```
assert_st_eq  THM

⊢  ∀s1,s2:SimpleType. st_eq(s1;s2) ⟺ s1 = s2  by Auto
|
|\
| 1. s1 : SimpleType
| 2. s2 : SimpleType
| 3. st_eq(s1;s2)
| ⊢  s1 = s2  by MoveToConcl 2 THEN MoveToConcl 1
| |
| ⊢  ∀s1,s2:SimpleType. st_eq(s1;s2) ⇒ s1 = s2
| | by RepeatFor 2 (
| D 0 THENA Auto THEN Unfold ‘st‘ (-1)
| ...
| ...
| |
| |\
| | 1. SimpleType
| | 2. u : SimpleType→Type
| | 3. s1:{v:SimpleType| u(v) }→∀s2:SimpleType.
| | |  st_eq(s1;s2) ⇒ s1 = s2
| | 4. x : Label+Unit
| | 5. SimpleType
| | 6. u1 : SimpleType→Type
| | 7. s2:{v:SimpleType| u1(v) }→st_eq(tree_leaf(x);s2) ⇒
| | |  tree_leaf(x) = s2
| | 8. x1 : Label+Unit
| | 9. InjCase(x; x'. InjCase(x1; y'. x' =_1 y'; b. false_2); a.
| | |  InjCase(x1; y'. false_2; b. true_2))
| | ⊢  tree_leaf(x) = tree_leaf(x1)
| | | by D 8 THEN D 4 THEN All Reduce THEN
| | Try Trivial THEN
| | ...
| | |
| | |\
| | | 1. SimpleType
| | | 2. u : SimpleType→Type
| | | 3. s1:{v:SimpleType| u(v) }→∀s2:SimpleType.
| | |    st_eq(s1;s2) ⇒ s1 = s2
| | | 4. x3 : Label
| | | 5. SimpleType
| | | 6. u1 : SimpleType→Type
| | | 7. s2:{v:SimpleType| u1(v) }→st_eq(tree_leaf(inl(x3));s2) ⇒
| | |    tree_leaf(inl(x3)) = s2
| | | 8. x2 : Label
| | | 9. x3 =_1 x2
| | | ⊢  x3 = x2  by EqLblFwd THEN HypSubst (-1) 0 THEN Auto
| | |
| | .\
| | . 1. SimpleType
| | . 2. u : SimpleType→Type
| | . 3. s1:{v:SimpleType| u(v) }→∀s2:SimpleType.
| | .    st_eq(s1;s2) ⇒ s1 = s2
| | . 4. y1 : Unit
| | . 5. SimpleType
```

```
| | . 6. u1 : SimpleType→Type
| | . 7. s2:{v:SimpleType| u1(v) }→st_eq(tree_leaf(inr(y1));s2) ⇒
| | .    tree_leaf(inr(y1)) = s2
| | . 8. y : Unit
| | . 9. True
| | . ⊢  tree_leaf(inr(y1)) = tree_leaf(inr(y))
| | .   by D 8 THEN D 4
| | . THEN Fold ‘it‘ 0 THEN Fold ‘st_top‘ 0
| | . ...
| |
| .\
| . 1. SimpleType
| . 2. u : SimpleType→Type
| . 3. s1:{v:SimpleType| u(v) }→∀s2:SimpleType.
| .    st_eq(s1;s2) ⇒ s1 = s2
| . 4. y1 : {v:SimpleType| u(v) }
| . 5. y2 : {v:SimpleType| u(v) }
| . 6. SimpleType
| . 7. u1 : SimpleType→Type
| . 8. s2:{v:SimpleType| u1(v) }→st_eq(tree_node(<y1, y2>);s2) ⇒
| .    tree_node(<y1, y2>) = s2
| . 9. y3 : {v:SimpleType| u1(v) }
| . 10. y4 : {v:SimpleType| u1(v) }
| . 11. st_eq(y1;y3)∧2st_eq(y2;y4)
| . ⊢  tree_node(<y1, y2>) = tree_node(<y3, y4>)
| .   by ...
| . THEN EqCD  THEN All (Fold ‘st‘) THEN SmAuto
|
.\
. 1. s1 : SimpleType
. 2. s2 : SimpleType
. 3. s1 = s2
. ⊢  st_eq(s1;s2)  by HypSubst (-1) 0 THENA Auto
. |
. ...
END
```