ⇒ (e < e')
| . . | 12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
| . . | 13. ∀e,e':E.
| . . | |   (e < e')
| . . | |   ⇒
| . . | |   ¬first(e') & (e < pred(e')) ∨ e = pred(e')
| . . | |   ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
| . . | 14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
```

```
| . . | 15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
| . . | 16. ∀e,e':E.
| . . | |   isrcv(e)
| . . | |   ⇒
| . . | |   isrcv(e')
| . . | |   ⇒
| . . | |   lnk(e) = lnk(e')
| . . | |   ⇒
| . . | |   ((e <loc e')
| . . | |    ⟺
| . . | |    (sender(e) <loc sender(e'))
| . . | |    ∨ sender(e) = sender(e') & index(e)<index(e'))
| . . | 17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
| . . | |   ∃e':E.
| . . | |   isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
| . . | 18. e : E
| . . | 19. e' : E
| . . | 20. (e <loc e')
| . . | 21. ¬first(e')
| . . | 22. (pred(e') <loc e')
| . . | |   & (∀e'@0:E. ¬((pred(e') <loc e'@0) & (e'@0 <loc e')))
| . . | 23. ¬(pred(e') <loc e)
| . . | ⊢  loc(e) = loc(pred(e'))
| . . | | by RWW
| . . | |    Thm* ∀the_es:ES, e:E.
| . . | |         ¬first(e) ⇒ loc(pred(e)) = loc(e) ∈ Id
| . . | |    {es_DASH_loc_DASH_pred}
| . . | |    0
| . . | |
| . . | 1. the_es : ES
| . . | 2. Trans e,e':E. (e <loc e')
| . . | 3. SWellFounded((e <loc e'))
| . . | 4. ∀e,e':E.
| . . |    loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
| . . | 5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
| . . | 6. ∀e:E.
| . . |    ¬first(e)
| . . |    ⇒
| . . |    (pred(e) <loc e)
| . . |    & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
| . . | 7. ∀e:E. ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
| . . | 8. Trans e,e':E. (e < e')
| . . | 9. SWellFounded((e < e'))
| . . | 10. ∀e:E.
| . . |     isrcv(e)
| . . |     ⇒
| . . |     sends(lnk(e);sender(e))[index(e)]
| . . |     =
| . . |     msg(lnk(e);tag(e);val(e))
| . . | 11. ∀e,e':E. (e <loc e') ⇒ (e < e')
| . . | 12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
| . . | 13. ∀e,e':E.
| . . |     (e < e')
| . . |     ⇒
| . . |     ¬first(e') & (e < pred(e')) ∨ e = pred(e')
```

```
| . . |     ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
| . . | 14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
| . . | 15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
| . . | 16. ∀e,e':E.
| . . |     isrcv(e)
| . . |     ⇒
| . . |     isrcv(e')
| . . |     ⇒
| . . |     lnk(e) = lnk(e')
| . . |     ⇒
| . . |     ((e <loc e')
| . . |      ⟺
| . . |      (sender(e) <loc sender(e'))
| . . |      ∨ sender(e) = sender(e') & index(e)<index(e'))
| . . | 17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
| . . |     ∃e':E.
| . . |     isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
| . . | 18. e : E
| . . | 19. e' : E
| . . | 20. (e <loc e')
| . . | 21. ¬first(e')
| . . | 22. (pred(e') <loc e')
| . . |     & (∀e'@0:E. ¬((pred(e') <loc e'@0) & (e'@0 <loc e')))
| . . | 23. ¬(pred(e') <loc e)
| . . | ⊢  loc(e) = loc(e')  by BackThru 4 THEN Obvious
| . . |
| . . .\
| . . . 1. the_es : ES
| . . . 2. Trans e,e':E. (e <loc e')
| . . . 3. SWellFounded((e <loc e'))
| . . . 4. ∀e,e':E.
| . . . |  loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
| . . . 5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
| . . . 6. ∀e:E.
| . . . |  ¬first(e)
| . . . |  ⇒
| . . . |  (pred(e) <loc e)
| . . . |  & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
| . . . 7. ∀e:E. ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
| . . . 8. Trans e,e':E. (e < e')
| . . . 9. SWellFounded((e < e'))
| . . . 10. ∀e:E.
| . . . |   isrcv(e)
| . . . |   ⇒
| . . . |   sends(lnk(e);sender(e))[index(e)]
| . . . |   =
| . . . |   msg(lnk(e);tag(e);val(e))
| . . . 11. ∀e,e':E. (e <loc e') ⇒ (e < e')
| . . . 12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
| . . . 13. ∀e,e':E.
| . . . |   (e < e')
| . . . |   ⇒
| . . . |   ¬first(e') & (e < pred(e')) ∨ e = pred(e')
| . . . |   ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
| . . . 14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
```

```
| . . . 15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
| . . . 16. ∀e,e':E.
| . . . |   isrcv(e)
| . . . |   ⇒
| . . . |   isrcv(e')
| . . . |   ⇒
| . . . |   lnk(e) = lnk(e')
| . . . |   ⇒
| . . . |   ((e <loc e')
| . . . |    ⟺
| . . . |    (sender(e) <loc sender(e'))
| . . . |    ∨ sender(e) = sender(e') & index(e)<index(e'))
| . . . 17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
| . . . |   ∃e':E.
| . . . |   isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
| . . . 18. e : E
| . . . 19. e' : E
| . . . 20. (e <loc e')
| . . . 21. ¬first(e')
| . . . 22. (pred(e') <loc e')
| . . . |   & (∀e'@0:E. ¬((pred(e') <loc e'@0) & (e'@0 <loc e')))
| . . . 23. ¬(pred(e') <loc e)
| . . . 24. (e <loc pred(e')) ∨ e = pred(e') ∨ (pred(e') <loc e)
| . . . ⊢  ¬first(e') & e = pred(e') ∨ (e <loc pred(e'))
| . . . | by Analyze -1
| . . . |
| . . . |\
| . . . | 1. the_es : ES
| . . . | 2. Trans e,e':E. (e <loc e')
| . . . | 3. SWellFounded((e <loc e'))
| . . . | 4. ∀e,e':E.
| . . . |    loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
| . . . | 5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
| . . . | 6. ∀e:E.
| . . . |    ¬first(e)
| . . . |    ⇒
| . . . |    (pred(e) <loc e)
| . . . |    & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
| . . . | 7. ∀e:E.
| . . . |    ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
| . . . | 8. Trans e,e':E. (e < e')
| . . . | 9. SWellFounded((e < e'))
| . . . | 10. ∀e:E.
| . . . |     isrcv(e)
| . . . |     ⇒
| . . . |     sends(lnk(e);sender(e))[index(e)]
| . . . |     =
| . . . |     msg(lnk(e);tag(e);val(e))
| . . . | 11. ∀e,e':E. (e <loc e') ⇒ (e < e')
| . . . | 12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
| . . . | 13. ∀e,e':E.
| . . . |     (e < e')
| . . . |     ⇒
| . . . |     ¬first(e') & (e < pred(e')) ∨ e = pred(e')
| . . . |     ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
```

```
| . . . | 14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
| . . . | 15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
| . . . | 16. ∀e,e':E.
| . . . |     isrcv(e)
| . . . |     ⇒
| . . . |     isrcv(e')
| . . . |     ⇒
| . . . |     lnk(e) = lnk(e')
| . . . |     ⇒
| . . . |     ((e <loc e')
| . . . |      ⟺
| . . . |      (sender(e) <loc sender(e'))
| . . . |      ∨ sender(e) = sender(e') & index(e)<index(e'))
| . . . | 17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
| . . . |     ∃e':E.
| . . . |     isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
| . . . | 18. e : E
| . . . | 19. e' : E
| . . . | 20. (e <loc e')
| . . . | 21. ¬first(e')
| . . . | 22. (pred(e') <loc e')
| . . . | 23. ∀e'@0:E. ¬((pred(e') <loc e'@0) & (e'@0 <loc e'))
| . . . | 24. ¬(pred(e') <loc e)
| . . . | 25. (e <loc pred(e'))
| . . . | ⊢  e = pred(e') ∨ (e <loc pred(e'))  by Obvious
| . . . |
| . . . .\
| . . . . 1. the_es : ES
| . . . . 2. Trans e,e':E. (e <loc e')
| . . . . 3. SWellFounded((e <loc e'))
| . . . . 4. ∀e,e':E.
| . . . .    loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
| . . . . 5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
| . . . . 6. ∀e:E.
| . . . .    ¬first(e)
| . . . .    ⇒
| . . . .    (pred(e) <loc e)
| . . . .    & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
| . . . . 7. ∀e:E.
| . . . .    ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
| . . . . 8. Trans e,e':E. (e < e')
| . . . . 9. SWellFounded((e < e'))
| . . . . 10. ∀e:E.
| . . . .     isrcv(e)
| . . . .     ⇒
| . . . .     sends(lnk(e);sender(e))[index(e)]
| . . . .     =
| . . . .     msg(lnk(e);tag(e);val(e))
| . . . . 11. ∀e,e':E. (e <loc e') ⇒ (e < e')
| . . . . 12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
| . . . . 13. ∀e,e':E.
| . . . .     (e < e')

| . . . .     ⇒
| . . . .     ¬first(e') & (e < pred(e')) ∨ e = pred(e')
| . . . .     ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
```

```
| . . . . 14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
| . . . . 15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
| . . . . 16. ∀e,e':E.
| . . . .     isrcv(e)
| . . . .     ⇒
| . . . .     isrcv(e')
| . . . .     ⇒
| . . . .     lnk(e) = lnk(e')
| . . . .     ⇒
| . . . .     ((e <loc e')
| . . . .      ⟺
| . . . .      (sender(e) <loc sender(e'))
| . . . .      ∨ sender(e) = sender(e') & index(e)<index(e'))
| . . . . 17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
| . . . .     ∃e':E.
| . . . .     isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
| . . . . 18. e : E
| . . . . 19. e' : E
| . . . . 20. (e <loc e')
| . . . . 21. ¬first(e')
| . . . . 22. (pred(e') <loc e')
| . . . . 23. ∀e'@0:E. ¬((pred(e') <loc e'@0) & (e'@0 <loc e'))
| . . . . 24. ¬(pred(e') <loc e)
| . . . . 25. e = pred(e') ∨ (pred(e') <loc e)
| . . . . ⊢  e = pred(e') ∨ (e <loc pred(e'))
| . . . .   by Analyze -1 THEN Obvious
|
.\
. 1. the_es : ES
. 2. Trans e,e':E. (e <loc e')
. 3. SWellFounded((e <loc e'))
. 4. ∀e,e':E. loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
. 5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
. 6. ∀e:E.
. |  ¬first(e)
. |  ⇒
. |  (pred(e) <loc e) & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
. 7. ∀e:E. ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
. 8. Trans e,e':E. (e < e')
. 9. SWellFounded((e < e'))
. 10. ∀e:E.
. |   isrcv(e)
. |   ⇒
. |   sends(lnk(e);sender(e))[index(e)] = msg(lnk(e);tag(e);val(e))
. 11. ∀e,e':E. (e <loc e') ⇒ (e < e')
. 12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
. 13. ∀e,e':E.
. |   (e < e')
. |   ⇒
. |   ¬first(e') & (e < pred(e')) ∨ e = pred(e')
. |   ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
. 14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
. 15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
. 16. ∀e,e':E.
. |   isrcv(e)
```

```
. |   ⇒
. |   isrcv(e')
. |   ⇒
. |   lnk(e) = lnk(e')
. |   ⇒
. |   ((e <loc e')
. |    ⟺
. |    (sender(e) <loc sender(e'))
. |    ∨ sender(e) = sender(e') & index(e)<index(e'))
. 17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
. |   ∃e':E. isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
. 18. e : E
. 19. e' : E
. 20. ¬first(e') & e = pred(e') ∨ (e <loc pred(e'))
. ⊢  (e <loc e')  by Analyze -1 THEN InstHyp [e'] 6 THEN Analyze -3
. |
. |\
. | 1. the_es : ES
. | 2. Trans e,e':E. (e <loc e')
. | 3. SWellFounded((e <loc e'))
. | 4. ∀e,e':E. loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
. | 5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
. | 6. ∀e:E.
. |    ¬first(e)
. |    ⇒
. |    (pred(e) <loc e) & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
. | 7. ∀e:E. ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
. | 8. Trans e,e':E. (e < e')
. | 9. SWellFounded((e < e'))
. | 10. ∀e:E.
. |     isrcv(e)
. |     ⇒
. |     sends(lnk(e);sender(e))[index(e)] = msg(lnk(e);tag(e);val(e))
. | 11. ∀e,e':E. (e <loc e') ⇒ (e < e')
. | 12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
. | 13. ∀e,e':E.
. |     (e < e')
. |     ⇒
. |     ¬first(e') & (e < pred(e')) ∨ e = pred(e')
. |     ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
. | 14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
. | 15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
. | 16. ∀e,e':E.
. |     isrcv(e)
. |     ⇒
. |     isrcv(e')
. |     ⇒
. |     lnk(e) = lnk(e')
. |     ⇒
. |     ((e <loc e')
. |      ⟺
. |      (sender(e) <loc sender(e'))
. |      ∨ sender(e) = sender(e') & index(e)<index(e'))
. | 17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
. |     ∃e':E.
```

```
. |     isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
. | 18. e : E
. | 19. e' : E
. | 20. ¬first(e')
. | 21. e = pred(e')
. | 22. (pred(e') <loc e')
. | 23. ∀e'@0:E. ¬((pred(e') <loc e'@0) & (e'@0 <loc e'))
. | ⊢  (e <loc e')  by HypSubst -3 0
. |
. .\
. . 1. the_es : ES
. . 2. Trans e,e':E. (e <loc e')
. . 3. SWellFounded((e <loc e'))
. . 4. ∀e,e':E. loc(e) = loc(e') ⟺ (e <loc e') ∨ e = e' ∨ (e' <loc e)
. . 5. ∀e:E. first(e) ⟺ (∀e':E. ¬(e' <loc e))
. . 6. ∀e:E.
. .    ¬first(e)
. .    ⇒
. .    (pred(e) <loc e) & (∀e':E. ¬((pred(e) <loc e') & (e' <loc e)))
. . 7. ∀e:E. ¬first(e) ⇒ (∀x:Id. (x when e) = (x after pred(e)))
. . 8. Trans e,e':E. (e < e')
. . 9. SWellFounded((e < e'))
. . 10. ∀e:E.
. .     isrcv(e)
. .     ⇒
. .     sends(lnk(e);sender(e))[index(e)] = msg(lnk(e);tag(e);val(e))
. . 11. ∀e,e':E. (e <loc e') ⇒ (e < e')
. . 12. ∀e:E. isrcv(e) ⇒ (sender(e) < e)
. . 13. ∀e,e':E.
. .     (e < e')
. .     ⇒
. .     ¬first(e') & (e < pred(e')) ∨ e = pred(e')
. .     ∨ isrcv(e') & (e < sender(e')) ∨ e = sender(e')
. . 14. ∀e:E. isrcv(e) ⇒ loc(e) = destination(lnk(e))
. . 15. ∀e:E, l:IdLnk. ¬loc(e) = source(l) ⇒ sends(l;e) = nil
. . 16. ∀e,e':E.
. .     isrcv(e)
. .     ⇒
. .     isrcv(e')
. .     ⇒
. .     lnk(e) = lnk(e')
. .     ⇒
. .     ((e <loc e')
. .      ⟺
. .      (sender(e) <loc sender(e'))
. .      ∨ sender(e) = sender(e') & index(e)<index(e'))
. . 17. ∀e:E, l:IdLnk, n:ℕ||sends(l;e)||.
. .     ∃e':E.
. .     isrcv(e') & lnk(e') = l & sender(e') = e & index(e') = n
. . 18. e : E
. . 19. e' : E
. . 20. ¬first(e')
. . 21. (e <loc pred(e'))
. . 22. (pred(e') <loc e')
. . 23. ∀e'@0:E. ¬((pred(e') <loc e'@0) & (e'@0 <loc e'))
```

```
. . ⊢  (e <loc e')  by UseTrans pred(e')
END
```