```
term_subst_tc  THM

⊢  ∀t:Term, s:SimpleType, as:(Label×Term) List, ds:Collection(dec())
|  , da:Collection(SimpleType), de:sig().
|  s ∈ term_types(ds;da;de;t) ⇒
|  (∀x:Label.
|   (x ∈ term_vars(t)) ⇒
|   (∀t:SimpleType.
|    mk_dec(x, t) ∈ ds ⇒
|    t ∈ term_types(ds;da;de;apply_alist(as;x;x))))
|  ⇒
|  s ∈ term_types(ds;da;de;term_subst(as;t))
| by D 0 THENA Auto THEN TermInd (-1)
THEN Unfold `term_subst` 0

|    THEN Reduce 0
THEN Try (Fold `term_subst` 0)

|    THEN Try (Complete Auto)
THEN Try (Auto THEN

|         BackThruSomeHyp THEN

|         Try (RWW "member_singleton" 0 THEN Auto)

|         THEN AllHyps (RWW "member_dec_lookup") THEN Auto)
|
|\
| 1. Term
| 2. u : Term→Type
| 3. t:{v:Term| u(v) }→∀s:SimpleType, as:(Label×Term) List
| |  , ds:Collection(dec()), da:Collection(SimpleType), de:sig().
| |  s ∈ term_types(ds;da;de;t) ⇒
| |  (∀x:Label.
| |   (x ∈ term_vars(t)) ⇒
| |   (∀t:SimpleType.
| |    mk_dec(x, t) ∈ ds ⇒
| |    t ∈ term_types(ds;da;de;apply_alist(as;x;x))))
| |  ⇒
| |  s ∈ term_types(ds;da;de;term_subst(as;t))
| 4. x : Label
| ⊢  ∀s:SimpleType, as:(Label×Term) List, ds:Collection(dec())
| |  , da:Collection(SimpleType), de:sig().
| |  s ∈ dec_lookup(ds;x) ⇒
| |  (∀x@0:Label.
| |   (x@0 ∈ [x]) ⇒
| |   (∀t:SimpleType.
| |    mk_dec(x@0, t) ∈ ds ⇒
| |    t ∈ term_types(ds;da;de;apply_alist(as;x@0;x@0))))
| |  ⇒
| |  s ∈ term_types(ds;da;de;apply_alist(as;x;x'))
| | by Auto
| THEN Subst
| |    term_types(ds;da;de;apply_alist(as;x;x')) ∼ ...0
| |
```

```
| |\
| | .....equality .....
| | 1. Term
| | 2. u : Term→Type
| | 3. t:{v:Term| u(v) }→∀s:SimpleType, as:(Label×Term) List
| |    , ds:Collection(dec()), da:Collection(SimpleType), de:sig().
| |    s ∈ term_types(ds;da;de;t) ⇒
| |    (∀x:Label.
| |     (x ∈ term_vars(t)) ⇒
| |     (∀t:SimpleType.
| |      mk_dec(x, t) ∈ ds ⇒
| |      t ∈ term_types(ds;da;de;apply_alist(as;x;x))))
| |    ⇒
| |    s ∈ term_types(ds;da;de;term_subst(as;t))
| | 4. x : Label
| | 5. s : SimpleType
| | 6. as : (Label×Term) List
| | 7. ds : Collection(dec())
| | 8. da : Collection(SimpleType)
| | 9. de : sig()
| | 10. s ∈ dec_lookup(ds;x)
| | 11. ∀x@0:Label.
| |     (x@0 ∈ [x]) ⇒
| |     (∀t:SimpleType.
| |      mk_dec(x@0, t) ∈ ds ⇒
| |      t ∈ term_types(ds;da;de;apply_alist(as;x@0;x@0)))
| | ⊢  term_types(ds;da;de;apply_alist(as;x;x')) ∼ ...
| |   by AllHyps Thin THEN
| |
| |      Unfold `apply_alist` 0 THEN Unfold `find` 0
| |
| |      THEN ListInd 2 THEN Reduce 0
| | THEN Try Trivial
| |
| |      THEN SplitOnConclITE
| | THENA Auto
| | %THEN Unfold `let` 0%
| |
| |      THEN Reduce 0
| | THEN Try Trivial
| |
| |
| .\
| . 1. Term
| . 2. u : Term→Type
| . 3. t:{v:Term| u(v) }→∀s:SimpleType, as:(Label×Term) List
| .    , ds:Collection(dec()), da:Collection(SimpleType), de:sig().
| .    s ∈ term_types(ds;da;de;t) ⇒
| .    (∀x:Label.
| .     (x ∈ term_vars(t)) ⇒
| .     (∀t:SimpleType.
| .      mk_dec(x, t) ∈ ds ⇒
| .      t ∈ term_types(ds;da;de;apply_alist(as;x;x))))
| .    ⇒
| .    s ∈ term_types(ds;da;de;term_subst(as;t))
```

```
| . 4. x : Label
| . 5. s : SimpleType
| . 6. as : (Label×Term) List
| . 7. ds : Collection(dec())
| . 8. da : Collection(SimpleType)
| . 9. de : sig()
| . 10. s ∈ dec_lookup(ds;x)
| . 11. ∀x@0:Label.
| .     (x@0 ∈ [x]) ⇒
| .     (∀t:SimpleType.
| .      mk_dec(x@0, t) ∈ ds ⇒
| .      t ∈ term_types(ds;da;de;apply_alist(as;x@0;x@0)))
| . ⊢  s ∈ term_types(ds;da;de;apply_alist(as;x;x))
| .   by BackThruSomeHyp THEN
| .

| .           Try (RWW "member_singleton" 0 THEN Auto)
| .
| .           THEN AllHyps (RWW "member_dec_lookup") THEN Auto
|
.\
. 1. Term
. 2. u : Term→Type
. 3. t:{v:Term| u(v) }→∀s:SimpleType, as:(Label×Term) List
. |  , ds:Collection(dec()), da:Collection(SimpleType), de:sig().
. |  s ∈ term_types(ds;da;de;t) ⇒
. |  (∀x:Label.
. |   (x ∈ term_vars(t)) ⇒
. |   (∀t:SimpleType.
. |    mk_dec(x, t) ∈ ds ⇒
. |    t ∈ term_types(ds;da;de;apply_alist(as;x;x))))
. |  ⇒
. |  s ∈ term_types(ds;da;de;term_subst(as;t))
. 4. y1 : {v:Term| u(v) }
. 5. y2 : {v:Term| u(v) }
. ⊢  ∀s:SimpleType, as:(Label×Term) List, ds:Collection(dec())
. |  , da:Collection(SimpleType), de:sig().
. |  s ∈ st_app(term_types(ds;da;de;y1);term_types(ds;da;de;y2)) ⇒
. |  (∀x:Label.
. |   (x ∈ term_vars(y1) @ term_vars(y2)) ⇒
. |   (∀t:SimpleType.
. |    mk_dec(x, t) ∈ ds ⇒
. |    t ∈ term_types(ds;da;de;apply_alist(as;x;x))))
. |  ⇒
. |
. |  s ∈ st_app(term_types(ds;da;de;term_subst(as;y1));...)
. | by Unfold ‘term_types‘ 0 THEN Reduce 0
. THEN Fold ‘term_types‘ 0
. |
. THEN RWW "member_st_app" 0
. THEN Auto
. |
. 1. Term
. 2. u : Term→Type
. 3. t:{v:Term| u(v) }→∀s:SimpleType, as:(Label×Term) List
.    , ds:Collection(dec()), da:Collection(SimpleType), de:sig().
```

```
.    s ∈ term_types(ds;da;de;t) ⇒
.    (∀x:Label.
.     (x ∈ term_vars(t)) ⇒
.     (∀t:SimpleType.
.      mk_dec(x, t) ∈ ds ⇒
.      t ∈ term_types(ds;da;de;apply_alist(as;x;x))))
.    ⇒
.    s ∈ term_types(ds;da;de;term_subst(as;t))
. 4. y1 : {v:Term| u(v) }
. 5. y2 : {v:Term| u(v) }
. 6. s : SimpleType
. 7. as : (Label×Term) List
. 8. ds : Collection(dec())
. 9. da : Collection(SimpleType)
. 10. de : sig()
. 11. ∃a:SimpleType.
.     a ∈ term_types(ds;da;de;y2) & a→s ∈ term_types(ds;da;de;y1)
. 12. ∀x:Label.
.     (x ∈ term_vars(y1) @ term_vars(y2)) ⇒
.     (∀t:SimpleType.
.      mk_dec(x, t) ∈ ds ⇒
.      t ∈ term_types(ds;da;de;apply_alist(as;x;x)))
. ⊢  ∃a:SimpleType.
.    a ∈ term_types(ds;da;de;term_subst(as;y2))
.    & a→s ∈ term_types(ds;da;de;term_subst(as;y1))
.   by ExRepD THEN InstConcl [a] THEN Auto
.
.      THEN Repeat(BackThruSomeHyp THEN Auto)
.
.      THEN RWW "member_append" 0 THENM TrivialOr
. THEN Auto
END
```