```
trace_consistent_pred_addprime  THM

⊢  ∀da:Collection(dec()), P:Fmla, rho:Decl, te:(Label→Label→𝔹).
|  trace_consistent_pred(rho;da;te;P)
|  ⇒
|  trace_consistent_pred(rho;da;te;(P)')
| by Auto THEN RepeatFor 2 (ParallelOp (-1))

|    THEN Unfold `pred_addprime` 0 THEN RW ColMemberC 0 THEN Auto

|    THEN Try (Fold `pred` 0 THEN Auto)
THEN ExRepD THEN SubstFor r
|     0 THEN AllHyps (InstHyp [r@0]) THEN Auto
THEN Lemmaize [(-1)]
|
⊢  ∀da:Collection(dec()), rho:Decl, te:(Label→Label→𝔹), r@0:rel().
|  trace_consistent_rel(rho;da;te;r@0)
|  ⇒
|  trace_consistent_rel(rho;da;te;(r@0)')
| by Auto THEN Unfold `rel_addprime` 0 THEN ParallelOp (-1)

|    THEN Reduce 0 THEN RWO "map_length" 0 THENA Auto

|    THEN ParallelOp (-1)
THEN Subst' map(λt.(t)';r@0.args)[i]
|                                      =
|                                      (r@0.args[i])' 0

|    THEN Try (RWO "map_length" 0 THEN Auto)

|    THEN Try (RWO "map_select" 0 THEN Auto THEN Reduce 0 THEN Auto)

|    THEN MoveToConcl (-1)
THEN GenConcl r@0.args[i] = t THENA Auto

|    THEN Lemmaize []
|
⊢  ∀da:Collection(dec()), rho:Decl, te:(Label→Label→𝔹), t:Term.
|  trace_consistent(rho;da;te;t) ⇒ trace_consistent(rho;da;te;(t)')
| by Auto THEN RepeatFor 3 (ParallelOp (-1)) THEN Lemmaize [(-1)]
|
⊢  ∀t:Term, g:Label.
   term_mentions_guard(g;(t)') ⇒ term_mentions_guard(g;t)
  by Auto THEN MoveToConcl (-1) THEN Unfold `addprime` 0 THEN

     TermInd 1 THEN All Reduce THEN Auto

     THEN AllHyps (Fold `addprime`)

     THEN All (RW assert_pushdownC) THENA Auto

     THEN ParallelOp (-1) THEN EasyHyp
END
```