```
ma-compatible-join  THM

⊢  ∀A,B,C:MsgA.
|  A || B
|  ⇒
|  ma-frame-compatible(A; B)
|  ⇒
|  ma-sframe-compatible(A; B)
|  ⇒
|  C || A
|  ⇒
|  ma-frame-compatible(C; A)
|  ⇒
|  ma-sframe-compatible(C; A)
|  ⇒
|  C || B
|  ⇒
|  ma-frame-compatible(C; B)
|  ⇒
|  ma-sframe-compatible(C; B)
|  ⇒
|  C || A ⊕ B
|  & ma-frame-compatible(C; A ⊕ B)
|  & ma-sframe-compatible(C; A ⊕ B)
| by Auto THEN Unfold_MsgA -12 THEN Unfold_MsgA -11
|    THEN
|    Unfold_MsgA -10
|    THEN
|    All
|    (Unfolds
|     [‘ma-compatible‘;‘ma-frame-compatible‘;‘ma-sframe-compatible‘])
|    THEN
|    All (Unfolds [‘ma-compatible-decls‘])
|    THEN
|    All Reduce
|    THEN
|    Unfold ‘guard‘ 0
|    THEN
|    SplitAndConcl
|    THEN
|    Try
|    (BackThru Thm: fpf_DASH_compatible_DASH_join THEN Complete Auto)
|    THEN
|    Try
|    (Complete
|     (Auto
|      THEN
|      Inst Thm: fpf_DASH_compatible_DASH_triple [Id;IdDeq;ds;d1;d3]
|      THEN
|      Inst Thm: fpf_DASH_compatible_DASH_triple
|      [Knd;KindDeq;da;d2;d4]
|      THEN
|      Try
|      (Unfold ‘fpf-compatible‘ 0 THEN DupHyp -1
|       THEN
```

```
|       RWO
|       Thm* ∀eq:EqDecider(A), f,g:a:A fp-> Top, x:A.
|            x ∈ dom(f ⊕ g) ⟺ x ∈ dom(f) ∨ x ∈ dom(g)
|       {fpf_DASH_join_DASH_dom2}
|       -1
|       THEN
|       RWO
|       Thm* ∀eq:EqDecider(A), f:a:A fp-> Top, g:Top, x:A.
|            f ⊕ g(x) ∼ if x ∈ dom(f)→ f(x) else g(x) fi
|       {fpf_DASH_join_DASH_ap_DASH_sq}
|       0
|       THEN
|       SplitOnConclITE
|       THEN
|       SplitOrHyps
|       THEN
|       AllHyps
|       (λh.
|        Unfold `fpf-compatible` h THEN InstHyp [x] h
|        THENA
|        Complete Auto)
|       THENL
|       [Id;Thin -1;Id]
|       THEN
|       HypSubst -1 0
|       THEN
|       Try (Fold `member` 0)
|       THEN
|       Try (Analyze 0 THEN DoSubsume)
|       THEN
|       Try DoSubsume
|       THEN
|       Try (Unfold `ma-valtype` 0)
|       THEN
|       Try
|       (BackThru
|        Thm* ∀ds,ds':ltg:Id fp-> Type.
|             ds ⊆ ds' ⇒ (State(ds') ⊆r State(ds))
|        {ma_DASH_state_DASH_subtype})
|       THEN
|       Try
|       (BackThru
|        Thm* ∀eq:EqDecider(X), f,g:x:X fp-> Type, x:X.
|             g ⊆ f ⇒ (f(x)?T ⊆r g(x)?Top)
|        {subtype_DASH_fpf_DASH_cap_DASH_top})
|       THEN
|       Try
|       (BackThru
|        Thm* ∀eq:EqDecider(X), f,g:x:X fp-> Type, x:X.
|             f ⊆ g ⇒ (f(x)?Void ⊆r g(x)?T)
|        {subtype_DASH_fpf_DASH_cap_DASH_void})
|       THEN
|       Try
|       (BackThru
|        Thm* ∀B1,B2:(A→Type), eq:EqDecider(A), f:a:A fp-> B1(a),
```

```
|              g:a:A fp-> B2(a). f ⊆ f ⊕ g
|        {fpf_DASH_sub_DASH_join_DASH_left}))))
|
|\
| 1. ds : x:Id fp-> Type
| 2. da : a:Knd fp-> Type
| 3. init : x:Id fp-> ds(x)?Void
| 4. pre : a:Id fp-> State(ds)→da(locl(a))?Top→Prop
| 5. ef : kx:Knd×Id fp-> State(ds)→da(1of(kx))?Top→ds(2of(kx))?Void
| 6. send :
|    kl:Knd×IdLnk fp-> (tg:Id
|                       ×State(ds)→da(1of(kl))?Top→
|                       (da(rcv(2of(kl); tg))?Void List)) List
| 7. frame : x:Id fp-> Knd List
| 8. sframe : ltg:IdLnk×Id fp-> Knd List
| 9. Top
| 10. d1 : x:Id fp-> Type
| 11. d2 : a:Knd fp-> Type
| 12. i1 : x:Id fp-> d1(x)?Void
| 13. p1 : a:Id fp-> State(d1)→d2(locl(a))?Top→Prop
| 14. e1 : kx:Knd×Id fp-> State(d1)→d2(1of(kx))?Top→d1(2of(kx))?Void
| 15. s1 :
|     kl:Knd×IdLnk fp-> (tg:Id
|                        ×State(d1)→d2(1of(kl))?Top→
|                        (d2(rcv(2of(kl); tg))?Void List)) List
| 16. f1 : x:Id fp-> Knd List
| 17. s2 : ltg:IdLnk×Id fp-> Knd List
| 18. Top
| 19. d3 : x:Id fp-> Type
| 20. d4 : a:Knd fp-> Type
| 21. i2 : x:Id fp-> d3(x)?Void
| 22. p2 : a:Id fp-> State(d3)→d4(locl(a))?Top→Prop
| 23. e2 : kx:Knd×Id fp-> State(d3)→d4(1of(kx))?Top→d3(2of(kx))?Void
| 24. s3 :
|     kl:Knd×IdLnk fp-> (tg:Id
|                        ×State(d3)→d4(1of(kl))?Top→
|                        (d4(rcv(2of(kl); tg))?Void List)) List
| 25. f2 : x:Id fp-> Knd List
| 26. s4 : ltg:IdLnk×Id fp-> Knd List
| 27. Top
| 28. ds || d1 & da || d2
|     & init || i1
|     & pre || p1
|     & ef || e1
|     & send || s1
|     & frame || f1
|     & sframe || s2
| 29. ∀kx:(Knd×Id).
|     (kx ∈ dom(ef)
|      ⇒
|      ¬2of(kx) ∈ dom(frame)
|      ⇒
|      2of(kx) ∈ dom(f1) ⇒ deq-member(KindDeq;1of(kx);f1(2of(kx))))
|     & (kx ∈ dom(e1)
|        ⇒
```

```
|        ¬2of(kx) ∈ dom(f1)
|        ⇒
|        2of(kx) ∈ dom(frame)
|        ⇒
|        deq-member(KindDeq;1of(kx);frame(2of(kx))))
| 30. ∀kl:(Knd×IdLnk), tg:Id.
|     (kl ∈ dom(send)
|      ⇒
|      (tg ∈ map(λp.1of(p);send(kl)))
|      ⇒
|      ¬<2of(kl),tg> ∈ dom(sframe)
|      ⇒
|      <2of(kl),tg> ∈ dom(s2)
|      ⇒
|      deq-member(KindDeq;1of(kl);s2(<2of(kl),tg>)))
|     & (kl ∈ dom(s1)
|        ⇒
|        (tg ∈ map(λp.1of(p);s1(kl)))
|        ⇒
|        ¬<2of(kl),tg> ∈ dom(s2)
|        ⇒
|        <2of(kl),tg> ∈ dom(sframe)
|        ⇒
|        deq-member(KindDeq;1of(kl);sframe(<2of(kl),tg>)))
| 31. d3 || ds & d4 || da
|     & i2 || init
|     & p2 || pre
|     & e2 || ef
|     & s3 || send
|     & f2 || frame
|     & s4 || sframe
| 32. ∀kx:(Knd×Id).
|     (kx ∈ dom(e2)
|      ⇒
|      ¬2of(kx) ∈ dom(f2)
|      ⇒
|      2of(kx) ∈ dom(frame)
|      ⇒
|      deq-member(KindDeq;1of(kx);frame(2of(kx))))
|     & (kx ∈ dom(ef)
|        ⇒
|        ¬2of(kx) ∈ dom(frame)
|        ⇒
|        2of(kx) ∈ dom(f2) ⇒ deq-member(KindDeq;1of(kx);f2(2of(kx))))
| 33. ∀kl:(Knd×IdLnk), tg:Id.
|     (kl ∈ dom(s3)
|      ⇒
|      (tg ∈ map(λp.1of(p);s3(kl)))
|      ⇒
|      ¬<2of(kl),tg> ∈ dom(s4)
|      ⇒
|      <2of(kl),tg> ∈ dom(sframe)
|      ⇒
|      deq-member(KindDeq;1of(kl);sframe(<2of(kl),tg>)))
|     & (kl ∈ dom(send)
```

```
|        ⇒
|        (tg ∈ map(λp.1of(p);send(kl)))
|        ⇒
|        ¬<2of(kl),tg> ∈ dom(sframe)
|        ⇒
|        <2of(kl),tg> ∈ dom(s4)
|        ⇒
|        deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>)))
| 34. d3 || d1 & d4 || d2
|     & i2 || i1
|     & p2 || p1
|     & e2 || e1
|     & s3 || s1
|     & f2 || f1
|     & s4 || s2
| 35. ∀kx:(Knd×Id).
|     (kx ∈ dom(e2)
|      ⇒
|      ¬2of(kx) ∈ dom(f2)
|      ⇒
|      2of(kx) ∈ dom(f1) ⇒ deq-member(KindDeq;1of(kx);f1(2of(kx))))
|     & (kx ∈ dom(e1)
|        ⇒
|        ¬2of(kx) ∈ dom(f1)
|        ⇒
|        2of(kx) ∈ dom(f2) ⇒ deq-member(KindDeq;1of(kx);f2(2of(kx))))
| 36. ∀kl:(Knd×IdLnk), tg:Id.
|     (kl ∈ dom(s3)
|      ⇒
|      (tg ∈ map(λp.1of(p);s3(kl)))
|      ⇒
|      ¬<2of(kl),tg> ∈ dom(s4)
|      ⇒
|      <2of(kl),tg> ∈ dom(s2)
|      ⇒
|      deq-member(KindDeq;1of(kl);s2(<2of(kl),tg>)))
|     & (kl ∈ dom(s1)
|        ⇒
|        (tg ∈ map(λp.1of(p);s1(kl)))
|        ⇒
|        ¬<2of(kl),tg> ∈ dom(s2)
|        ⇒
|        <2of(kl),tg> ∈ dom(s4)
|        ⇒
|        deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>)))
| ⊢  ∀kx:(Knd×Id).
|    (kx ∈ dom(e2)
|     ⇒
|     ¬2of(kx) ∈ dom(f2)
|     ⇒
|     2of(kx) ∈ dom(frame ⊕ f1)
|     ⇒
|     deq-member(KindDeq;1of(kx);frame ⊕ f1(2of(kx))))
|    & (kx ∈ dom(ef ⊕ e1)
|       ⇒
```

```
|       ¬2of(kx) ∈ dom(frame ⊕ f1)
|       ⇒
|       2of(kx) ∈ dom(f2) ⇒ deq-member(KindDeq;1of(kx);f2(2of(kx))))
|   by Auto
|      THEN
|      Try
|      (RWO
|       Thm* ∀eq:EqDecider(A), f,g:a:A fp-> Top, x:A.
|            x ∈ dom(f ⊕ g) ⟺ x ∈ dom(f) ∨ x ∈ dom(g)
|       {fpf_DASH_join_DASH_dom2}
|       -3
|       THEN
|       Analyze -3
|       THEN
|       AllHyps
|       (λh.
|        BackThru h THEN ParallelOp -2
|        THEN
|        RWO
|        Thm* ∀eq:EqDecider(A), f,g:a:A fp-> Top, x:A.
|             x ∈ dom(f ⊕ g) ⟺ x ∈ dom(f) ∨ x ∈ dom(g)
|        {fpf_DASH_join_DASH_dom2}
|        0
|        THEN
|        ((OrRight THEN Complete Auto)
|         ORELSE
|         (OrLeft THEN Complete Auto))))
|      THEN
|      Try
|      (RWO
|       Thm* ∀eq:EqDecider(A), f:a:A fp-> Top, g:Top, x:A.
|            f ⊕ g(x) ∼ if x ∈ dom(f)→ f(x) else g(x) fi
|       {fpf_DASH_join_DASH_ap_DASH_sq}
|       0
|       THEN
|       RWO
|       Thm* ∀B:(A→Type), eq:EqDecider(A), f,g:a:A fp-> B(a), x:A.
|            x ∈ dom(f ⊕ g) ⟺ x ∈ dom(f) ∨ x ∈ dom(g)
|       {fpf_DASH_join_DASH_dom}
|       -1
|       THEN
|       SplitOnConclITE
|       THEN
|       BackThruSomeHyp
|       THEN
|       SplitOrHyps)
|
.\
. 1. ds : x:Id fp-> Type
. 2. da : a:Knd fp-> Type
. 3. init : x:Id fp-> ds(x)?Void
. 4. pre : a:Id fp-> State(ds)→da(locl(a))?Top→Prop
. 5. ef : kx:Knd×Id fp-> State(ds)→da(1of(kx))?Top→ds(2of(kx))?Void
. 6. send :
. |  kl:Knd×IdLnk fp-> (tg:Id
```

```
. |                    ×State(ds)→da(1of(kl))?Top→
. |                    (da(rcv(2of(kl); tg))?Void List)) List
. 7. frame : x:Id fp-> Knd List
. 8. sframe : ltg:IdLnk×Id fp-> Knd List
. 9. Top
. 10. d1 : x:Id fp-> Type
. 11. d2 : a:Knd fp-> Type
. 12. i1 : x:Id fp-> d1(x)?Void
. 13. p1 : a:Id fp-> State(d1)→d2(locl(a))?Top→Prop
. 14. e1 : kx:Knd×Id fp-> State(d1)→d2(1of(kx))?Top→d1(2of(kx))?Void
. 15. s1 :
. |   kl:Knd×IdLnk fp-> (tg:Id
. |                    ×State(d1)→d2(1of(kl))?Top→
. |                    (d2(rcv(2of(kl); tg))?Void List)) List
. 16. f1 : x:Id fp-> Knd List
. 17. s2 : ltg:IdLnk×Id fp-> Knd List
. 18. Top
. 19. d3 : x:Id fp-> Type
. 20. d4 : a:Knd fp-> Type
. 21. i2 : x:Id fp-> d3(x)?Void
. 22. p2 : a:Id fp-> State(d3)→d4(locl(a))?Top→Prop
. 23. e2 : kx:Knd×Id fp-> State(d3)→d4(1of(kx))?Top→d3(2of(kx))?Void
. 24. s3 :
. |   kl:Knd×IdLnk fp-> (tg:Id
. |                    ×State(d3)→d4(1of(kl))?Top→
. |                    (d4(rcv(2of(kl); tg))?Void List)) List
. 25. f2 : x:Id fp-> Knd List
. 26. s4 : ltg:IdLnk×Id fp-> Knd List
. 27. Top
. 28. ds || d1 & da || d2
. |   & init || i1
. |   & pre || p1
. |   & ef || e1
. |   & send || s1
. |   & frame || f1
. |   & sframe || s2
. 29. ∀kx:(Knd×Id).
. |   (kx ∈ dom(ef)
. |    ⇒
. |    ¬2of(kx) ∈ dom(frame)
. |    ⇒
. |    2of(kx) ∈ dom(f1) ⇒ deq-member(KindDeq;1of(kx);f1(2of(kx))))
. |   & (kx ∈ dom(e1)
. |      ⇒
. |      ¬2of(kx) ∈ dom(f1)
. |      ⇒
. |      2of(kx) ∈ dom(frame)
. |      ⇒
. |      deq-member(KindDeq;1of(kx);frame(2of(kx))))
. 30. ∀kl:(Knd×IdLnk), tg:Id.
. |   (kl ∈ dom(send)
. |    ⇒
. |    (tg ∈ map(λp.1of(p);send(kl)))
. |    ⇒
. |    ¬<2of(kl),tg> ∈ dom(sframe)
```

```
. |    ⇒
. |    <2of(kl),tg> ∈ dom(s2)
. |    ⇒
. |    deq-member(KindDeq;1of(kl);s2(<2of(kl),tg>))
. |   & (kl ∈ dom(s1)
. |      ⇒
. |      (tg ∈ map(λp.1of(p);s1(kl)))
. |      ⇒
. |      ¬<2of(kl),tg> ∈ dom(s2)
. |      ⇒
. |      <2of(kl),tg> ∈ dom(sframe)
. |      ⇒
. |      deq-member(KindDeq;1of(kl);sframe(<2of(kl),tg>)))
. 31. d3 || ds & d4 || da
. |   & i2 || init
. |   & p2 || pre
. |   & e2 || ef
. |   & s3 || send
. |   & f2 || frame
. |   & s4 || sframe
. 32. ∀kx:(Knd×Id).
. |   (kx ∈ dom(e2)
. |    ⇒
. |    ¬2of(kx) ∈ dom(f2)
. |    ⇒
. |    2of(kx) ∈ dom(frame)
. |    ⇒
. |    deq-member(KindDeq;1of(kx);frame(2of(kx))))
. |   & (kx ∈ dom(ef)
. |      ⇒
. |      ¬2of(kx) ∈ dom(frame)
. |      ⇒
. |      2of(kx) ∈ dom(f2) ⇒ deq-member(KindDeq;1of(kx);f2(2of(kx))))
. 33. ∀kl:(Knd×IdLnk), tg:Id.
. |   (kl ∈ dom(s3)
. |    ⇒
. |    (tg ∈ map(λp.1of(p);s3(kl)))
. |    ⇒
. |    ¬<2of(kl),tg> ∈ dom(s4)
. |    ⇒
. |    <2of(kl),tg> ∈ dom(sframe)
. |    ⇒
. |    deq-member(KindDeq;1of(kl);sframe(<2of(kl),tg>)))
. |   & (kl ∈ dom(send)
. |      ⇒
. |      (tg ∈ map(λp.1of(p);send(kl)))
. |      ⇒
. |      ¬<2of(kl),tg> ∈ dom(sframe)
. |      ⇒
. |      <2of(kl),tg> ∈ dom(s4)
. |      ⇒
. |      deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>)))
. 34. d3 || d1 & d4 || d2
. |   & i2 || i1
. |   & p2 || p1
```

```
. |   & e2 || e1
. |   & s3 || s1
. |   & f2 || f1
. |   & s4 || s2
. 35. ∀kx:(Knd×Id).
. |   (kx ∈ dom(e2)
. |    ⇒
. |    ¬2of(kx) ∈ dom(f2)
. |    ⇒
. |    2of(kx) ∈ dom(f1) ⇒ deq-member(KindDeq;1of(kx);f1(2of(kx))))
. |   & (kx ∈ dom(e1)
. |      ⇒
. |      ¬2of(kx) ∈ dom(f1)
. |      ⇒
. |      2of(kx) ∈ dom(f2) ⇒ deq-member(KindDeq;1of(kx);f2(2of(kx))))
. 36. ∀kl:(Knd×IdLnk), tg:Id.
. |   (kl ∈ dom(s3)
. |    ⇒
. |    (tg ∈ map(λp.1of(p);s3(kl)))
. |    ⇒
. |    ¬<2of(kl),tg> ∈ dom(s4)
. |    ⇒
. |    <2of(kl),tg> ∈ dom(s2)
. |    ⇒
. |    deq-member(KindDeq;1of(kl);s2(<2of(kl),tg>)))
. |   & (kl ∈ dom(s1)
. |      ⇒
. |      (tg ∈ map(λp.1of(p);s1(kl)))
. |      ⇒
. |      ¬<2of(kl),tg> ∈ dom(s2)
. |      ⇒
. |      <2of(kl),tg> ∈ dom(s4)
. |      ⇒
. |      deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>)))
. ⊢  ∀kl:(Knd×IdLnk), tg:Id.
. |  (kl ∈ dom(s3)
. |   ⇒
. |   (tg ∈ map(λp.1of(p);s3(kl)))
. |   ⇒
. |   ¬<2of(kl),tg> ∈ dom(s4)
. |   ⇒
. |   <2of(kl),tg> ∈ dom(sframe ⊕ s2)
. |   ⇒
. |   deq-member(KindDeq;1of(kl);sframe ⊕ s2(<2of(kl),tg>)))
. |  & (kl ∈ dom(send ⊕ s1)
. |     ⇒
. |     (tg ∈ map(λp.1of(p);send ⊕ s1(kl)))
. |     ⇒
. |     ¬<2of(kl),tg> ∈ dom(sframe ⊕ s2)
. |     ⇒
. |     <2of(kl),tg> ∈ dom(s4)
. |     ⇒
. |     deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>)))
. | by Repeat (Analyze 0 THENA Complete Auto)
. |    THEN
```

```
. |    Try
. |    (RWO
. |     Thm* ∀eq:EqDecider(A), f:a:A fp-> Top, g:Top, x:A.
. |          f ⊕ g(x) ∼ if x ∈ dom(f)→ f(x) else g(x) fi
. |     {fpf_DASH_join_DASH_ap_DASH_sq}
. |     0
. |     THEN
. |     RWO
. |     Thm* ∀B:(A→Type), eq:EqDecider(A), f,g:a:A fp-> B(a), x:A.
. |          x ∈ dom(f ⊕ g) ⟺ x ∈ dom(f) ∨ x ∈ dom(g)
. |     {fpf_DASH_join_DASH_dom}
. |     -1
. |     THEN
. |     SplitOnConclITE
. |     THEN
. |     BackThruSomeHyp
. |     THEN
. |     SplitOrHyps)
. |
. 1. ds : x:Id fp-> Type
. 2. da : a:Knd fp-> Type
. 3. init : x:Id fp-> ds(x)?Void
. 4. pre : a:Id fp-> State(ds)→da(locl(a))?Top→Prop
. 5. ef : kx:Knd×Id fp-> State(ds)→da(1of(kx))?Top→ds(2of(kx))?Void
. 6. send :
. |  kl:Knd×IdLnk fp-> (tg:Id
. |                      ×State(ds)→da(1of(kl))?Top→
. |                      (da(rcv(2of(kl); tg))?Void List)) List
. 7. frame : x:Id fp-> Knd List
. 8. sframe : ltg:IdLnk×Id fp-> Knd List
. 9. Top
. 10. d1 : x:Id fp-> Type
. 11. d2 : a:Knd fp-> Type
. 12. i1 : x:Id fp-> d1(x)?Void
. 13. p1 : a:Id fp-> State(d1)→d2(locl(a))?Top→Prop
. 14. e1 : kx:Knd×Id fp-> State(d1)→d2(1of(kx))?Top→d1(2of(kx))?Void
. 15. s1 :
. |   kl:Knd×IdLnk fp-> (tg:Id
. |                       ×State(d1)→d2(1of(kl))?Top→
. |                       (d2(rcv(2of(kl); tg))?Void List)) List
. 16. f1 : x:Id fp-> Knd List
. 17. s2 : ltg:IdLnk×Id fp-> Knd List
. 18. Top
. 19. d3 : x:Id fp-> Type
. 20. d4 : a:Knd fp-> Type
. 21. i2 : x:Id fp-> d3(x)?Void
. 22. p2 : a:Id fp-> State(d3)→d4(locl(a))?Top→Prop
. 23. e2 : kx:Knd×Id fp-> State(d3)→d4(1of(kx))?Top→d3(2of(kx))?Void
. 24. s3 :
. |   kl:Knd×IdLnk fp-> (tg:Id
. |                       ×State(d3)→d4(1of(kl))?Top→
. |                       (d4(rcv(2of(kl); tg))?Void List)) List
. 25. f2 : x:Id fp-> Knd List
. 26. s4 : ltg:IdLnk×Id fp-> Knd List
. 27. Top
```

```
. 28. ds || d1 & da || d2
. |    & init || i1
. |    & pre || p1
. |    & ef || e1
. |    & send || s1
. |    & frame || f1
. |    & sframe || s2
. 29. ∀kx:(Knd×Id).
. |    (kx ∈ dom(ef)
. |     ⇒
. |     ¬2of(kx) ∈ dom(frame)
. |     ⇒
. |     2of(kx) ∈ dom(f1) ⇒ deq-member(KindDeq;1of(kx);f1(2of(kx))))
. |    & (kx ∈ dom(e1)
. |       ⇒
. |       ¬2of(kx) ∈ dom(f1)
. |       ⇒
. |       2of(kx) ∈ dom(frame)
. |       ⇒
. |       deq-member(KindDeq;1of(kx);frame(2of(kx))))
. 30. ∀kl:(Knd×IdLnk), tg:Id.
. |    (kl ∈ dom(send)
. |     ⇒
. |     (tg ∈ map(λp.1of(p);send(kl)))
. |     ⇒
. |     ¬<2of(kl),tg> ∈ dom(sframe)
. |     ⇒
. |     <2of(kl),tg> ∈ dom(s2)
. |     ⇒
. |     deq-member(KindDeq;1of(kl);s2(<2of(kl),tg>)))
. |    & (kl ∈ dom(s1)
. |       ⇒
. |       (tg ∈ map(λp.1of(p);s1(kl)))
. |       ⇒
. |       ¬<2of(kl),tg> ∈ dom(s2)
. |       ⇒
. |       <2of(kl),tg> ∈ dom(sframe)
. |       ⇒
. |       deq-member(KindDeq;1of(kl);sframe(<2of(kl),tg>)))
. 31. d3 || ds & d4 || da
. |    & i2 || init
. |    & p2 || pre
. |    & e2 || ef
. |    & s3 || send
. |    & f2 || frame
. |    & s4 || sframe
. 32. ∀kx:(Knd×Id).
. |    (kx ∈ dom(e2)
. |     ⇒
. |     ¬2of(kx) ∈ dom(f2)
. |     ⇒
. |     2of(kx) ∈ dom(frame)
. |     ⇒
. |     deq-member(KindDeq;1of(kx);frame(2of(kx))))
. |    & (kx ∈ dom(ef)
```

```
. |      ⇒
. |      ¬2of(kx) ∈ dom(frame)
. |      ⇒
. |      2of(kx) ∈ dom(f2) ⇒ deq-member(KindDeq;1of(kx);f2(2of(kx)))
. 33. ∀kl:(Knd×IdLnk), tg:Id.
. |   (kl ∈ dom(s3)
. |    ⇒
. |    (tg ∈ map(λp.1of(p);s3(kl)))
. |    ⇒
. |    ¬<2of(kl),tg> ∈ dom(s4)
. |    ⇒
. |    <2of(kl),tg> ∈ dom(sframe)
. |    ⇒
. |    deq-member(KindDeq;1of(kl);sframe(<2of(kl),tg>))
. |   & (kl ∈ dom(send)
. |      ⇒
. |      (tg ∈ map(λp.1of(p);send(kl)))
. |      ⇒
. |      ¬<2of(kl),tg> ∈ dom(sframe)
. |      ⇒
. |      <2of(kl),tg> ∈ dom(s4)
. |      ⇒
. |      deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>))
. 34. d3 || d1 & d4 || d2
. |   & i2 || i1
. |   & p2 || p1
. |   & e2 || e1
. |   & s3 || s1
. |   & f2 || f1
. |   & s4 || s2
. 35. ∀kx:(Knd×Id).
. |   (kx ∈ dom(e2)
. |    ⇒
. |    ¬2of(kx) ∈ dom(f2)
. |    ⇒
. |    2of(kx) ∈ dom(f1) ⇒ deq-member(KindDeq;1of(kx);f1(2of(kx)))
. |   & (kx ∈ dom(e1)
. |      ⇒
. |      ¬2of(kx) ∈ dom(f1)
. |      ⇒
. |      2of(kx) ∈ dom(f2) ⇒ deq-member(KindDeq;1of(kx);f2(2of(kx)))
. 36. ∀kl:(Knd×IdLnk), tg:Id.
. |   (kl ∈ dom(s3)
. |    ⇒
. |    (tg ∈ map(λp.1of(p);s3(kl)))
. |    ⇒
. |    ¬<2of(kl),tg> ∈ dom(s4)
. |    ⇒
. |    <2of(kl),tg> ∈ dom(s2)
. |    ⇒
. |    deq-member(KindDeq;1of(kl);s2(<2of(kl),tg>))
. |   & (kl ∈ dom(s1)
. |      ⇒
. |      (tg ∈ map(λp.1of(p);s1(kl)))
. |      ⇒
```

```
. |      ¬<2of(kl),tg> ∈ dom(s2)
. |      ⇒
. |      <2of(kl),tg> ∈ dom(s4)
. |      ⇒
. |      deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>)))
. 37. kl : Knd×IdLnk
. 38. tg : Id
. 39. kl ∈ dom(send ⊕ s1)
. ⊢  (tg ∈ map(λp.1of(p);send ⊕ s1(kl)))
. |  ⇒
. |  ¬<2of(kl),tg> ∈ dom(sframe ⊕ s2)
. |  ⇒
. |  <2of(kl),tg> ∈ dom(s4)
. |  ⇒
. |  deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>))  by Analyze 0
. |
. |\
. | 1. ds : x:Id fp-> Type
. | 2. da : a:Knd fp-> Type
. | 3. init : x:Id fp-> ds(x)?Void
. | 4. pre : a:Id fp-> State(ds)→da(locl(a))?Top→Prop
. | 5. ef :
. |    kx:Knd×Id fp-> State(ds)→da(1of(kx))?Top→ds(2of(kx))?Void
. | 6. send :
. |    kl:Knd×IdLnk fp-> (tg:Id
. |                       ×State(ds)→da(1of(kl))?Top→
. |                       (da(rcv(2of(kl); tg))?Void List)) List
. | 7. frame : x:Id fp-> Knd List
. | 8. sframe : ltg:IdLnk×Id fp-> Knd List
. | 9. Top
. | 10. d1 : x:Id fp-> Type
. | 11. d2 : a:Knd fp-> Type
. | 12. i1 : x:Id fp-> d1(x)?Void
. | 13. p1 : a:Id fp-> State(d1)→d2(locl(a))?Top→Prop
. | 14. e1 :
. |     kx:Knd×Id fp-> State(d1)→d2(1of(kx))?Top→d1(2of(kx))?Void
. | 15. s1 :
. |     kl:Knd×IdLnk fp-> (tg:Id
. |                        ×State(d1)→d2(1of(kl))?Top→
. |                        (d2(rcv(2of(kl); tg))?Void List)) List
. | 16. f1 : x:Id fp-> Knd List
. | 17. s2 : ltg:IdLnk×Id fp-> Knd List
. | 18. Top
. | 19. d3 : x:Id fp-> Type
. | 20. d4 : a:Knd fp-> Type
. | 21. i2 : x:Id fp-> d3(x)?Void
. | 22. p2 : a:Id fp-> State(d3)→d4(locl(a))?Top→Prop
. | 23. e2 :
. |     kx:Knd×Id fp-> State(d3)→d4(1of(kx))?Top→d3(2of(kx))?Void
. | 24. s3 :
. |     kl:Knd×IdLnk fp-> (tg:Id
. |                        ×State(d3)→d4(1of(kl))?Top→
. |                        (d4(rcv(2of(kl); tg))?Void List)) List
. | 25. f2 : x:Id fp-> Knd List
. | 26. s4 : ltg:IdLnk×Id fp-> Knd List
```

```
. | 27. Top
. | 28. ds || d1 & da || d2
. |     & init || i1
. |     & pre || p1
. |     & ef || e1
. |     & send || s1
. |     & frame || f1
. |     & sframe || s2
. | 29. ∀kx:(Knd×Id).
. |     (kx ∈ dom(ef)
. |      ⇒
. |      ¬2of(kx) ∈ dom(frame)
. |      ⇒
. |      2of(kx) ∈ dom(f1) ⇒ deq-member(KindDeq;1of(kx);f1(2of(kx))))
. |     & (kx ∈ dom(e1)
. |        ⇒
. |        ¬2of(kx) ∈ dom(f1)
. |        ⇒
. |        2of(kx) ∈ dom(frame)
. |        ⇒
. |        deq-member(KindDeq;1of(kx);frame(2of(kx))))
. | 30. ∀kl:(Knd×IdLnk), tg:Id.
. |     (kl ∈ dom(send)
. |      ⇒
. |      (tg ∈ map(λp.1of(p);send(kl)))
. |      ⇒
. |      ¬<2of(kl),tg> ∈ dom(sframe)
. |      ⇒
. |      <2of(kl),tg> ∈ dom(s2)
. |      ⇒
. |      deq-member(KindDeq;1of(kl);s2(<2of(kl),tg>)))
. |     & (kl ∈ dom(s1)
. |        ⇒
. |        (tg ∈ map(λp.1of(p);s1(kl)))
. |        ⇒
. |        ¬<2of(kl),tg> ∈ dom(s2)
. |        ⇒
. |        <2of(kl),tg> ∈ dom(sframe)
. |        ⇒
. |        deq-member(KindDeq;1of(kl);sframe(<2of(kl),tg>)))
. | 31. d3 || ds & d4 || da
. |     & i2 || init
. |     & p2 || pre
. |     & e2 || ef
. |     & s3 || send
. |     & f2 || frame
. |     & s4 || sframe
. | 32. ∀kx:(Knd×Id).
. |     (kx ∈ dom(e2)
. |      ⇒
. |      ¬2of(kx) ∈ dom(f2)
. |      ⇒
. |      2of(kx) ∈ dom(frame)
. |      ⇒
. |      deq-member(KindDeq;1of(kx);frame(2of(kx))))
```

```
. |     & (kx ∈ dom(ef)
. |        ⇒
. |        ¬2of(kx) ∈ dom(frame)
. |        ⇒
. |        2of(kx) ∈ dom(f2)
. |        ⇒
. |        deq-member(KindDeq;1of(kx);f2(2of(kx))))
. | 33. ∀kl:(Knd×IdLnk), tg:Id.
. |     (kl ∈ dom(s3)
. |      ⇒
. |      (tg ∈ map(λp.1of(p);s3(kl)))
. |      ⇒
. |      ¬<2of(kl),tg> ∈ dom(s4)
. |      ⇒
. |      <2of(kl),tg> ∈ dom(sframe)
. |      ⇒
. |      deq-member(KindDeq;1of(kl);sframe(<2of(kl),tg>)))
. |     & (kl ∈ dom(send)
. |        ⇒
. |        (tg ∈ map(λp.1of(p);send(kl)))
. |        ⇒
. |        ¬<2of(kl),tg> ∈ dom(sframe)
. |        ⇒
. |        <2of(kl),tg> ∈ dom(s4)
. |        ⇒
. |        deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>)))
. | 34. d3 || d1 & d4 || d2
. |     & i2 || i1
. |     & p2 || p1
. |     & e2 || e1
. |     & s3 || s1
. |     & f2 || f1
. |     & s4 || s2
. | 35. ∀kx:(Knd×Id).
. |     (kx ∈ dom(e2)
. |      ⇒
. |      ¬2of(kx) ∈ dom(f2)
. |      ⇒
. |      2of(kx) ∈ dom(f1) ⇒ deq-member(KindDeq;1of(kx);f1(2of(kx))))
. |     & (kx ∈ dom(e1)
. |        ⇒
. |        ¬2of(kx) ∈ dom(f1)
. |        ⇒
. |        2of(kx) ∈ dom(f2)
. |        ⇒
. |        deq-member(KindDeq;1of(kx);f2(2of(kx))))
. | 36. ∀kl:(Knd×IdLnk), tg:Id.
. |     (kl ∈ dom(s3)
. |      ⇒
. |      (tg ∈ map(λp.1of(p);s3(kl)))
. |      ⇒
. |      ¬<2of(kl),tg> ∈ dom(s4)
. |      ⇒
. |      <2of(kl),tg> ∈ dom(s2)
. |      ⇒
```

```
. |      deq-member(KindDeq;1of(kl);s2(<2of(kl),tg>)))
. |     & (kl ∈ dom(s1)
. |        ⇒
. |        (tg ∈ map(λp.1of(p);s1(kl)))
. |        ⇒
. |        ¬<2of(kl),tg> ∈ dom(s2)
. |        ⇒
. |        <2of(kl),tg> ∈ dom(s4)
. |        ⇒
. |        deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>)))
. | 37. kl : Knd×IdLnk
. | 38. tg : Id
. | 39. kl ∈ dom(send ⊕ s1)
. | 40. (tg ∈ map(λp.1of(p);send ⊕ s1(kl)))
. | ⊢  ¬<2of(kl),tg> ∈ dom(sframe ⊕ s2)
. |    ⇒
. |    <2of(kl),tg> ∈ dom(s4)
. |    ⇒
. |    deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>))
. |   by Auto THEN Decide kl ∈ dom(send)
. |      THENL
. |      [Id
. |      ;RWO
. |       Thm* ∀eq:EqDecider(A), f,g:a:A fp-> Top, x:A.
. |            x ∈ dom(f ⊕ g) ⟺ x ∈ dom(f) ∨ x ∈ dom(g)
. |       {fpf_DASH_join_DASH_dom2}
. |       -5
. |       THEN
. |       Analyze -5
. |       THEN
. |       Try Trivial]
. |      THEN
. |      AllHyps
. |      (λh.
. |       BackThru h
. |       THENL
. |       [NthHypSq -4 THEN RepeatFor 2 (Analyze THEN Try Trivial)
. |        THEN
. |        RWO
. |        Thm* ∀eq:EqDecider(A), f:a:A fp-> Top, g:Top, x:A.
. |             f ⊕ g(x) ∼ if x ∈ dom(f)→ f(x) else g(x) fi
. |        {fpf_DASH_join_DASH_ap_DASH_sq}
. |        0
. |        THEN
. |        SplitOnConclITE
. |       ;ParallelOp -3
. |        THEN
. |        RWO
. |        Thm* ∀eq:EqDecider(A), f,g:a:A fp-> Top, x:A.
. |             x ∈ dom(f ⊕ g) ⟺ x ∈ dom(f) ∨ x ∈ dom(g)
. |        {fpf_DASH_join_DASH_dom2}
. |        0
. |        THEN
. |        ((OrRight THEN Complete Auto)
. |         ORELSE
```

```
. |         (OrLeft THEN Complete Auto))])
. |
. .\
. . .....wf .....
. . 1. ds : x:Id fp-> Type
. . 2. da : a:Knd fp-> Type
. . 3. init : x:Id fp-> ds(x)?Void
. . 4. pre : a:Id fp-> State(ds)→da(locl(a))?Top→Prop
. . 5. ef :
. . |  kx:Knd×Id fp-> State(ds)→da(1of(kx))?Top→ds(2of(kx))?Void
. . 6. send :
. . |  kl:Knd×IdLnk fp-> (tg:Id
. . |                     ×State(ds)→da(1of(kl))?Top→
. . |                     (da(rcv(2of(kl); tg))?Void List)) List
. . 7. frame : x:Id fp-> Knd List
. . 8. sframe : ltg:IdLnk×Id fp-> Knd List
. . 9. Top
. . 10. d1 : x:Id fp-> Type
. . 11. d2 : a:Knd fp-> Type
. . 12. i1 : x:Id fp-> d1(x)?Void
. . 13. p1 : a:Id fp-> State(d1)→d2(locl(a))?Top→Prop
. . 14. e1 :
. . |   kx:Knd×Id fp-> State(d1)→d2(1of(kx))?Top→d1(2of(kx))?Void
. . 15. s1 :
. . |   kl:Knd×IdLnk fp-> (tg:Id
. . |                      ×State(d1)→d2(1of(kl))?Top→
. . |                      (d2(rcv(2of(kl); tg))?Void List)) List
. . 16. f1 : x:Id fp-> Knd List
. . 17. s2 : ltg:IdLnk×Id fp-> Knd List
. . 18. Top
. . 19. d3 : x:Id fp-> Type
. . 20. d4 : a:Knd fp-> Type
. . 21. i2 : x:Id fp-> d3(x)?Void
. . 22. p2 : a:Id fp-> State(d3)→d4(locl(a))?Top→Prop
. . 23. e2 :
. . |   kx:Knd×Id fp-> State(d3)→d4(1of(kx))?Top→d3(2of(kx))?Void
. . 24. s3 :
. . |   kl:Knd×IdLnk fp-> (tg:Id
. . |                      ×State(d3)→d4(1of(kl))?Top→
. . |                      (d4(rcv(2of(kl); tg))?Void List)) List
. . 25. f2 : x:Id fp-> Knd List
. . 26. s4 : ltg:IdLnk×Id fp-> Knd List
. . 27. Top
. . 28. ds || d1 & da || d2
. . |   & init || i1
. . |   & pre || p1
. . |   & ef || e1
. . |   & send || s1
. . |   & frame || f1
. . |   & sframe || s2
. . 29. ∀kx:(Knd×Id).
. . |   (kx ∈ dom(ef)
. . |    ⇒
. . |    ¬2of(kx) ∈ dom(frame)
. . |    ⇒
```

```
. . |    2of(kx) ∈ dom(f1) ⇒ deq-member(KindDeq;1of(kx);f1(2of(kx))))
. . |   & (kx ∈ dom(e1)
. . |      ⇒
. . |      ¬2of(kx) ∈ dom(f1)
. . |      ⇒
. . |      2of(kx) ∈ dom(frame)
. . |      ⇒
. . |      deq-member(KindDeq;1of(kx);frame(2of(kx))))
. . 30. ∀kl:(Knd×IdLnk), tg:Id.
. . |   (kl ∈ dom(send)
. . |    ⇒
. . |    (tg ∈ map(λp.1of(p);send(kl)))
. . |    ⇒
. . |    ¬<2of(kl),tg> ∈ dom(sframe)
. . |    ⇒
. . |    <2of(kl),tg> ∈ dom(s2)
. . |    ⇒
. . |    deq-member(KindDeq;1of(kl);s2(<2of(kl),tg>)))
. . |   & (kl ∈ dom(s1)
. . |      ⇒
. . |      (tg ∈ map(λp.1of(p);s1(kl)))
. . |      ⇒
. . |      ¬<2of(kl),tg> ∈ dom(s2)
. . |      ⇒
. . |      <2of(kl),tg> ∈ dom(sframe)
. . |      ⇒
. . |      deq-member(KindDeq;1of(kl);sframe(<2of(kl),tg>)))
. . 31. d3 || ds & d4 || da
. . |   & i2 || init
. . |   & p2 || pre
. . |   & e2 || ef
. . |   & s3 || send
. . |   & f2 || frame
. . |   & s4 || sframe
. . 32. ∀kx:(Knd×Id).
. . |   (kx ∈ dom(e2)
. . |    ⇒
. . |    ¬2of(kx) ∈ dom(f2)
. . |    ⇒
. . |    2of(kx) ∈ dom(frame)
. . |    ⇒
. . |    deq-member(KindDeq;1of(kx);frame(2of(kx))))
. . |   & (kx ∈ dom(ef)
. . |      ⇒
. . |      ¬2of(kx) ∈ dom(frame)
. . |      ⇒
. . |      2of(kx) ∈ dom(f2)
. . |      ⇒
. . |      deq-member(KindDeq;1of(kx);f2(2of(kx))))
. . 33. ∀kl:(Knd×IdLnk), tg:Id.
. . |   (kl ∈ dom(s3)
. . |    ⇒
. . |    (tg ∈ map(λp.1of(p);s3(kl)))
. . |    ⇒
. . |    ¬<2of(kl),tg> ∈ dom(s4)
```

```
. . |    ⇒
. . |    <2of(kl),tg> ∈ dom(sframe)
. . |    ⇒
. . |    deq-member(KindDeq;1of(kl);sframe(<2of(kl),tg>))
. . |   & (kl ∈ dom(send)
. . |      ⇒
. . |      (tg ∈ map(λp.1of(p);send(kl)))
. . |      ⇒
. . |      ¬<2of(kl),tg> ∈ dom(sframe)
. . |      ⇒
. . |      <2of(kl),tg> ∈ dom(s4)
. . |      ⇒
. . |      deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>))
. . 34. d3 || d1 & d4 || d2
. . |   & i2 || i1
. . |   & p2 || p1
. . |   & e2 || e1
. . |   & s3 || s1
. . |   & f2 || f1
. . |   & s4 || s2
. . 35. ∀kx:(Knd×Id).
. . |   (kx ∈ dom(e2)
. . |    ⇒
. . |    ¬2of(kx) ∈ dom(f2)
. . |    ⇒
. . |    2of(kx) ∈ dom(f1) ⇒ deq-member(KindDeq;1of(kx);f1(2of(kx))))
. . |   & (kx ∈ dom(e1)
. . |      ⇒
. . |      ¬2of(kx) ∈ dom(f1)
. . |      ⇒
. . |      2of(kx) ∈ dom(f2)
. . |      ⇒
. . |      deq-member(KindDeq;1of(kx);f2(2of(kx))))
. . 36. ∀kl:(Knd×IdLnk), tg:Id.
. . |   (kl ∈ dom(s3)
. . |    ⇒
. . |    (tg ∈ map(λp.1of(p);s3(kl)))
. . |    ⇒
. . |    ¬<2of(kl),tg> ∈ dom(s4)
. . |    ⇒
. . |    <2of(kl),tg> ∈ dom(s2)
. . |    ⇒
. . |    deq-member(KindDeq;1of(kl);s2(<2of(kl),tg>))
. . |   & (kl ∈ dom(s1)
. . |      ⇒
. . |      (tg ∈ map(λp.1of(p);s1(kl)))
. . |      ⇒
. . |      ¬<2of(kl),tg> ∈ dom(s2)
. . |      ⇒
. . |      <2of(kl),tg> ∈ dom(s4)
. . |      ⇒
. . |      deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>))
. . 37. kl : Knd×IdLnk
. . 38. tg : Id
. . 39. kl ∈ dom(send ⊕ s1)
```

```
. . ⊢  (tg ∈ map(λp.1of(p);send ⊕ s1(kl))) ∈ Prop
. . | by GenConcl (send ⊕ s1(kl) = L) THEN Try (Complete Auto)
. . |
. . .....wf .....
. . 1. ds : x:Id fp-> Type
. . 2. da : a:Knd fp-> Type
. . 3. init : x:Id fp-> ds(x)?Void
. . 4. pre : a:Id fp-> State(ds)→da(locl(a))?Top→Prop
. . 5. ef :
. .    kx:Knd×Id fp-> State(ds)→da(1of(kx))?Top→ds(2of(kx))?Void
. . 6. send :
. .    kl:Knd×IdLnk fp-> (tg:Id
. .                       ×State(ds)→da(1of(kl))?Top→
. .                       (da(rcv(2of(kl); tg))?Void List)) List
. . 7. frame : x:Id fp-> Knd List
. . 8. sframe : ltg:IdLnk×Id fp-> Knd List
. . 9. Top
. . 10. d1 : x:Id fp-> Type
. . 11. d2 : a:Knd fp-> Type
. . 12. i1 : x:Id fp-> d1(x)?Void
. . 13. p1 : a:Id fp-> State(d1)→d2(locl(a))?Top→Prop
. . 14. e1 :
. .     kx:Knd×Id fp-> State(d1)→d2(1of(kx))?Top→d1(2of(kx))?Void
. . 15. s1 :
. .     kl:Knd×IdLnk fp-> (tg:Id
. .                        ×State(d1)→d2(1of(kl))?Top→
. .                        (d2(rcv(2of(kl); tg))?Void List)) List
. . 16. f1 : x:Id fp-> Knd List
. . 17. s2 : ltg:IdLnk×Id fp-> Knd List
. . 18. Top
. . 19. d3 : x:Id fp-> Type
. . 20. d4 : a:Knd fp-> Type
. . 21. i2 : x:Id fp-> d3(x)?Void
. . 22. p2 : a:Id fp-> State(d3)→d4(locl(a))?Top→Prop
. . 23. e2 :
. .     kx:Knd×Id fp-> State(d3)→d4(1of(kx))?Top→d3(2of(kx))?Void
. . 24. s3 :
. .     kl:Knd×IdLnk fp-> (tg:Id
. .                        ×State(d3)→d4(1of(kl))?Top→
. .                        (d4(rcv(2of(kl); tg))?Void List)) List
. . 25. f2 : x:Id fp-> Knd List
. . 26. s4 : ltg:IdLnk×Id fp-> Knd List
. . 27. Top
. . 28. ds || d1 & da || d2
. .     & init || i1
. .     & pre || p1
. .     & ef || e1
. .     & send || s1
. .     & frame || f1
. .     & sframe || s2
. . 29. ∀kx:(Knd×Id).
. .     (kx ∈ dom(ef)
. .      ⇒
. .      ¬2of(kx) ∈ dom(frame)
. .      ⇒
```

```
. .      2of(kx) ∈ dom(f1) ⇒ deq-member(KindDeq;1of(kx);f1(2of(kx))))
. .     & (kx ∈ dom(e1)
. .        ⇒
. .        ¬2of(kx) ∈ dom(f1)
. .        ⇒
. .        2of(kx) ∈ dom(frame)
. .        ⇒
. .        deq-member(KindDeq;1of(kx);frame(2of(kx))))
. . 30. ∀kl:(Knd×IdLnk), tg:Id.
. .     (kl ∈ dom(send)
. .      ⇒
. .      (tg ∈ map(λp.1of(p);send(kl)))
. .      ⇒
. .      ¬<2of(kl),tg> ∈ dom(sframe)
. .      ⇒
. .      <2of(kl),tg> ∈ dom(s2)
. .      ⇒
. .      deq-member(KindDeq;1of(kl);s2(<2of(kl),tg>)))
. .     & (kl ∈ dom(s1)
. .        ⇒
. .        (tg ∈ map(λp.1of(p);s1(kl)))
. .        ⇒
. .        ¬<2of(kl),tg> ∈ dom(s2)
. .        ⇒
. .        <2of(kl),tg> ∈ dom(sframe)
. .        ⇒
. .        deq-member(KindDeq;1of(kl);sframe(<2of(kl),tg>)))
. . 31. d3 || ds & d4 || da
. .     & i2 || init
. .     & p2 || pre
. .     & e2 || ef
. .     & s3 || send
. .     & f2 || frame
. .     & s4 || sframe
. . 32. ∀kx:(Knd×Id).
. .     (kx ∈ dom(e2)
. .      ⇒
. .      ¬2of(kx) ∈ dom(f2)
. .      ⇒
. .      2of(kx) ∈ dom(frame)
. .      ⇒
. .      deq-member(KindDeq;1of(kx);frame(2of(kx))))
. .     & (kx ∈ dom(ef)
. .        ⇒
. .        ¬2of(kx) ∈ dom(frame)
. .        ⇒
. .        2of(kx) ∈ dom(f2)
. .        ⇒
. .        deq-member(KindDeq;1of(kx);f2(2of(kx))))
. . 33. ∀kl:(Knd×IdLnk), tg:Id.
. .     (kl ∈ dom(s3)
. .      ⇒
. .      (tg ∈ map(λp.1of(p);s3(kl)))
. .      ⇒
. .      ¬<2of(kl),tg> ∈ dom(s4)
```

```
. .      ⇒
. .      <2of(kl),tg> ∈ dom(sframe)
. .      ⇒
. .      deq-member(KindDeq;1of(kl);sframe(<2of(kl),tg>))
. .     & (kl ∈ dom(send)
. .        ⇒
. .        (tg ∈ map(λp.1of(p);send(kl)))
. .        ⇒
. .        ¬<2of(kl),tg> ∈ dom(sframe)
. .        ⇒
. .        <2of(kl),tg> ∈ dom(s4)
. .        ⇒
. .        deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>)))
. . 34. d3 || d1 & d4 || d2
. .     & i2 || i1
. .     & p2 || p1
. .     & e2 || e1
. .     & s3 || s1
. .     & f2 || f1
. .     & s4 || s2
. . 35. ∀kx:(Knd×Id).
. .     (kx ∈ dom(e2)
. .      ⇒
. .      ¬2of(kx) ∈ dom(f2)
. .      ⇒
. .      2of(kx) ∈ dom(f1) ⇒ deq-member(KindDeq;1of(kx);f1(2of(kx))))
. .     & (kx ∈ dom(e1)
. .        ⇒
. .        ¬2of(kx) ∈ dom(f1)
. .        ⇒
. .        2of(kx) ∈ dom(f2)
. .        ⇒
. .        deq-member(KindDeq;1of(kx);f2(2of(kx))))
. . 36. ∀kl:(Knd×IdLnk), tg:Id.
. .     (kl ∈ dom(s3)
. .      ⇒
. .      (tg ∈ map(λp.1of(p);s3(kl)))
. .      ⇒
. .      ¬<2of(kl),tg> ∈ dom(s4)
. .      ⇒
. .      <2of(kl),tg> ∈ dom(s2)
. .      ⇒
. .      deq-member(KindDeq;1of(kl);s2(<2of(kl),tg>)))
. .     & (kl ∈ dom(s1)
. .        ⇒
. .        (tg ∈ map(λp.1of(p);s1(kl)))
. .        ⇒
. .        ¬<2of(kl),tg> ∈ dom(s2)
. .        ⇒
. .        <2of(kl),tg> ∈ dom(s4)
. .        ⇒
. .        deq-member(KindDeq;1of(kl);s4(<2of(kl),tg>)))
. . 37. kl : Knd×IdLnk
. . 38. Id
. . 39. kl ∈ dom(send ⊕ s1)
```

```
. . ⊢  send ⊕ s1(kl) ∈ (Id×Top) List
. .   by AssertBY (s1 ∈ kl:Knd×IdLnk fp-> (Id×Top) List) FpfSubtype
. .      THEN
. .      FpfSubtype
END
```