```
trivial_term_subst  THM

⊢  ∀t:Term, as:(Label×Term) List.
|  (∀x:Label. unprime(apply_alist(as;x;x)) = x) ⇒
|  unprime(term_subst(as;t)) = unprime(t)
| by D 0 THENA Auto THEN

|    TermInd 1 THEN Unfold ‘term_subst‘ 0 THEN Reduce 0

|    THEN Try (Fold ‘term_subst‘ 0)
THEN %Sm%Auto

|    THEN Try (RW (AddrC [3] (UnfoldC ‘unprime‘)) 0 THEN Reduce 0

|              THEN Complete SmAuto)

|    THEN Try (Unfold ‘unprime‘ 0 THEN Reduce 0 THEN Fold ‘unprime‘ 0
|
|     THEN EqCD THEN SmAuto)
|
|
1. Term
2. u : Term→Type
3. t:{v:Term| u(v) }→∀as:(Label×Term) List.
|  (∀x:Label. unprime(apply_alist(as;x;x)) = x) ⇒
|  unprime(term_subst(as;t)) = unprime(t)
4. x : Label
5. as : (Label×Term) List
6. ∀x:Label. unprime(apply_alist(as;x;x)) = x
⊢  unprime(apply_alist(as;x;x’)) = unprime(x’)
| by InstHyp [x] (-1) THENA Auto
|  THEN Assert (x ∈ 1of(unzip(as)))
|    ∨ ¬(x ∈ 1of(unzip(as))) ·
|
|\
| .....assertion .....
| 1. Term
| 2. u : Term→Type
| 3. t:{v:Term| u(v) }→∀as:(Label×Term) List.
|    (∀x:Label. unprime(apply_alist(as;x;x)) = x) ⇒
|    unprime(term_subst(as;t)) = unprime(t)
| 4. x : Label
| 5. as : (Label×Term) List
| 6. ∀x:Label. unprime(apply_alist(as;x;x)) = x
| 7. unprime(apply_alist(as;x;x)) = x
| ⊢  (x ∈ 1of(unzip(as))) ∨ ¬(x ∈ 1of(unzip(as)))
|   by Fold ‘decidable‘ 0 THEN BackThruLemma ‘l_member_decidable‘
|
|         THEN Auto THEN Easy
|
|
.\
. 1. Term
. 2. u : Term→Type
. 3. t:{v:Term| u(v) }→∀as:(Label×Term) List.
```

```
. |  (∀x:Label. unprime(apply_alist(as;x;x)) = x) ⇒
. |  unprime(term_subst(as;t)) = unprime(t)
. 4. x : Label
. 5. as : (Label×Term) List
. 6. ∀x:Label. unprime(apply_alist(as;x;x)) = x
. 7. unprime(apply_alist(as;x;x)) = x
. 8. (x ∈ 1of(unzip(as))) ∨ ¬(x ∈ 1of(unzip(as)))
. ⊢  unprime(apply_alist(as;x;x')) = unprime(x')  by D (-1)
. |
. |\
. | 1. Term
. | 2. u : Term→Type
. | 3. t:{v:Term| u(v) }→∀as:(Label×Term) List.
. | |  (∀x:Label. unprime(apply_alist(as;x;x)) = x) ⇒
. | |  unprime(term_subst(as;t)) = unprime(t)
. | 4. x : Label
. | 5. as : (Label×Term) List
. | 6. ∀x:Label. unprime(apply_alist(as;x;x)) = x
. | 7. unprime(apply_alist(as;x;x)) = x
. | 8. (x ∈ 1of(unzip(as)))
. | ⊢  unprime(apply_alist(as;x;x')) = unprime(x')
. | | by Subst apply_alist(as;x;x') = apply_alist(as;x;x) 0 THEN Auto
. | |
. | |\
. | | .....equality .....
. | | 1. Term
. | | 2. u : Term→Type
. | | 3. t:{v:Term| u(v) }→∀as:(Label×Term) List.
. | |    (∀x:Label. unprime(apply_alist(as;x;x)) = x) ⇒
. | |    unprime(term_subst(as;t)) = unprime(t)
. | | 4. x : Label
. | | 5. as : (Label×Term) List
. | | 6. ∀x:Label. unprime(apply_alist(as;x;x)) = x
. | | 7. unprime(apply_alist(as;x;x)) = x
. | | 8. (x ∈ 1of(unzip(as)))
. | | ⊢  apply_alist(as;x;x') = apply_alist(as;x;x)  by Easy
. | |
. | .\
. | . 1. Term
. | . 2. u : Term→Type
. | . 3. t:{v:Term| u(v) }→∀as:(Label×Term) List.
. | .    (∀x:Label. unprime(apply_alist(as;x;x)) = x) ⇒
. | .    unprime(term_subst(as;t)) = unprime(t)
. | . 4. x : Label
. | . 5. as : (Label×Term) List
. | . 6. ∀x:Label. unprime(apply_alist(as;x;x)) = x
. | . 7. unprime(apply_alist(as;x;x)) = x
. | . 8. (x ∈ 1of(unzip(as)))
. | . ⊢  unprime(apply_alist(as;x;x)) = unprime(x')
. | .   by ...
. |
. .\
. . 1. Term
. . 2. u : Term→Type
. . 3. t:{v:Term| u(v) }→∀as:(Label×Term) List.
```

```
. .    (∀x:Label. unprime(apply_alist(as;x;x)) = x) ⇒
. .    unprime(term_subst(as;t)) = unprime(t)
. . 4. x : Label
. . 5. as : (Label×Term) List
. . 6. ∀x:Label. unprime(apply_alist(as;x;x)) = x
. . 7. unprime(apply_alist(as;x;x)) = x

. . 8. ¬(x ∈ 1of(unzip(as)))
. . ⊢  unprime(apply_alist(as;x;x')) = unprime(x')
. .   by RWW "apply_alist_non_member" 0 THEN Auto
END
```