```
possible-world-monotonic  THM

⊢  ∀A,B:Dsys.
|  A ⊆ B ⇒ (∀w:World. PossibleWorld(B;w) ⇒ PossibleWorld(A;w))
| by Auto THEN RepeatFor 2 (ParallelOp -1) THEN ExRepD THEN Analyze 0
|    THENL
|    [RepeatFor 5 (Thin -1)
|    ;Analyze 0
|     THENL
|     [RepeatFor 5 (Thin -1)
|     ;Analyze 0
|      THENL
|      [RepeatFor 5 (Thin -1)
|      ;Analyze 0 THENL [RepeatFor 5 (Thin -1);Id]]]]
|
|\
| 1. A : Dsys
| 2. B : Dsys
| 3. A ⊆ B
| 4. w : World
| 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
| ⊢  ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
|   by Auto
|      THEN
|      Using [‘B’,M(i).ds(x)]
|      (BackThru
|       Thm* (A ⊆r B) ⇒ (B ⊆r C) ⇒ (A ⊆r C)
|       {subtype_rel_transitivity})
|      THEN
|      Try BackThruSomeHyp
|      THEN
|      BackThru
|      Thm* ∀M1,M2:MsgA, x:Id. M1 ⊆ M2 ⇒ (M2.ds(x) ⊆r M1.ds(x))
|      {ma_DASH_ds_DASH_sub}
|      THEN
|      AllHyps (λh.Unfold ‘d-sub‘ h THEN BackThru h)
|
|\
| 1. A : Dsys
| 2. B : Dsys
| 3. A ⊆ B
| 4. w : World
| 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
| 6. ∀i:Id, a:Action(i).
|    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
| ⊢  ∀i:Id, a:Action(i).
|    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
|   by Auto
|      THEN
|      Using [‘B’,M(i).da(kind(a))]
|      (BackThru
|       Thm* (A ⊆r B) ⇒ (B ⊆r C) ⇒ (A ⊆r C)
|       {subtype_rel_transitivity})
|      THEN
|      Try BackThruSomeHyp
```

```
|      THEN
|      BackThru
|      Thm* ∀M1,M2:MsgA, k:Knd. M1 ⊆ M2 ⇒ (M2.da(k) ⊆r M1.da(k))
|      {ma_DASH_da_DASH_sub}
|      THEN
|      AllHyps (λh.Unfold `d-sub` h THEN BackThru h)
|
|\
| 1. A : Dsys
| 2. B : Dsys
| 3. A ⊆ B
| 4. w : World
| 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
| 6. ∀i:Id, a:Action(i).
|    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
| 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
| ⊢  ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
|   by Auto
|      THEN
|      Using [`B',M(source(l)).da(rcv(l; tg))]
|      (BackThru
|       Thm* (A ⊆r B) ⇒ (B ⊆r C) ⇒ (A ⊆r C)
|       {subtype_rel_transitivity})
|      THEN
|      Try BackThruSomeHyp
|      THEN
|      BackThru
|      Thm* ∀M1,M2:MsgA, k:Knd. M1 ⊆ M2 ⇒ (M2.da(k) ⊆r M1.da(k))
|      {ma_DASH_da_DASH_sub}
|      THEN
|      AllHyps (λh.Unfold `d-sub` h THEN BackThru h)
|
|\
| 1. A : Dsys
| 2. B : Dsys
| 3. A ⊆ B
| 4. w : World
| 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
| 6. ∀i:Id, a:Action(i).
|    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
| 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
| 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
| ⊢  ∀i,x:Id. M(i).init(x,s(i;0).x)
|   by Auto
|      THEN
|      Using [`M2',M(i);`V',(λ2x. M(i).ds(x))]
|      (BackThru
|       Thm* ∀V:(Id→Type), M1,M2:MsgA.
|            (∀x:Id. V(x) ⊆r M2.ds(x))
|            ⇒
|            M1 ⊆ M2 ⇒ (∀x:Id, v:V(x). M2.init(x,v) ⇒ M1.init(x,v))
|       {ma_DASH_init_DASH_sub})
|      THEN
|      Try BackThruSomeHyp
|      THEN
```

```
|      AllHyps (λh.Unfold `d-sub` h THEN BackThru h)
|      THEN
|      DoSubsume
|      THEN
|      BackThruSomeHyp
|
.\
. 1. A : Dsys
. 2. B : Dsys
. 3. A ⊆ B
. 4. w : World
. 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. 6. ∀i:Id, a:Action(i).
. |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. 9. ∀i:Id, t:ℕ.
. |  ¬isnull(a(i;t))
. |  ⇒
. |  (islocal(kind(a(i;t)))
. |   ⇒
. |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. |  & (∀x:Id.
. |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t+1).x))
. |  & (∀l:IdLnk.
. |     M(i).send(kind(a(i;t));l;λx.
. |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. |  & (∀x:Id.
. |     ¬M(i).frame(kind(a(i;t)) affects x) ⇒ s(i;t).x = s(i;t+1).x)
. |  & (∀l:IdLnk, tg:Id.
. |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. |     ⇒
. |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. 10. ∀i,a:Id, t:ℕ.
. |   ∃t':ℕ.
. |   t≤t'
. |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. |     ∨ ¬a declared in M(i)
. |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. 12. ∀i:Id, a:Action(i).
. |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. 13. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. ⊢  (∀i:Id, t:ℕ.
. |   ¬isnull(a(i;t))
. |   ⇒
. |   (islocal(kind(a(i;t)))
. |    ⇒
. |    M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. |   & (∀x:Id.
. |      M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t+1).x))
. |   & (∀l:IdLnk.
. |      M(i).send(kind(a(i;t));l;λx.
. |      s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
```

```
. |   & (∀x:Id.
. |      ¬M(i).frame(kind(a(i;t)) affects x) ⇒ s(i;t).x = s(i;t+1).x)
. |   & (∀l:IdLnk, tg:Id.
. |      ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. |      ⇒
. |      w-tagged(tg; onlnk(l;m(i;t))) = nil))
. |  & (∀i,a:Id, t:ℕ.
. |     ∃t':ℕ.
. |     t≤t'
. |     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. |       ∨ ¬a declared in M(i)
. |       ∨ unsolvable M(i).pre(a,λx.s(i;t').x))  by Analyze 0
. |
. |\
. | 1. A : Dsys
. | 2. B : Dsys
. | 3. A ⊆ B
. | 4. w : World
. | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | 6. ∀i:Id, a:Action(i).
. | |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | 9. ∀i:Id, t:ℕ.
. | |  ¬isnull(a(i;t))
. | |  ⇒
. | |  (islocal(kind(a(i;t)))
. | |   ⇒
. | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | |  & (∀x:Id.
. | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t+1).x))
. | |  & (∀l:IdLnk.
. | |     M(i).send(kind(a(i;t));l;λx.
. | |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | |  & (∀x:Id.
. | |     ¬M(i).frame(kind(a(i;t)) affects x)
. | |     ⇒
. | |     s(i;t).x = s(i;t+1).x)
. | |  & (∀l:IdLnk, tg:Id.
. | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | |     ⇒
. | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | 10. ∀i,a:Id, t:ℕ.
. | |   ∃t':ℕ.
. | |   t≤t'
. | |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | |     ∨ ¬a declared in M(i)
. | |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | 12. ∀i:Id, a:Action(i).
. | |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | 13. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | ⊢  ∀i:Id, t:ℕ.
. | |  ¬isnull(a(i;t))
```

```
. | |  ⇒
. | |  (islocal(kind(a(i;t)))
. | |   ⇒
. | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | |  & (∀x:Id.
. | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t+1).x))
. | |  & (∀l:IdLnk.
. | |     M(i).send(kind(a(i;t));l;λx.
. | |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | |  & (∀x:Id.
. | |     ¬M(i).frame(kind(a(i;t)) affects x)
. | |     ⇒
. | |     s(i;t).x = s(i;t+1).x)
. | |  & (∀l:IdLnk, tg:Id.
. | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | |     ⇒
. | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | | by RepeatFor 3 (Analyze 0)
. | |    THEN
. | |    AllHyps (λj.InstHyp [i;t] j THENA (Auto THEN Trivial))
. | |
. | 1. A : Dsys
. | 2. B : Dsys
. | 3. A ⊆ B
. | 4. w : World
. | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | 6. ∀i:Id, a:Action(i).
. | |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | 9. ∀i:Id, t:ℕ.
. | |  ¬isnull(a(i;t))
. | |  ⇒
. | |  (islocal(kind(a(i;t)))
. | |   ⇒
. | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | |  & (∀x:Id.
. | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t+1).x))
. | |  & (∀l:IdLnk.
. | |     M(i).send(kind(a(i;t));l;λx.
. | |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | |  & (∀x:Id.
. | |     ¬M(i).frame(kind(a(i;t)) affects x)
. | |     ⇒
. | |     s(i;t).x = s(i;t+1).x)
. | |  & (∀l:IdLnk, tg:Id.
. | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | |     ⇒
. | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | 10. ∀i,a:Id, t:ℕ.
. | |   ∃t':ℕ.
. | |   t≤t'
. | |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | |     ∨ ¬a declared in M(i)
. | |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
```

```
. | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | 12. ∀i:Id, a:Action(i).
. | |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | 13. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | 15. i : Id
. | 16. t : ℕ
. | 17. ¬isnull(a(i;t))
. | 18. (islocal(kind(a(i;t)))
. | |     ⇒
. | |     M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | |    & (∀x:Id.
. | |       M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t+1).x))
. | |    & (∀l:IdLnk.
. | |       M(i).send(kind(a(i;t));l;λx.
. | |       s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | |    & (∀x:Id.
. | |       ¬M(i).frame(kind(a(i;t)) affects x)
. | |       ⇒
. | |       s(i;t).x = s(i;t+1).x)
. | |    & (∀l:IdLnk, tg:Id.
. | |       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | |       ⇒
. | |       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | ⊢  (islocal(kind(a(i;t)))
. | |    ⇒
. | |    M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | |  & (∀x:Id.
. | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t+1).x))
. | |  & (∀l:IdLnk.
. | |     M(i).send(kind(a(i;t));l;λx.
. | |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | |  & (∀x:Id.
. | |     ¬M(i).frame(kind(a(i;t)) affects x)
. | |     ⇒
. | |     s(i;t).x = s(i;t+1).x)
. | |  & (∀l:IdLnk, tg:Id.
. | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | |     ⇒
. | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | | by AssertBY ((λx.s(i;t).x) ∈ M(i).state)
. | |    (Unfold ‘ma-st‘ 0 THEN Unfold ‘ma-state‘ 0
. | |     THEN
. | |     Fold ‘ma-ds‘ 0
. | |     THEN
. | |     DoSubsume
. | |     THEN
. | |     Try PWTrivial
. | |     THEN
. | |     BackThruSomeHyp
. | |     THEN
. | |     Try PWTrivial)
. | |    THEN
. | |    AssertBY ((λx.s(i;t+1).x) ∈ M(i).state)
. | |    (Unfold ‘ma-st‘ 0 THEN Unfold ‘ma-state‘ 0
```

```
. | |     THEN
. | |     Fold ‘ma-ds‘ 0
. | |     THEN
. | |     DoSubsume
. | |     THEN
. | |     Try PWTrivial
. | |     THEN
. | |     BackThruSomeHyp
. | |     THEN
. | |     Try PWTrivial)
. | |
. | 1. A : Dsys
. | 2. B : Dsys
. | 3. A ⊆ B
. | 4. w : World
. | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | 6. ∀i:Id, a:Action(i).
. | |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | 9. ∀i:Id, t:ℕ.
. | |  ¬isnull(a(i;t))
. | |  ⇒
. | |  (islocal(kind(a(i;t)))
. | |   ⇒
. | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | |  & (∀x:Id.
. | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t+1).x))
. | |  & (∀l:IdLnk.
. | |     M(i).send(kind(a(i;t));l;λx.
. | |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | |  & (∀x:Id.
. | |     ¬M(i).frame(kind(a(i;t)) affects x)
. | |     ⇒
. | |     s(i;t).x = s(i;t+1).x)
. | |  & (∀l:IdLnk, tg:Id.
. | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | |     ⇒
. | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | 10. ∀i,a:Id, t:ℕ.
. | |   ∃t':ℕ.
. | |   t≤t'
. | |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | |     ∨ ¬a declared in M(i)
. | |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | 12. ∀i:Id, a:Action(i).
. | |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | 13. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | 15. i : Id
. | 16. t : ℕ
. | 17. ¬isnull(a(i;t))
. | 18. (islocal(kind(a(i;t)))
. | |    ⇒
```

```
. | |    M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | |   & (∀x:Id.
. | |      M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t+1).x))
. | |   & (∀l:IdLnk.
. | |      M(i).send(kind(a(i;t));l;λx.
. | |      s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | |   & (∀x:Id.
. | |      ¬M(i).frame(kind(a(i;t)) affects x)
. | |      ⇒
. | |      s(i;t).x = s(i;t+1).x)
. | |   & (∀l:IdLnk, tg:Id.
. | |      ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | |      ⇒
. | |      w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | 19. (λx.s(i;t).x) ∈ M(i).state
. | 20. (λx.s(i;t+1).x) ∈ M(i).state
. | ⊢  (islocal(kind(a(i;t)))
. | |   ⇒
. | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | |  & (∀x:Id.
. | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t+1).x))
. | |  & (∀l:IdLnk.
. | |     M(i).send(kind(a(i;t));l;λx.
. | |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | |  & (∀x:Id.
. | |     ¬M(i).frame(kind(a(i;t)) affects x)
. | |     ⇒
. | |     s(i;t).x = s(i;t+1).x)
. | |  & (∀l:IdLnk, tg:Id.
. | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | |     ⇒
. | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | | by AssertBY M(i) ⊆ M(i)
. | |    (AllHyps (λh.Unfold ‘d-sub‘ h THEN BackThru h))
. | |    THEN
. | |    Assert (∀x:Id. M(i).ds(x) ⊆r M(i).ds(x))
. | |
. | |\
. | | .....assertion .....
. | | 1. A : Dsys
. | | 2. B : Dsys
. | | 3. A ⊆ B
. | | 4. w : World
. | | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | | 6. ∀i:Id, a:Action(i).
. | |    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | | 9. ∀i:Id, t:ℕ.
. | |    ¬isnull(a(i;t))
. | |    ⇒
. | |    (islocal(kind(a(i;t)))
. | |     ⇒
. | |     M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | |    & (∀x:Id.
```

```
. | |       M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | |       +1).x))
. | |    & (∀l:IdLnk.
. | |       M(i).send(kind(a(i;t));l;λx.
. | |       s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | |    & (∀x:Id.
. | |       ¬M(i).frame(kind(a(i;t)) affects x)
. | |       ⇒
. | |       s(i;t).x = s(i;t+1).x)
. | |    & (∀l:IdLnk, tg:Id.
. | |       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | |       ⇒
. | |       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | | 10. ∀i,a:Id, t:ℕ.
. | |     ∃t':ℕ.
. | |     t≤t'
. | |     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | |       ∨ ¬a declared in M(i)
. | |       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | | 12. ∀i:Id, a:Action(i).
. | |     ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | | 13. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | | 15. i : Id
. | | 16. t : ℕ
. | | 17. ¬isnull(a(i;t))
. | | 18. (islocal(kind(a(i;t)))
. | |      ⇒
. | |      M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | |     & (∀x:Id.
. | |        M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | |        +1).x))
. | |     & (∀l:IdLnk.
. | |        M(i).send(kind(a(i;t));l;λx.
. | |        s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | |     & (∀x:Id.
. | |        ¬M(i).frame(kind(a(i;t)) affects x)
. | |        ⇒
. | |        s(i;t).x = s(i;t+1).x)
. | |     & (∀l:IdLnk, tg:Id.
. | |        ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | |        ⇒
. | |        w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | | 19. (λx.s(i;t).x) ∈ M(i).state
. | | 20. (λx.s(i;t+1).x) ∈ M(i).state
. | | 21. M(i) ⊆ M(i)
. | | ⊢  ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | |   by Analyze 0
. | |      THEN
. | |      BackThru
. | |      Thm* ∀M1,M2:MsgA, x:Id. M1 ⊆ M2 ⇒ (M2.ds(x) ⊆r M1.ds(x))
. | |      {ma_DASH_ds_DASH_sub}
. | |
. | .\
```

```
. | . 1. A : Dsys
. | . 2. B : Dsys
. | . 3. A ⊆ B
. | . 4. w : World
. | . 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . 6. ∀i:Id, a:Action(i).
. | . |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . 9. ∀i:Id, t:ℕ.
. | . |  ¬isnull(a(i;t))
. | . |  ⇒
. | . |  (islocal(kind(a(i;t)))
. | . |   ⇒
. | . |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . |  & (∀x:Id.
. | . |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . |     +1).x))
. | . |  & (∀l:IdLnk.
. | . |     M(i).send(kind(a(i;t));l;λx.
. | . |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . |  & (∀x:Id.
. | . |     ¬M(i).frame(kind(a(i;t)) affects x)
. | . |     ⇒
. | . |     s(i;t).x = s(i;t+1).x)
. | . |  & (∀l:IdLnk, tg:Id.
. | . |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . |     ⇒
. | . |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . 10. ∀i,a:Id, t:ℕ.
. | . |   ∃t':ℕ.
. | . |   t≤t'
. | . |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . |     ∨ ¬a declared in M(i)
. | . |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . 12. ∀i:Id, a:Action(i).
. | . |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . 13. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . 15. i : Id
. | . 16. t : ℕ
. | . 17. ¬isnull(a(i;t))
. | . 18. (islocal(kind(a(i;t)))
. | . |    ⇒
. | . |    M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . |   & (∀x:Id.
. | . |      M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . |      +1).x))
. | . |   & (∀l:IdLnk.
. | . |      M(i).send(kind(a(i;t));l;λx.
. | . |      s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . |   & (∀x:Id.
. | . |      ¬M(i).frame(kind(a(i;t)) affects x)
. | . |      ⇒
```

```
. | . |       s(i;t).x = s(i;t+1).x)
. | . |    & (∀l:IdLnk, tg:Id.
. | . |       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . |       ⇒
. | . |       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . 19. (λx.s(i;t).x) ∈ M(i).state
. | . 20. (λx.s(i;t+1).x) ∈ M(i).state
. | . 21. M(i) ⊆ M(i)
. | . 22. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . ⊢  (islocal(kind(a(i;t)))
. | . |    ⇒
. | . |    M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . |  & (∀x:Id.
. | . |      M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . |      +1).x))
. | . |  & (∀l:IdLnk.
. | . |      M(i).send(kind(a(i;t));l;λx.
. | . |      s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . |  & (∀x:Id.
. | . |      ¬M(i).frame(kind(a(i;t)) affects x)
. | . |      ⇒
. | . |      s(i;t).x = s(i;t+1).x)
. | . |  & (∀l:IdLnk, tg:Id.
. | . |      ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . |      ⇒
. | . |      w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | by ParallelAnd -5
. | . |     THENL
. | . |     [Id
. | . |     ;ParallelAnd -5
. | . |      THENL
. | . |      [Id;ParallelAnd -5 THENL [Id;ParallelAnd -5]]]
. | . |
. | . |\
. | . | 1. A : Dsys
. | . | 2. B : Dsys
. | . | 3. A ⊆ B
. | . | 4. w : World
. | . | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 6. ∀i:Id, a:Action(i).
. | . | |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 9. ∀i:Id, t:ℕ.
. | . | |  ¬isnull(a(i;t))
. | . | |  ⇒
. | . | |  (islocal(kind(a(i;t)))
. | . | |   ⇒
. | . | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | |  & (∀x:Id.
. | . | |      M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | |      +1).x))
. | . | |  & (∀l:IdLnk.
. | . | |      M(i).send(kind(a(i;t));l;λx.
. | . | |      s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
```

```
. | . | |  & (∀x:Id.
. | . | |     ¬M(i).frame(kind(a(i;t)) affects x)
. | . | |     ⇒
. | . | |     s(i;t).x = s(i;t+1).x)
. | . | |  & (∀l:IdLnk, tg:Id.
. | . | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | |     ⇒
. | . | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | 10. ∀i,a:Id, t:ℕ.
. | . | |   ∃t':ℕ.
. | . | |   t≤t'
. | . | |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | |     ∨ ¬a declared in M(i)
. | . | |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 12. ∀i:Id, a:Action(i).
. | . | |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 13. ∀l:IdLnk, tg:Id.
. | . | |   (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 15. i : Id
. | . | 16. t : ℕ
. | . | 17. ¬isnull(a(i;t))
. | . | 18. islocal(kind(a(i;t)))
. | . | |   ⇒
. | . | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t)))
. | . | 19. (λx.s(i;t).x) ∈ M(i).state
. | . | 20. (λx.s(i;t+1).x) ∈ M(i).state
. | . | 21. M(i) ⊆ M(i)
. | . | 22. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | ⊢  islocal(kind(a(i;t)))
. | . | |  ⇒
. | . | |  M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t)))
. | . | | by Analyze 0 THEN Analyze -6
. | . | |
. | . | 1. A : Dsys
. | . | 2. B : Dsys
. | . | 3. A ⊆ B
. | . | 4. w : World
. | . | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 6. ∀i:Id, a:Action(i).
. | . | |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 9. ∀i:Id, t:ℕ.
. | . | |  ¬isnull(a(i;t))
. | . | |  ⇒
. | . | |  (islocal(kind(a(i;t)))
. | . | |   ⇒
. | . | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | |  & (∀x:Id.
. | . | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | |     +1).x))
. | . | |  & (∀l:IdLnk.
. | . | |     M(i).send(kind(a(i;t));l;λx.
```

```
. | . | |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . | |  & (∀x:Id.
. | . | |     ¬M(i).frame(kind(a(i;t)) affects x)
. | . | |     ⇒
. | . | |     s(i;t).x = s(i;t+1).x)
. | . | |  & (∀l:IdLnk, tg:Id.
. | . | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | |     ⇒
. | . | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | 10. ∀i,a:Id, t:ℕ.
. | . | |   ∃t':ℕ.
. | . | |   t≤t'
. | . | |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | |     ∨ ¬a declared in M(i)
. | . | |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 12. ∀i:Id, a:Action(i).
. | . | |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 13. ∀l:IdLnk, tg:Id.
. | . | |   (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 15. i : Id
. | . | 16. t : ℕ
. | . | 17. ¬isnull(a(i;t))
. | . | 18. (λx.s(i;t).x) ∈ M(i).state
. | . | 19. (λx.s(i;t+1).x) ∈ M(i).state
. | . | 20. M(i) ⊆ M(i)
. | . | 21. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | 22. islocal(kind(a(i;t)))
. | . | 23. M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t)))
. | . | ⊢  M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t)))
. | . | | by Using [`M2',M(i)]
. | . | |    (BackThru
. | . | |     Thm* ∀M1,M2:MsgA.
. | . | |          M1 ⊆ M2
. | . | |          ⇒
. | . | |          (∀a:Id, s:M2.state, v:M2.da(locl(a)).
. | . | |           M2.pre(a,s,v) ⇒ M1.pre(a,s,v))
. | . | |     {ma_DASH_pre_DASH_sub})
. | . | |
. | . | .....wf .....
. | . | 1. A : Dsys
. | . | 2. B : Dsys
. | . | 3. A ⊆ B
. | . | 4. w : World
. | . | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 6. ∀i:Id, a:Action(i).
. | . | |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 9. ∀i:Id, t:ℕ.
. | . | |  ¬isnull(a(i;t))
. | . | |  ⇒
. | . | |  (islocal(kind(a(i;t)))
. | . | |   ⇒
```

```
. | . | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | |  & (∀x:Id.
. | . | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | |     +1).x))
. | . | |  & (∀l:IdLnk.
. | . | |     M(i).send(kind(a(i;t));l;λx.
. | . | |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . | |  & (∀x:Id.
. | . | |     ¬M(i).frame(kind(a(i;t)) affects x)
. | . | |     ⇒
. | . | |     s(i;t).x = s(i;t+1).x)
. | . | |  & (∀l:IdLnk, tg:Id.
. | . | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | |     ⇒
. | . | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | 10. ∀i,a:Id, t:ℕ.
. | . | |   ∃t':ℕ.
. | . | |   t≤t'
. | . | |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | |     ∨ ¬a declared in M(i)
. | . | |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 12. ∀i:Id, a:Action(i).
. | . | |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 13. ∀l:IdLnk, tg:Id.
. | . | |   (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 15. i : Id
. | . | 16. t : ℕ
. | . | 17. ¬isnull(a(i;t))
. | . | 18. (λx.s(i;t).x) ∈ M(i).state
. | . | 19. (λx.s(i;t+1).x) ∈ M(i).state
. | . | 20. M(i) ⊆ M(i)
. | . | 21. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | 22. islocal(kind(a(i;t)))
. | . | 23. M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t)))
. | . | ⊢  val(a(i;t)) ∈ M(i).da(locl(act(kind(a(i;t)))))
. | . | | by SubsumeC valtype(i;a(i;t)) THEN Try PWTrivial
. | . | |    THEN
. | . | |    Using ['B',M(i).da(kind(a(i;t)))]
. | . | |    (BackThru
. | . | |     Thm* (A ⊆r B) ⇒ (B ⊆r C) ⇒ (A ⊆r C)
. | . | |     {subtype_rel_transitivity})
. | . | |
. | . | |\
. | . | | 1. A : Dsys
. | . | | 2. B : Dsys
. | . | | 3. A ⊆ B
. | . | | 4. w : World
. | . | | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | | 6. ∀i:Id, a:Action(i).
. | . | |    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | | 7. ∀l:IdLnk, tg:Id.
. | . | |    (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
```

```
. | . | | 9. ∀i:Id, t:ℕ.
. | . | |    ¬isnull(a(i;t))
. | . | |    ⇒
. | . | |    (islocal(kind(a(i;t)))
. | . | |     ⇒
. | . | |     M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | |    & (∀x:Id.
. | . | |       M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | |       +1).x))
. | . | |    & (∀l:IdLnk.
. | . | |       M(i).send(kind(a(i;t));l;λx.
. | . | |       s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . | |    & (∀x:Id.
. | . | |       ¬M(i).frame(kind(a(i;t)) affects x)
. | . | |       ⇒
. | . | |       s(i;t).x = s(i;t+1).x)
. | . | |    & (∀l:IdLnk, tg:Id.
. | . | |       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | |       ⇒
. | . | |       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | | 10. ∀i,a:Id, t:ℕ.
. | . | |     ∃t':ℕ.
. | . | |     t≤t'
. | . | |     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | |       ∨ ¬a declared in M(i)
. | . | |       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | | 12. ∀i:Id, a:Action(i).
. | . | |     ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | | 13. ∀l:IdLnk, tg:Id.
. | . | |     (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)

. | . | | 15. i : Id
. | . | | 16. t : ℕ
. | . | | 17. ¬isnull(a(i;t))
. | . | | 18. (λx.s(i;t).x) ∈ M(i).state
. | . | | 19. (λx.s(i;t+1).x) ∈ M(i).state
. | . | | 20. M(i) ⊆ M(i)
. | . | | 21. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | | 22. islocal(kind(a(i;t)))
. | . | | 23. M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t)))
. | . | | ⊢  valtype(i;a(i;t)) ⊆r M(i).da(kind(a(i;t)))
. | . | |   by BackThruSomeHyp
. | . | |
. | . | .\
. | . | . 1. A : Dsys
. | . | . 2. B : Dsys
. | . | . 3. A ⊆ B
. | . | . 4. w : World
. | . | . 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | . 6. ∀i:Id, a:Action(i).
. | . | . |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | . 7. ∀l:IdLnk, tg:Id.
. | . | . |  (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | . 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
```

```
. | . | . 9. ∀i:Id, t:ℕ.
. | . | . |  ¬isnull(a(i;t))
. | . | . |  ⇒
. | . | . |  (islocal(kind(a(i;t)))
. | . | . |   ⇒
. | . | . |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | . |  & (∀x:Id.
. | . | . |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | . |     +1).x))
. | . | . |  & (∀l:IdLnk.
. | . | . |     M(i).send(kind(a(i;t));l;λx.
. | . | . |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . | . |  & (∀x:Id.
. | . | . |     ¬M(i).frame(kind(a(i;t)) affects x)
. | . | . |     ⇒
. | . | . |     s(i;t).x = s(i;t+1).x)
. | . | . |  & (∀l:IdLnk, tg:Id.
. | . | . |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | . |     ⇒
. | . | . |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | . 10. ∀i,a:Id, t:ℕ.
. | . | . |   ∃t':ℕ.
. | . | . |   t≤t'
. | . | . |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | . |     ∨ ¬a declared in M(i)
. | . | . |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | . 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | . 12. ∀i:Id, a:Action(i).
. | . | . |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | . 13. ∀l:IdLnk, tg:Id.
. | . | . |   (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | . 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | . 15. i : Id
. | . | . 16. t : ℕ
. | . | . 17. ¬isnull(a(i;t))
. | . | . 18. (λx.s(i;t).x) ∈ M(i).state
. | . | . 19. (λx.s(i;t+1).x) ∈ M(i).state
. | . | . 20. M(i) ⊆ M(i)
. | . | . 21. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | . 22. islocal(kind(a(i;t)))
. | . | . 23. M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t)))
. | . | . ⊢  M(i).da(kind(a(i;t))) ⊆r M(i).da(locl(act(kind(a(i;t)))))
. | . | . | by Subst (locl(act(kind(a(i;t)))) ~ kind(a(i;t))) 0
. | . | . |
. | . | . |\
. | . | . | .....equality .....
. | . | . | 1. A : Dsys
. | . | . | 2. B : Dsys
. | . | . | 3. A ⊆ B
. | . | . | 4. w : World
. | . | . | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | . | 6. ∀i:Id, a:Action(i).
. | . | . |    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | . | 7. ∀l:IdLnk, tg:Id.
. | . | . |    (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
```

```
. | . | . | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | . | 9. ∀i:Id, t:ℕ.
. | . | . |    ¬isnull(a(i;t))
. | . | . |    ⇒
. | . | . |    (islocal(kind(a(i;t)))
. | . | . |     ⇒
. | . | . |     M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | . |    & (∀x:Id.
. | . | . |       M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | . |       +1).x))
. | . | . |    & (∀l:IdLnk.
. | . | . |       M(i).send(kind(a(i;t));l;λx.
. | . | . |       s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . | . |    & (∀x:Id.
. | . | . |       ¬M(i).frame(kind(a(i;t)) affects x)
. | . | . |       ⇒
. | . | . |       s(i;t).x = s(i;t+1).x)
. | . | . |    & (∀l:IdLnk, tg:Id.
. | . | . |       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | . |       ⇒
. | . | . |       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | . | 10. ∀i,a:Id, t:ℕ.
. | . | . |     ∃t':ℕ.
. | . | . |     t≤t'
. | . | . |     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | . |       ∨ ¬a declared in M(i)
. | . | . |       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | . | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | . | 12. ∀i:Id, a:Action(i).
. | . | . |     ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | . | 13. ∀l:IdLnk, tg:Id.
. | . | . |     (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | . | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | . | 15. i : Id
. | . | . | 16. t : ℕ
. | . | . | 17. ¬isnull(a(i;t))
. | . | . | 18. (λx.s(i;t).x) ∈ M(i).state
. | . | . | 19. (λx.s(i;t+1).x) ∈ M(i).state
. | . | . | 20. M(i) ⊆ M(i)
. | . | . | 21. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | . | 22. islocal(kind(a(i;t)))
. | . | . | 23. M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t)))
. | . | . | ⊢  locl(act(kind(a(i;t)))) ~ kind(a(i;t))
. | . | . |   by MoveToConcl -2 THEN GenConclAtAddr [1;1;1]
. | . | . |      THEN
. | . | . |      Analyze -2
. | . | . |      THEN
. | . | . |      Unfolds [`locl`;`actof`;`islocal`] 0
. | . | . |      THEN
. | . | . |      Reduce 0
. | . | . |
. | . | . .\
. | . | . . 1. A : Dsys
. | . | . . 2. B : Dsys
. | . | . . 3. A ⊆ B
```

```
. | . | . . 4. w : World
. | . | . . 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | . . 6. ∀i:Id, a:Action(i).
. | . | . .    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | . . 7. ∀l:IdLnk, tg:Id.
. | . | . .    (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | . . 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | . . 9. ∀i:Id, t:ℕ.
. | . | . .    ¬isnull(a(i;t))
. | . | . .    ⇒
. | . | . .    (islocal(kind(a(i;t)))
. | . | . .     ⇒
. | . | . .     M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | . .    & (∀x:Id.
. | . | . .       M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | . .       +1).x))
. | . | . .    & (∀l:IdLnk.
. | . | . .       M(i).send(kind(a(i;t));l;λx.
. | . | . .       s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . | . .    & (∀x:Id.
. | . | . .       ¬M(i).frame(kind(a(i;t)) affects x)
. | . | . .       ⇒
. | . | . .       s(i;t).x = s(i;t+1).x)
. | . | . .    & (∀l:IdLnk, tg:Id.
. | . | . .       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | . .       ⇒
. | . | . .       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | . . 10. ∀i,a:Id, t:ℕ.
. | . | . .     ∃t':ℕ.
. | . | . .     t≤t'
. | . | . .     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | . .       ∨ ¬a declared in M(i)
. | . | . .       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | . . 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | . . 12. ∀i:Id, a:Action(i).
. | . | . .     ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | . . 13. ∀l:IdLnk, tg:Id.
. | . | . .     (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | . . 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | . . 15. i : Id
. | . | . . 16. t : ℕ
. | . | . . 17. ¬isnull(a(i;t))
. | . | . . 18. (λx.s(i;t).x) ∈ M(i).state
. | . | . . 19. (λx.s(i;t+1).x) ∈ M(i).state
. | . | . . 20. M(i) ⊆ M(i)
. | . | . . 21. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | . . 22. islocal(kind(a(i;t)))
. | . | . . 23. M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t)))
. | . | . . ⊢  M(i).da(kind(a(i;t))) ⊆r M(i).da(kind(a(i;t)))
. | . | . .   by Auto
. | . |
. | . |\
. | . | 1. A : Dsys
. | . | 2. B : Dsys
. | . | 3. A ⊆ B
```

```
. | . | 4. w : World
. | . | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 6. ∀i:Id, a:Action(i).
. | . | |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 9. ∀i:Id, t:ℕ.
. | . | |  ¬isnull(a(i;t))
. | . | |  ⇒
. | . | |  (islocal(kind(a(i;t)))
. | . | |   ⇒
. | . | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | |  & (∀x:Id.
. | . | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | |     +1).x))
. | . | |  & (∀l:IdLnk.
. | . | |     M(i).send(kind(a(i;t));l;λx.
. | . | |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . | |  & (∀x:Id.
. | . | |     ¬M(i).frame(kind(a(i;t)) affects x)
. | . | |     ⇒
. | . | |     s(i;t).x = s(i;t+1).x)
. | . | |  & (∀l:IdLnk, tg:Id.
. | . | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | |     ⇒
. | . | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | 10. ∀i,a:Id, t:ℕ.
. | . | |   ∃t':ℕ.
. | . | |   t≤t'
. | . | |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | |     ∨ ¬a declared in M(i)
. | . | |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 12. ∀i:Id, a:Action(i).
. | . | |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 13. ∀l:IdLnk, tg:Id.
. | . | |   (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 15. i : Id
. | . | 16. t : ℕ
. | . | 17. ¬isnull(a(i;t))
. | . | 18. ∀x:Id.
. | . | |   M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t+1).x)
. | . | 19. (λx.s(i;t).x) ∈ M(i).state
. | . | 20. (λx.s(i;t+1).x) ∈ M(i).state
. | . | 21. M(i) ⊆ M(i)
. | . | 22. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | ⊢  ∀x:Id.
. | . | |  M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t+1).x)
. | . | | by ParallelOp -5
. | . | |    THEN
. | . | |    Using [`M2',M(i)]
. | . | |    (BackThru
. | . | |     Thm* ∀M1,M2:MsgA.
. | . | |          M1 ⊆ M2
```

```
. | . | |          ⇒
. | . | |          (∀k:Knd, x:Id, s:M2.state, v:M2.da(k), w:M2.ds(x).
. | . | |           M2.ef(k,x,s,v,w) ⇒ M1.ef(k,x,s,v,w))
. | . | |     {ma_DASH_ef_DASH_sub})
. | . | |
. | . | |\
. | . | | .....wf .....
. | . | | 1. A : Dsys
. | . | | 2. B : Dsys
. | . | | 3. A ⊆ B
. | . | | 4. w : World
. | . | | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | | 6. ∀i:Id, a:Action(i).
. | . | |    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | | 7. ∀l:IdLnk, tg:Id.
. | . | |    (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | | 9. ∀i:Id, t:ℕ.
. | . | |     ¬isnull(a(i;t))
. | . | |     ⇒
. | . | |    (islocal(kind(a(i;t)))
. | . | |     ⇒
. | . | |     M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | |    & (∀x:Id.
. | . | |       M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | |       +1).x))
. | . | |    & (∀l:IdLnk.
. | . | |       M(i).send(kind(a(i;t));l;λx.
. | . | |       s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . | |    & (∀x:Id.
. | . | |       ¬M(i).frame(kind(a(i;t)) affects x)
. | . | |       ⇒
. | . | |       s(i;t).x = s(i;t+1).x)
. | . | |    & (∀l:IdLnk, tg:Id.
. | . | |       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | |       ⇒
. | . | |       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | | 10. ∀i,a:Id, t:ℕ.
. | . | |     ∃t':ℕ.
. | . | |     t≤t'
. | . | |     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | |       ∨ ¬a declared in M(i)
. | . | |       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | | 12. ∀i:Id, a:Action(i).
. | . | |     ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | | 13. ∀l:IdLnk, tg:Id.
. | . | |     (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | | 15. i : Id
. | . | | 16. t : ℕ
. | . | | 17. ¬isnull(a(i;t))
. | . | | 18. ∀x:Id.
. | . | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | |     +1).x)
```

```
. | . | | 19. (λx.s(i;t).x) ∈ M(i).state
. | . | | 20. (λx.s(i;t+1).x) ∈ M(i).state
. | . | | 21. M(i) ⊆ M(i)
. | . | | 22. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | | 23. x : Id
. | . | | 24. M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | |     +1).x)
. | . | | ⊢  val(a(i;t)) ∈ M(i).da(kind(a(i;t)))
. | . | |   by DoSubsume THEN BackThruSomeHyp
. | . | |
. | . | .\
. | . | . .....wf .....
. | . | . 1. A : Dsys
. | . | . 2. B : Dsys
. | . | . 3. A ⊆ B
. | . | . 4. w : World
. | . | . 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | . 6. ∀i:Id, a:Action(i).
. | . | .    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | . 7. ∀l:IdLnk, tg:Id.
. | . | .    (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | . 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | . 9. ∀i:Id, t:ℕ.
. | . | .    ¬isnull(a(i;t))
. | . | .    ⇒
. | . | .    (islocal(kind(a(i;t)))
. | . | .     ⇒
. | . | .     M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | .    & (∀x:Id.
. | . | .       M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | .       +1).x))
. | . | .    & (∀l:IdLnk.
. | . | .       M(i).send(kind(a(i;t));l;λx.
. | . | .       s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . | .    & (∀x:Id.
. | . | .       ¬M(i).frame(kind(a(i;t)) affects x)
. | . | .       ⇒
. | . | .       s(i;t).x = s(i;t+1).x)
. | . | .    & (∀l:IdLnk, tg:Id.
. | . | .       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | .       ⇒
. | . | .       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | . 10. ∀i,a:Id, t:ℕ.
. | . | .     ∃t':ℕ.
. | . | .     t≤t'
. | . | .     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | .       ∨ ¬a declared in M(i)
. | . | .       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | . 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | . 12. ∀i:Id, a:Action(i).
. | . | .     ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | . 13. ∀l:IdLnk, tg:Id.
. | . | .     (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | . 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | . 15. i : Id
```

```
. | . | . 16. t : ℕ
. | . | . 17. ¬isnull(a(i;t))
. | . | . 18. ∀x:Id.
. | . | .     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | .     +1).x)
. | . | . 19. (λx.s(i;t).x) ∈ M(i).state
. | . | . 20. (λx.s(i;t+1).x) ∈ M(i).state
. | . | . 21. M(i) ⊆ M(i)
. | . | . 22. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | . 23. x : Id
. | . | . 24. M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | .     +1).x)
. | . | . ⊢  s(i;t+1).x ∈ M(i).ds(x)
. | . | .   by AllHyps
. | . | .      (λh.
. | . | .       Unfold ‘ma-st‘ h THEN Unfold ‘ma-state‘ h
. | . | .       THEN
. | . | .       Fold ‘ma-ds‘ h)
. | . | .      THEN
. | . | .      Subst (s(i;t+1).x ~ ((λx.s(i;t+1).x)(x))) 0
. | . | .      THENL
. | . | .      [Reduce 0;Id]
. | . |
. | . |\
. | . | 1. A : Dsys
. | . | 2. B : Dsys
. | . | 3. A ⊆ B
. | . | 4. w : World
. | . | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 6. ∀i:Id, a:Action(i).
. | . | |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 9. ∀i:Id, t:ℕ.
. | . | |  ¬isnull(a(i;t))
. | . | |  ⇒
. | . | |  (islocal(kind(a(i;t)))
. | . | |   ⇒
. | . | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | |  & (∀x:Id.
. | . | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | |     +1).x))
. | . | |  & (∀l:IdLnk.
. | . | |     M(i).send(kind(a(i;t));l;λx.
. | . | |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . | |  & (∀x:Id.
. | . | |     ¬M(i).frame(kind(a(i;t)) affects x)

. | . | |     ⇒
. | . | |     s(i;t).x = s(i;t+1).x)
. | . | |  & (∀l:IdLnk, tg:Id.
. | . | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | |     ⇒
. | . | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | 10. ∀i,a:Id, t:ℕ.
. | . | |   ∃t':ℕ.
```

```
. | . | |   t≤t'
. | . | |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | |     ∨ ¬a declared in M(i)
. | . | |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 12. ∀i:Id, a:Action(i).
. | . | |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 13. ∀l:IdLnk, tg:Id.
. | . | |   (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 15. i : Id
. | . | 16. t : ℕ
. | . | 17. ¬isnull(a(i;t))
. | . | 18. ∀l:IdLnk.
. | . | |   M(i).send(kind(a(i;t));l;λx.
. | . | |   s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i)
. | . | 19. (λx.s(i;t).x) ∈ M(i).state
. | . | 20. (λx.s(i;t+1).x) ∈ M(i).state
. | . | 21. M(i) ⊆ M(i)
. | . | 22. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | ⊢  ∀l:IdLnk.
. | . | |  M(i).send(kind(a(i;t));l;λx.
. | . | |  s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i)
. | . | | by ParallelOp -5
. | . | |    THEN
. | . | |    Using [`M2',M(i)]
. | . | |    (BackThru
. | . | |     Thm* ∀M1,M2:MsgA.
. | . | |          M1 ⊆ M2
. | . | |          ⇒
. | . | |          (∀k:Knd, l:IdLnk, s:M2.state, v:M2.da(k), i:Id,
. | . | |            ms:(tg:Id
. | . | |                 ×if source(l) = i→ M2.da(rcv(l; tg))
. | . | |                  else Top fi) List.
. | . | |           M2.send(k;l;s;v;ms;i) ⇒ M1.send(k;l;s;v;ms;i))
. | . | |     {ma_DASH_send_DASH_sub})
. | . | |    THEN
. | . | |    Try (Complete Auto)
. | . | |    THEN
. | . | |    Try (DoSubsume THEN BackThruSomeHyp)
. | . | |
. | . | .....wf .....
. | . | 1. A : Dsys
. | . | 2. B : Dsys
. | . | 3. A ⊆ B
. | . | 4. w : World
. | . | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 6. ∀i:Id, a:Action(i).
. | . | |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 9. ∀i:Id, t:ℕ.
. | . | |  ¬isnull(a(i;t))
. | . | |  ⇒
. | . | |  (islocal(kind(a(i;t)))
```

```
. | . | |   ⇒
. | . | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | |  & (∀x:Id.
. | . | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | |     +1).x))
. | . | |  & (∀l:IdLnk.
. | . | |     M(i).send(kind(a(i;t));l;λx.
. | . | |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . | |  & (∀x:Id.
. | . | |     ¬M(i).frame(kind(a(i;t)) affects x)
. | . | |     ⇒
. | . | |     s(i;t).x = s(i;t+1).x)
. | . | |  & (∀l:IdLnk, tg:Id.
. | . | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | |     ⇒
. | . | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | 10. ∀i,a:Id, t:ℕ.
. | . | |   ∃t':ℕ.
. | . | |   t≤t'
. | . | |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | |     ∨ ¬a declared in M(i)
. | . | |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 12. ∀i:Id, a:Action(i).
. | . | |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 13. ∀l:IdLnk, tg:Id.
. | . | |   (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 15. i : Id
. | . | 16. t : ℕ
. | . | 17. ¬isnull(a(i;t))
. | . | 18. ∀l:IdLnk.
. | . | |   M(i).send(kind(a(i;t));l;λx.
. | . | |   s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i)
. | . | 19. (λx.s(i;t).x) ∈ M(i).state
. | . | 20. (λx.s(i;t+1).x) ∈ M(i).state
. | . | 21. M(i) ⊆ M(i)
. | . | 22. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | 23. l : IdLnk
. | . | 24. M(i).send(kind(a(i;t));l;λx.
. | . | |   s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i)
. | . | ⊢  withlnk(l;m(i;t))
. | . | |  ∈ (tg:Id
. | . | |     ×if source(l) = i→ M(i).da(rcv(l; tg)) else Top fi) List
. | . | | by Using [`the_w',w] Auto
. | . | |    THEN
. | . | |    Try (Analyze -1 THEN Unfold `w-Msg` 0)
. | . | |    THEN
. | . | |    SplitOnConclITE
. | . | |
. | . | .....truecase .....
. | . | 1. A : Dsys
. | . | 2. B : Dsys
. | . | 3. A ⊆ B
. | . | 4. w : World
```

```
. | . | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 6. ∀i:Id, a:Action(i).
. | . |    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 9. ∀i:Id, t:ℕ.
. | . |    ¬isnull(a(i;t))
. | . |    ⇒
. | . |    (islocal(kind(a(i;t)))
. | . |     ⇒
. | . |     M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . |    & (∀x:Id.
. | . |       M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . |       +1).x))
. | . |    & (∀l:IdLnk.
. | . |       M(i).send(kind(a(i;t));l;λx.
. | . |       s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . |    & (∀x:Id.
. | . |       ¬M(i).frame(kind(a(i;t)) affects x)
. | . |       ⇒
. | . |       s(i;t).x = s(i;t+1).x)
. | . |    & (∀l:IdLnk, tg:Id.
. | . |       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . |       ⇒
. | . |       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | 10. ∀i,a:Id, t:ℕ.
. | . |     ∃t':ℕ.
. | . |     t≤t'
. | . |     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . |       ∨ ¬a declared in M(i)
. | . |       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 12. ∀i:Id, a:Action(i).
. | . |     ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 13. ∀l:IdLnk, tg:Id.
. | . |     (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 15. i : Id
. | . | 16. t : ℕ
. | . | 17. ¬isnull(a(i;t))
. | . | 18. ∀l:IdLnk.
. | . |     M(i).send(kind(a(i;t));l;λx.
. | . |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i)
. | . | 19. (λx.s(i;t).x) ∈ M(i).state
. | . | 20. (λx.s(i;t+1).x) ∈ M(i).state
. | . | 21. M(i) ⊆ M(i)
. | . | 22. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | 23. l : IdLnk
. | . | 24. M(i).send(kind(a(i;t));l;λx.
. | . |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i)
. | . | 25. withlnk(l;m(i;t)) = withlnk(l;m(i;t))
. | . | 26. (t:Id×w.M(l,t)) List
. | . | 27. (t:Id×w.M(l,t)) List
. | . | 28. t:Id×w.M(l,t)
. | . | 29. t1 : Id
```

```
. | . | 30. source(l) = i
. | . | ⊢  (w.M(l,t1)) ⊆ M(i).da(rcv(l; t1))
. | . |   by Analyze 0 THEN DoSubsume THEN RevHypSubst -2 0
. | . |      THEN
. | . |      BackThruSomeHyp
. | . |
. | . |\
. | . | 1. A : Dsys
. | . | 2. B : Dsys
. | . | 3. A ⊆ B
. | . | 4. w : World
. | . | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 6. ∀i:Id, a:Action(i).
. | . | |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 9. ∀i:Id, t:ℕ.
. | . | |  ¬isnull(a(i;t))
. | . | |  ⇒
. | . | |  (islocal(kind(a(i;t)))
. | . | |   ⇒
. | . | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | |  & (∀x:Id.
. | . | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | |     +1).x))
. | . | |  & (∀l:IdLnk.
. | . | |     M(i).send(kind(a(i;t));l;λx.
. | . | |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . | |  & (∀x:Id.
. | . | |     ¬M(i).frame(kind(a(i;t)) affects x)
. | . | |     ⇒
. | . | |     s(i;t).x = s(i;t+1).x)
. | . | |  & (∀l:IdLnk, tg:Id.
. | . | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | |     ⇒
. | . | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | 10. ∀i,a:Id, t:ℕ.
. | . | |   ∃t':ℕ.
. | . | |   t≤t'
. | . | |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | |     ∨ ¬a declared in M(i)
. | . | |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | 12. ∀i:Id, a:Action(i).
. | . | |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | 13. ∀l:IdLnk, tg:Id.
. | . | |   (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | 15. i : Id
. | . | 16. t : ℕ
. | . | 17. ¬isnull(a(i;t))
. | . | 18. ∀x:Id.
. | . | |   ¬M(i).frame(kind(a(i;t)) affects x)
. | . | |   ⇒
. | . | |   s(i;t).x = s(i;t+1).x
```

```
. | . | 19. (λx.s(i;t).x) ∈ M(i).state
. | . | 20. (λx.s(i;t+1).x) ∈ M(i).state
. | . | 21. M(i) ⊆ M(i)
. | . | 22. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | ⊢  ∀x:Id.
. | . | |  ¬M(i).frame(kind(a(i;t)) affects x)
. | . | |  ⇒
. | . | |  s(i;t).x = s(i;t+1).x  by ParallelOp -5 THEN ParallelOp -1
. | . | |
. | . | |\
. | . | | .....antecedent .....
. | . | | 1. A : Dsys
. | . | | 2. B : Dsys
. | . | | 3. A ⊆ B
. | . | | 4. w : World
. | . | | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | | 6. ∀i:Id, a:Action(i).
. | . | |    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | | 7. ∀l:IdLnk, tg:Id.
. | . | |    (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | | 9. ∀i:Id, t:ℕ.
. | . | |    ¬isnull(a(i;t))
. | . | |    ⇒
. | . | |    (islocal(kind(a(i;t)))
. | . | |     ⇒
. | . | |     M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | |    & (∀x:Id.
. | . | |       M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | |       +1).x))
. | . | |    & (∀l:IdLnk.
. | . | |       M(i).send(kind(a(i;t));l;λx.
. | . | |       s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . | |    & (∀x:Id.
. | . | |       ¬M(i).frame(kind(a(i;t)) affects x)
. | . | |       ⇒
. | . | |       s(i;t).x = s(i;t+1).x)
. | . | |    & (∀l:IdLnk, tg:Id.
. | . | |       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | |       ⇒
. | . | |       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | | 10. ∀i,a:Id, t:ℕ.
. | . | |     ∃t':ℕ.
. | . | |     t≤t'
. | . | |     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | |       ∨ ¬a declared in M(i)
. | . | |       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | | 12. ∀i:Id, a:Action(i).
. | . | |     ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | | 13. ∀l:IdLnk, tg:Id.
. | . | |     (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | | 15. i : Id
. | . | | 16. t : ℕ
```

```
. | . | | 17. ¬isnull(a(i;t))
. | . | | 18. ∀x:Id.
. | . | |     ¬M(i).frame(kind(a(i;t)) affects x)
. | . | |     ⇒
. | . | |     s(i;t).x = s(i;t+1).x
. | . | | 19. (λx.s(i;t).x) ∈ M(i).state
. | . | | 20. (λx.s(i;t+1).x) ∈ M(i).state
. | . | | 21. M(i) ⊆ M(i)
. | . | | 22. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | | 23. x : Id
. | . | | 24. ¬M(i).frame(kind(a(i;t)) affects x)
. | . | | ⊢  ¬M(i).frame(kind(a(i;t)) affects x)
. | . | |   by ParallelOp -1
. | . | |      THEN
. | . | |      Using [‘M2’,M(i)]
. | . | |      (BackThru
. | . | |       Thm* ∀M1,M2:MsgA.
. | . | |             M1 ⊆ M2
. | . | |             ⇒
. | . | |             (∀k:Knd, x:Id.
. | . | |              M2.frame(k affects x) ⇒ M1.frame(k affects x))
. | . | |       {ma_DASH_frame_DASH_sub})
. | . | |
. | . | .\
. | . | . 1. A : Dsys
. | . | . 2. B : Dsys
. | . | . 3. A ⊆ B
. | . | . 4. w : World
. | . | . 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | . 6. ∀i:Id, a:Action(i).
. | . | .     ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | . 7. ∀l:IdLnk, tg:Id.
. | . | .    (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | . 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | . 9. ∀i:Id, t:ℕ.
. | . | .     ¬isnull(a(i;t))
. | . | .     ⇒
. | . | .    (islocal(kind(a(i;t)))
. | . | .     ⇒
. | . | .     M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . | .    & (∀x:Id.
. | . | .       M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . | .       +1).x))
. | . | .    & (∀l:IdLnk.
. | . | .       M(i).send(kind(a(i;t));l;λx.
. | . | .       s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . | .    & (∀x:Id.
. | . | .       ¬M(i).frame(kind(a(i;t)) affects x)
. | . | .       ⇒
. | . | .       s(i;t).x = s(i;t+1).x)
. | . | .    & (∀l:IdLnk, tg:Id.
. | . | .       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . | .       ⇒
. | . | .       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . | . 10. ∀i,a:Id, t:ℕ.
```

```
. | . | .     ∃t':ℕ.
. | . | .     t≤t'
. | . | .     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . | .       ∨ ¬a declared in M(i)
. | . | .       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . | . 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . | . 12. ∀i:Id, a:Action(i).
. | . | .     ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . | . 13. ∀l:IdLnk, tg:Id.
. | . | .     (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . | . 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . | . 15. i : Id
. | . | . 16. t : ℕ
. | . | . 17. ¬isnull(a(i;t))
. | . | . 18. ∀x:Id.
. | . | .     ¬M(i).frame(kind(a(i;t)) affects x)
. | . | .     ⇒
. | . | .     s(i;t).x = s(i;t+1).x
. | . | . 19. (λx.s(i;t).x) ∈ M(i).state
. | . | . 20. (λx.s(i;t+1).x) ∈ M(i).state
. | . | . 21. M(i) ⊆ M(i)
. | . | . 22. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . | . 23. x : Id
. | . | . 24. ¬M(i).frame(kind(a(i;t)) affects x)
. | . | . 25. s(i;t).x = s(i;t+1).x
. | . | . ⊢  s(i;t).x = s(i;t+1).x
. | . | .   by HypSubst -1 0 THEN Try (Fold `member` 0) THEN DoSubsume
. | . | .      THEN
. | . | .      Try BackThruSomeHyp
. | . |
. | . .\
. | . . 1. A : Dsys
. | . . 2. B : Dsys
. | . . 3. A ⊆ B
. | . . 4. w : World
. | . . 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . . 6. ∀i:Id, a:Action(i).
. | . .    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . . 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . . 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . . 9. ∀i:Id, t:ℕ.
. | . .    ¬isnull(a(i;t))
. | . .    ⇒
. | . .    (islocal(kind(a(i;t)))
. | . .     ⇒
. | . .     M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. | . .    & (∀x:Id.
. | . .       M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. | . .       +1).x))
. | . .    & (∀l:IdLnk.
. | . .       M(i).send(kind(a(i;t));l;λx.
. | . .       s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. | . .    & (∀x:Id.
. | . .       ¬M(i).frame(kind(a(i;t)) affects x)
. | . .       ⇒
```

```
. | . .       s(i;t).x = s(i;t+1).x)
. | . .    & (∀l:IdLnk, tg:Id.
. | . .       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . .       ⇒
. | . .       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. | . . 10. ∀i,a:Id, t:ℕ.
. | . .     ∃t':ℕ.
. | . .     t≤t'
. | . .     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. | . .       ∨ ¬a declared in M(i)
. | . .       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. | . . 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. | . . 12. ∀i:Id, a:Action(i).
. | . .     ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. | . . 13. ∀l:IdLnk, tg:Id.
. | . .     (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. | . . 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. | . . 15. i : Id
. | . . 16. t : ℕ
. | . . 17. ¬isnull(a(i;t))
. | . . 18. ∀l:IdLnk, tg:Id.
. | . .     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . .     ⇒
. | . .     w-tagged(tg; onlnk(l;m(i;t))) = nil
. | . . 19. (λx.s(i;t).x) ∈ M(i).state
. | . . 20. (λx.s(i;t+1).x) ∈ M(i).state
. | . . 21. M(i) ⊆ M(i)
. | . . 22. ∀x:Id. M(i).ds(x) ⊆r M(i).ds(x)
. | . . ⊢  ∀l:IdLnk, tg:Id.
. | . .    ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. | . .    ⇒
. | . .    w-tagged(tg; onlnk(l;m(i;t))) = nil
. | . .   by ParallelOp -5 THEN ParallelOp -1 THEN ParallelOp -1
. | . .      THEN
. | . .      ParallelOp -1
. | . .      THEN
. | . .      Using [`M2',M(i)]
. | . .      (BackThru
. | . .       Thm* ∀M1,M2:MsgA.
. | . .             M1 ⊆ M2
. | . .             ⇒
. | . .             (∀k:Knd, l:IdLnk, tg:Id.
. | . .              M2.sframe(k sends <l,tg>)
. | . .              ⇒
. | . .              M1.sframe(k sends <l,tg>))
. | . .       {ma_DASH_sframe_DASH_sub})
. |
. .\
. . 1. A : Dsys
. . 2. B : Dsys
. . 3. A ⊆ B
. . 4. w : World
. . 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. . 6. ∀i:Id, a:Action(i).
. . |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
```

```
. . 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. . 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. . 9. ∀i:Id, t:ℕ.
. . |  ¬isnull(a(i;t))
. . |  ⇒
. . |  (islocal(kind(a(i;t)))
. . |   ⇒
. . |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. . |  & (∀x:Id.
. . |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t+1).x))
. . |  & (∀l:IdLnk.
. . |     M(i).send(kind(a(i;t));l;λx.
. . |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. . |  & (∀x:Id.
. . |     ¬M(i).frame(kind(a(i;t)) affects x)
. . |     ⇒
. . |     s(i;t).x = s(i;t+1).x)
. . |  & (∀l:IdLnk, tg:Id.
. . |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. . |     ⇒
. . |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. . 10. ∀i,a:Id, t:ℕ.
. . |   ∃t':ℕ.
. . |   t≤t'
. . |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . |     ∨ ¬a declared in M(i)
. . |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. . 12. ∀i:Id, a:Action(i).
. . |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. . 13. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. . 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. . 15. ∀i:Id, t:ℕ.
. . |   ¬isnull(a(i;t))
. . |   ⇒
. . |   (islocal(kind(a(i;t)))
. . |    ⇒
. . |    M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. . |   & (∀x:Id.
. . |      M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t+1).x))
. . |   & (∀l:IdLnk.
. . |      M(i).send(kind(a(i;t));l;λx.
. . |      s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. . |   & (∀x:Id.
. . |      ¬M(i).frame(kind(a(i;t)) affects x)
. . |      ⇒
. . |      s(i;t).x = s(i;t+1).x)
. . |   & (∀l:IdLnk, tg:Id.
. . |      ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. . |      ⇒
. . |      w-tagged(tg; onlnk(l;m(i;t))) = nil)
. . ⊢  ∀i,a:Id, t:ℕ.

. . |  ∃t':ℕ.
. . |  t≤t'
. . |  & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
```

```
. . |    ∨ ¬a declared in M(i)
. . |    ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . | by ParallelOp -6
. . |    THEN
. . |    RepeatFor 5
. . |    (ParallelOp -1
. . |     THEN
. . |     Try
. . |     (Unfold `ma-st` 0 THEN Unfold `ma-state` 0
. . |      THEN
. . |      Fold `ma-ds` 0
. . |      THEN
. . |      DoSubsume
. . |      THEN
. . |      BackThruSomeHyp))
. . |    THEN
. . |    Decide a declared in M(i)
. . |
. . |\
. . | 1. A : Dsys
. . | 2. B : Dsys
. . | 3. A ⊆ B
. . | 4. w : World
. . | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. . | 6. ∀i:Id, a:Action(i).
. . | |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. . | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. . | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. . | 9. ∀i:Id, t:ℕ.
. . | |  ¬isnull(a(i;t))
. . | |  ⇒
. . | |  (islocal(kind(a(i;t)))
. . | |   ⇒
. . | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. . | |  & (∀x:Id.
. . | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. . | |     +1).x))
. . | |  & (∀l:IdLnk.
. . | |     M(i).send(kind(a(i;t));l;λx.
. . | |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. . | |  & (∀x:Id.
. . | |     ¬M(i).frame(kind(a(i;t)) affects x)
. . | |     ⇒
. . | |     s(i;t).x = s(i;t+1).x)
. . | |  & (∀l:IdLnk, tg:Id.
. . | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. . | |     ⇒
. . | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. . | 10. ∀i,a:Id, t:ℕ.
. . | |   ∃t':ℕ.
. . | |   t≤t'
. . | |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . | |     ∨ ¬a declared in M(i)
. . | |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
```

```
. . | 12. ∀i:Id, a:Action(i).
. . | |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. . | 13. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. . | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. . | 15. ∀i:Id, t:ℕ.
. . | |   ¬isnull(a(i;t))
. . | |   ⇒
. . | |   (islocal(kind(a(i;t)))
. . | |    ⇒
. . | |    M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. . | |   & (∀x:Id.
. . | |      M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. . | |      +1).x))
. . | |   & (∀l:IdLnk.
. . | |      M(i).send(kind(a(i;t));l;λx.
. . | |      s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. . | |   & (∀x:Id.
. . | |      ¬M(i).frame(kind(a(i;t)) affects x)
. . | |      ⇒
. . | |      s(i;t).x = s(i;t+1).x)
. . | |   & (∀l:IdLnk, tg:Id.
. . | |      ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. . | |      ⇒
. . | |      w-tagged(tg; onlnk(l;m(i;t))) = nil)
. . | 16. i : Id
. . | 17. ∀a:Id, t:ℕ.
. . | |   ∃t':ℕ.
. . | |   t≤t'
. . | |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . | |     ∨ ¬a declared in M(i)
. . | |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . | 18. a : Id
. . | 19. ∀t:ℕ.
. . | |   ∃t':ℕ.
. . | |   t≤t'
. . | |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . | |     ∨ ¬a declared in M(i)
. . | |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . | 20. ℕ
. . | 21. t' : ℕ
. . | 22. ¬a declared in M(i) ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . | 23. a declared in M(i)

. . | ⊢  ¬a declared in M(i) ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . | | by ParallelOp -2
. . | |    THEN
. . | |    Try
. . | |    (Unfold `ma-st` 0 THEN Unfold `ma-state` 0
. . | |     THEN
. . | |     Fold `ma-ds` 0
. . | |     THEN
. . | |     DoSubsume
. . | |     THEN
. . | |     BackThruSomeHyp)
. . | |
. . | |\
```

```
. . | | 1. A : Dsys
. . | | 2. B : Dsys
. . | | 3. A ⊆ B
. . | | 4. w : World
. . | | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. . | | 6. ∀i:Id, a:Action(i).
. . | | |  ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. . | | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. . | | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. . | | 9. ∀i:Id, t:ℕ.
. . | | |  ¬isnull(a(i;t))
. . | | |  ⇒
. . | | |  (islocal(kind(a(i;t)))
. . | | |   ⇒
. . | | |   M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. . | | |  & (∀x:Id.
. . | | |     M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. . | | |     +1).x))
. . | | |  & (∀l:IdLnk.
. . | | |     M(i).send(kind(a(i;t));l;λx.
. . | | |     s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. . | | |  & (∀x:Id.
. . | | |     ¬M(i).frame(kind(a(i;t)) affects x)
. . | | |     ⇒
. . | | |     s(i;t).x = s(i;t+1).x)
. . | | |  & (∀l:IdLnk, tg:Id.
. . | | |     ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. . | | |     ⇒
. . | | |     w-tagged(tg; onlnk(l;m(i;t))) = nil)
. . | | 10. ∀i,a:Id, t:ℕ.
. . | | |   ∃t':ℕ.
. . | | |   t≤t'
. . | | |   & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . | | |     ∨ ¬a declared in M(i)
. . | | |     ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . | | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. . | | 12. ∀i:Id, a:Action(i).
. . | | |   ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. . | | 13. ∀l:IdLnk, tg:Id.
. . | | |   (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. . | | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. . | | 15. ∀i:Id, t:ℕ.
. . | | |   ¬isnull(a(i;t))
. . | | |   ⇒
. . | | |   (islocal(kind(a(i;t)))
. . | | |    ⇒
. . | | |    M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. . | | |   & (∀x:Id.
. . | | |      M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. . | | |      +1).x))
. . | | |   & (∀l:IdLnk.
. . | | |      M(i).send(kind(a(i;t));l;λx.
. . | | |      s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. . | | |   & (∀x:Id.
. . | | |      ¬M(i).frame(kind(a(i;t)) affects x)
```

```
. . | | |       ⇒
. . | | |       s(i;t).x = s(i;t+1).x)
. . | | |    & (∀l:IdLnk, tg:Id.
. . | | |       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. . | | |       ⇒
. . | | |       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. . | | 16. i : Id
. . | | 17. ∀a:Id, t:ℕ.
. . | | |    ∃t':ℕ.
. . | | |    t≤t'
. . | | |    & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . | | |      ∨ ¬a declared in M(i)
. . | | |      ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . | | 18. a : Id
. . | | 19. ∀t:ℕ.
. . | | |    ∃t':ℕ.
. . | | |    t≤t'
. . | | |    & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . | | |      ∨ ¬a declared in M(i)
. . | | |      ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . | | 20. ℕ
. . | | 21. ℕ
. . | | 22. ¬a declared in M(i)
. . | | 23. a declared in M(i)
. . | | ⊢  ¬a declared in M(i)
. . | | | by ParallelOp -2
. . | | |     THEN
. . | | |     Using [‘M1’,M(i)]
. . | | |     (BackThru
. . | | |      Thm* ∀M1,M2:MsgA.
. . | | |           M1 ⊆ M2
. . | | |           ⇒
. . | | |           (∀a:Id. a declared in M1 ⇒ a declared in M2)
. . | | |      {ma_DASH_decla_DASH_sub})
. . | | |
. . | | 1. A : Dsys
. . | | 2. B : Dsys
. . | | 3. A ⊆ B
. . | | 4. w : World
. . | | 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. . | | 6. ∀i:Id, a:Action(i).
. . | |    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. . | | 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. . | | 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. . | | 9. ∀i:Id, t:ℕ.
. . | |    ¬isnull(a(i;t))
. . | |    ⇒
. . | |    (islocal(kind(a(i;t)))
. . | |     ⇒
. . | |     M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. . | |    & (∀x:Id.
. . | |       M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. . | |       +1).x))
. . | |    & (∀l:IdLnk.
. . | |       M(i).send(kind(a(i;t));l;λx.
```

```
. . | |        s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. . | |     & (∀x:Id.
. . | |        ¬M(i).frame(kind(a(i;t)) affects x)
. . | |        ⇒
. . | |        s(i;t).x = s(i;t+1).x)
. . | |     & (∀l:IdLnk, tg:Id.
. . | |        ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. . | |        ⇒
. . | |        w-tagged(tg; onlnk(l;m(i;t))) = nil)
. . | | 10. ∀i,a:Id, t:ℕ.
. . | |      ∃t':ℕ.
. . | |      t≤t'
. . | |      & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . | |        ∨ ¬a declared in M(i)
. . | |        ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . | | 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. . | | 12. ∀i:Id, a:Action(i).
. . | |      ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. . | | 13. ∀l:IdLnk, tg:Id.
. . | |      (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. . | | 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. . | | 15. ∀i:Id, t:ℕ.
. . | |      ¬isnull(a(i;t))
. . | |      ⇒
. . | |      (islocal(kind(a(i;t)))
. . | |       ⇒
. . | |       M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. . | |      & (∀x:Id.
. . | |         M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. . | |         +1).x))
. . | |      & (∀l:IdLnk.
. . | |         M(i).send(kind(a(i;t));l;λx.
. . | |         s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. . | |      & (∀x:Id.
. . | |         ¬M(i).frame(kind(a(i;t)) affects x)
. . | |         ⇒
. . | |         s(i;t).x = s(i;t+1).x)
. . | |      & (∀l:IdLnk, tg:Id.
. . | |         ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. . | |         ⇒
. . | |         w-tagged(tg; onlnk(l;m(i;t))) = nil)
. . | | 16. i : Id
. . | | 17. ∀a:Id, t:ℕ.
. . | |      ∃t':ℕ.
. . | |      t≤t'
. . | |      & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . | |        ∨ ¬a declared in M(i)
. . | |        ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . | | 18. a : Id
. . | | 19. ∀t:ℕ.
. . | |      ∃t':ℕ.
. . | |      t≤t'
. . | |      & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . | |        ∨ ¬a declared in M(i)
. . | |        ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
```

```
. . | | 20. ℕ
. . | | 21. ℕ
. . | | 22. a declared in M(i)
. . | | 23. a declared in M(i)
. . | | ⊢  M(i) ⊆ M(i)
. . | |   by AllHyps (λh.Unfold ‘d-sub‘ h THEN BackThru h)
. . | |
. . | .\
. . | . 1. A : Dsys
. . | . 2. B : Dsys
. . | . 3. A ⊆ B
. . | . 4. w : World
. . | . 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. . | . 6. ∀i:Id, a:Action(i).
. . | .    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. . | . 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. . | . 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. . | . 9. ∀i:Id, t:ℕ.
. . | .    ¬isnull(a(i;t))
. . | .    ⇒
. . | .    (islocal(kind(a(i;t)))
. . | .     ⇒
. . | .     M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. . | .    & (∀x:Id.
. . | .       M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. . | .       +1).x))
. . | .    & (∀l:IdLnk.
. . | .       M(i).send(kind(a(i;t));l;λx.
. . | .       s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. . | .    & (∀x:Id.
. . | .       ¬M(i).frame(kind(a(i;t)) affects x)
. . | .       ⇒
. . | .       s(i;t).x = s(i;t+1).x)
. . | .    & (∀l:IdLnk, tg:Id.
. . | .       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. . | .       ⇒
. . | .       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. . | . 10. ∀i,a:Id, t:ℕ.
. . | .     ∃t':ℕ.
. . | .     t≤t'
. . | .     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . | .       ∨ ¬a declared in M(i)
. . | .       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . | . 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. . | . 12. ∀i:Id, a:Action(i).
. . | .     ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. . | . 13. ∀l:IdLnk, tg:Id.
. . | .     (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. . | . 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. . | . 15. ∀i:Id, t:ℕ.
. . | .     ¬isnull(a(i;t))
. . | .     ⇒
. . | .     (islocal(kind(a(i;t)))
. . | .      ⇒
. . | .      M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
```

```
. . | .     & (∀x:Id.
. . | .        M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. . | .        +1).x))
. . | .     & (∀l:IdLnk.
. . | .        M(i).send(kind(a(i;t));l;λx.
. . | .        s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. . | .     & (∀x:Id.
. . | .        ¬M(i).frame(kind(a(i;t)) affects x)
. . | .        ⇒
. . | .        s(i;t).x = s(i;t+1).x)
. . | .     & (∀l:IdLnk, tg:Id.
. . | .        ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. . | .        ⇒
. . | .        w-tagged(tg; onlnk(l;m(i;t))) = nil)
. . | . 16. i : Id
. . | . 17. ∀a:Id, t:ℕ.
. . | .     ∃t':ℕ.
. . | .     t≤t'
. . | .     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . | .       ∨ ¬a declared in M(i)
. . | .       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . | . 18. a : Id
. . | . 19. ∀t:ℕ.
. . | .     ∃t':ℕ.
. . | .     t≤t'
. . | .     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . | .       ∨ ¬a declared in M(i)
. . | .       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . | . 20. ℕ
. . | . 21. t' : ℕ
. . | . 22. unsolvable M(i).pre(a,λx.s(i;t').x)
. . | . 23. a declared in M(i)
. . | . ⊢  unsolvable M(i).pre(a,λx.s(i;t').x)
. . | .   by Using ['M2',M(i)]
. . | .      (BackThru
. . | .       Thm* ∀M1,M2:MsgA.
. . | .             M1 ⊆ M2
. . | .             ⇒
. . | .             (∀a:Id, s:M2.state.
. . | .              a declared in M1
. . | .              ⇒
. . | .              unsolvable M2.pre(a,s) ⇒ unsolvable M1.pre(a,s))
. . | .       {ma_DASH_npre_DASH_sub})
. . | .      THEN
. . | .      Try
. . | .      (Unfold 'ma-st' 0 THEN Unfold 'ma-state' 0
. . | .       THEN
. . | .       Fold 'ma-ds' 0
. . | .       THEN
. . | .       DoSubsume
. . | .       THEN
. . | .       BackThruSomeHyp)
. . | .      THEN
. . | .      AllHyps (λh.Unfold 'd-sub' h THEN BackThru h)
. . |
```

```
. . .\
. . . 1. A : Dsys
. . . 2. B : Dsys
. . . 3. A ⊆ B
. . . 4. w : World
. . . 5. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. . . 6. ∀i:Id, a:Action(i).
. . .    ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. . . 7. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. . . 8. ∀i,x:Id. M(i).init(x,s(i;0).x)
. . . 9. ∀i:Id, t:ℕ.
. . .    ¬isnull(a(i;t))

. . .    ⇒
. . .    (islocal(kind(a(i;t)))
. . .     ⇒
. . .     M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. . .    & (∀x:Id.
. . .       M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. . .       +1).x))
. . .    & (∀l:IdLnk.
. . .       M(i).send(kind(a(i;t));l;λx.
. . .       s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. . .    & (∀x:Id.
. . .       ¬M(i).frame(kind(a(i;t)) affects x)
. . .       ⇒
. . .       s(i;t).x = s(i;t+1).x)
. . .    & (∀l:IdLnk, tg:Id.
. . .       ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. . .       ⇒
. . .       w-tagged(tg; onlnk(l;m(i;t))) = nil)
. . . 10. ∀i,a:Id, t:ℕ.
. . .     ∃t':ℕ.
. . .     t≤t'
. . .     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . .       ∨ ¬a declared in M(i)
. . .       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . . 11. ∀i,x:Id. vartype(i;x) ⊆r M(i).ds(x)
. . . 12. ∀i:Id, a:Action(i).
. . .     ¬isnull(a) ⇒ (valtype(i;a) ⊆r M(i).da(kind(a)))
. . . 13. ∀l:IdLnk, tg:Id. (w.M(l,tg)) ⊆r M(source(l)).da(rcv(l; tg))
. . . 14. ∀i,x:Id. M(i).init(x,s(i;0).x)
. . . 15. ∀i:Id, t:ℕ.
. . .     ¬isnull(a(i;t))
. . .     ⇒
. . .     (islocal(kind(a(i;t)))
. . .      ⇒
. . .      M(i).pre(act(kind(a(i;t))),λx.s(i;t).x,val(a(i;t))))
. . .     & (∀x:Id.
. . .        M(i).ef(kind(a(i;t)),x,λx.s(i;t).x,val(a(i;t)),s(i;t
. . .        +1).x))
. . .     & (∀l:IdLnk.
. . .        M(i).send(kind(a(i;t));l;λx.
. . .        s(i;t).x;val(a(i;t));withlnk(l;m(i;t));i))
. . .     & (∀x:Id.
. . .        ¬M(i).frame(kind(a(i;t)) affects x)
```

```
. . .         ⇒
. . .         s(i;t).x = s(i;t+1).x)
. . .     & (∀l:IdLnk, tg:Id.
. . .         ¬M(i).sframe(kind(a(i;t)) sends <l,tg>)
. . .         ⇒
. . .         w-tagged(tg; onlnk(l;m(i;t))) = nil)
. . . 16. i : Id
. . . 17. ∀a:Id, t:ℕ.
. . .     ∃t':ℕ.
. . .     t≤t'
. . .     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . .       ∨ ¬a declared in M(i)
. . .       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . . 18. a : Id
. . . 19. ∀t:ℕ.
. . .     ∃t':ℕ.
. . .     t≤t'
. . .     & ¬isnull(a(i;t')) & kind(a(i;t')) = locl(a)
. . .       ∨ ¬a declared in M(i)
. . .       ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . . 20. ℕ
. . . 21. t' : ℕ
. . . 22. ¬a declared in M(i) ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . . 23. ¬a declared in M(i)
. . . ⊢  ¬a declared in M(i) ∨ unsolvable M(i).pre(a,λx.s(i;t').x)
. . .   by OrLeft
. . .      THEN
. . .      Try
. . .      (Unfold `ma-st` 0 THEN Unfold `ma-state` 0
. . .       THEN
. . .       Fold `ma-ds` 0
. . .       THEN
. . .       DoSubsume
. . .       THEN
. . .       BackThruSomeHyp)
END
```