```
rel_arg_typing  THM

⊢  ∀r:rel(), i:ℕ, ds,da:Collection(dec()), st1:Collection(SimpleType)
|  , de:sig(), rho:Decl, s:{[[ds]] rho}, e:{1of([[de]] rho)}
|  , a:[[st1]] rho, tr:trace_env([[da]] rho).
|  trace_consistent_rel(rho;da;tr.proj;r)
|  ⇒
|  tc(r;ds;st1;de)
|  ⇒
|  i<||r.args||
|  ⇒
|  [[r.args[i]]] e s a tr ∈ [[rel_arg_typ(r.name;i;de)]] rho
| by Unfolds ``tc rel_arg_typ`` 0
THEN D 0 THENA Auto

|    THEN D 1 THEN D 1 THEN Reduce 0
THEN Auto'
THEN Try (
|    ...
THEN ExRepD
THEN Using [`de',de
|    ] (BackThruLemma `term_typing`)
THEN SmAuto
THEN
|    AllHyps (\h. Unfold `trace_consistent_rel` h THEN Reduce h THEN

|                InstHyp [i] h THEN Complete Auto)
|
1. x : SimpleType
2. r1 : Term List
3. i : ℕ
4. ds : Collection(dec())
5. da : Collection(dec())
6. st1 : Collection(SimpleType)
7. de : sig()
8. rho : Decl
9. {[[ds]] rho}
10. {1of([[de]] rho)}
11. [[st1]] rho
12. tr : trace_env([[da]] rho)
13. trace_consistent_rel(rho;da;tr.proj;<inl(x),r1>)
14. ||r1|| = 2
15. x ∈ term_types(ds;st1;de;r1[0])
16. x ∈ term_types(ds;st1;de;r1[1])
17. i<||r1||
⊢  x ∈ term_types(ds;st1;de;r1[i])
  by CaseNat 0 `i' THEN Try Trivial

    THEN CaseNat 1 `i' THEN Try Trivial
THEN Auto'·
END
```