```
(FILECREATED " 6-Jun-79 15:36:27" <AFFIRM>PROOFSTRUCTURE.;2 13873

     changes to:  CheckForPropositionStatus RootOf Roots SubProofNumbers1

     previous date: "15-May-79 21:18:12" <AFFIRM>PROOFSTRUCTURE.;1)


(PRETTYCOMPRINT PROOFSTRUCTURECOMS)

(RPAQQ PROOFSTRUCTURECOMS ((FNS * PROOFSTRUCTUREFNS)))

(RPAQQ PROOFSTRUCTUREFNS (AddPredicate CheckAndSplit CheckForPropositionStatus CheckForSuccess FetchLog
                                       GetPredicate GetProof Log PrintBoth PrintBoth1 PrintProof PrintProofs
                                       RootOf Roots SubProofNumbers SubProofNumbers1 SubProofs UpdateProof
                                       UpdateProofList WhatRoots leaves nextis retrycaller))
(DEFINEQ

(AddPredicate
  [LAMBDA (p status)
    (CLISP: UNDOABLE)                                          (* R.Erickson "19-DEC-78 15:43")
    (PROG (n m)
          (n←(SASSOC p TheoremList:PredicateList)::1)
          (if n
              then (if n=CurrentTheoremNumber
                       then (RETURN n)
                     else (TERPRI)
                          (PRIN1 n)
                          (PRIN1 " (")
                          (PRIN1 (GetProof n TheoremList):2)
                          (PRIN1 ")")
                          (TERPRI)
                          (RETURN n))
            else n← TheoremList:NumberPreds + 1 (PRINTLINES T n "(new)" T)
                 (/SET 'TheoremList (create TheoremList
                                            NumberPreds ← n
                                            PredicateList ←(<<p ! n> ! TheoremList:PredicateList >)
                                            ProofList ←(<!! TheoremList:ProofList (if (ATOM status)
                                                                                      then
                                                                                       <n status
                                                                                         <'# >>
                                                                                    else
                                                                                     <n ! status
                                                                                       <'# >>)

                                                      >)))
                 (RETURN n])

(CheckAndSplit
  [LAMBDA (qex subs)

          (* given a parent Qexpression and a list of (non-qexpr) subgoals, check if their find sets are disjoint.
          If so, set up the subgoals.)


    (PROG (qfind qgiven finds sofar common)
          (qfind←(for f in qex:find collect f:1))              (* extract var. names)
          (qgiven←(for f in qex:given collect f:1))
          (finds←(for s in subs collect (INTERSECTION (FreeVars s)
                                                      qfind)))
          (sofar←NIL)
          (common←(for f in finds bind dup eachtime (PROGN dup←(INTERSECTION sofar f)
                                                           sofar← <!! sofar ! f>)
                     when dup collect dup))
          (if common
              then (TERPRI)
                   (PRIN1 "Unable to split; these 'find' variables are used in >1 subgoal:")
                   (PRINT common)
                   (TERPRI)
            else (Subgoal)
                 (UpdateProof CurrentTheoremNumber (for s in subs as f in finds
                                                      collect (AddPredicate (create Qexpression
                                                                                    expr ← s using qex)
                                                                            '(new proposition)))

                              TheoremList:ProofList])
```

```
(CheckForPropositionStatus
  [LAMBDA (n)                                                    (* D.Musser " 4-May-79 13:50")
    (if (MEMB (GetProof n TheoremList):2 '(untried new assumed pending proved basis))
      else (PRINTLINES T "item" n "is not a proposition" T)
           (ERROR!))

(CheckForSuccess
  [LAMBDA NIL                                                    (* D.Musser "18-OCT-78 11:16")
    (PROG (n)
          (n←(SASSOC CurrentTheorem TheoremList:PredicateList)::1)
          (if AfterNorm=TRUE
              then (if (UpdateProof n (REVERSE AssumptionsUsed)
                                    TheoremList:ProofList)='proved
                       then (UpdateProofList TheoremList:ProofList <n>))
            else (TERPRI)
                 (PRIN1 "(CurrentGoal is Proposition ")
                 (PRIN1 n)
                 (PRIN1 ")")
                 (TERPRI)))

(FetchLog
  [LAMBDA (tno)                                                  (* Edited by Erickson on 16-AUG-78;
                                                                 no file)
                                                                 (* We want the log, which replaces the
                                                                 (#). Ignore the first element, a dummy)
    (CDR (for r in (ASSOC tno TheoremList:ProofList) thereis (LISTP r)))

(GetPredicate
  [LAMBDA (n TL)
    (if ~(MINUSP n) and n leq TL:NumberPreds
        then (NTH TL:PredicateList TL:NumberPreds-n+1)::1:1
      else (PRINTLINES T n "?" T)
           (ERROR!))

(GetProof
  [LAMBDA (n TL)
    (if ~(MINUSP n) and n leq TL:NumberPreds
        then (ASSOC n TL:ProofList)
      else (PRINTLINES T n "?" T)
           (ERROR!))

(Log
  [LAMBDA (command data)
    (CLISP: UNDOABLE)                                            (* R.Erickson "19-DEC-78 15:52")
    (/SET 'RetryLog <!! RetryLog <command data>>])

(PrintBoth
  [LAMBDA (which)                                                (* edited: "18-SEP-78 11:01")
                                                                 (* print out the proof tree + predicates for
                                                                 selected roots)
    (PROG (roots)
          (roots←(WhatRoots which))
          (for p in roots do (PrintBoth1 (ASSOC p TheoremList:ProofList)
                                         0))

(PrintBoth1
  [LAMBDA (pf1 depth)                                            (* edited: "18-SEP-78 10:27")
                                                                 (* print out predicates + proof structure for
                                                                 this proof list)
    (TERPRI)
    (TAB depth*INDENTATION)
    (for p on pf1 while (NLISTP p:1) do (PRINTLINES p:1 " ") finally (PrettyPrint (GetPredicate pf1:1
                                                                                    TheoremList)
                                                                                  T)
                                                                     (for y in p::1 do (PrintBoth1 y depth+1])

(PrintProof
  [LAMBDA (list depth)                                           (* Edited by Erickson on 16-AUG-78;
                                                                 from version 7)
    (TAB depth*INDENTATION)
    (for x on list while (NLISTP x:1) as i from 1 do (PRINTLINES x:1 (if i=1
                                                                        then " "))
```

```
      finally (for y in x::1 do (PrintProof y depth+1))

(PrintProofs
  [LAMBDA (which)                                          (* edited: "18-SEP-78 10:07")
    (Heading "Proof status:")
    (for x in (WhatRoots which) do (PrintProof (ASSOC x TheoremList:ProofList)
                                               0)
                                   (TERPRI))

(RootOf
  [LAMBDA (n)                                              (* R.Erickson "11-May-79 15:05")
    (PROG ((root n)
           pfl scan)
          (pfl←TheoremList:ProofList)
          (while pfl do

          (* We keep seeking a parent of root, who is the best guess so far. When we find one, we start scanning over
          again.)


                    (scan←pfl:1)
                    (pfl←pfl::1)
                    (for sub in (SubProofs scan) when sub:1=root do
                                                           (* sub and scan are prooflist entries:
                                                           (n --))
                       (root←scan:1)
                       (pfl←TheoremList:ProofList)
                                                           (* start over)))
          (RETURN root]))

(Roots
  [LAMBDA NIL                                              (* R.Erickson "11-May-79 15:24")
                                                           (* return a list of the numbers of all
                                                           predicates which are roots in the dependency
                                                           forest)
    (PROG (candidates)
          (candidates←(for i to TheoremList:NumberPreds when (ASSOC i TheoremList:ProofList) collect i))
          [for pf in TheoremList:ProofList do (for sub in (SubProofs pf)
                                                  do       (* sub= (n --))
                                                     (if sub:1 MEMB candidates
                                                        then candidates←(DREMOVE sub:1 candidates]
          (RETURN candidates]))

(SubProofNumbers
  [LAMBDA (n)                                              (* edited: "18-SEP-78 16:15")
                                                           (* given a propn #, return all lower thm #s)
    (SubProofNumbers1 (SubProofs (ASSOC n TheoremList:ProofList]))

(SubProofNumbers1
  [LAMBDA (pflist)                                         (* R.Erickson "11-May-79 15:06")
                                                           (* given a list of ProofList entries, return
                                                           a list of a theorem numbers involved)
    (REMOVEDUPLICATES (for p in pflist join <p:1 ! (SubProofNumbers1 (SubProofs p))
                                            >]))

(SubProofs
  [LAMBDA (pf)                                             (* given a ProofList entry, skip down to the
                                                           subproof portion, just past
                                                           (QUOTE (#)))
    (CDR (for x on pf thereis (LISTP x:1]))

(UpdateProof
  [LAMBDA (key keylst proofList)                           (* edited: " 6-Apr-79 08:48")
    (PROG (x y status)
          (x←(for k in keylst collect (ASSOC k proofList)))
          (y←(ASSOC key proofList))
          (status←(if (for p in x never p:2 MEMB '(assumed pending untried new))
                      then 'proved
                    else 'pending))
          (/PUTASSOC key <status !(for u in y::2 until (LISTP u) collect u)
                         <'filler ! RetryLog> ! x>
                     proofList)
```

```
        (if status='proved or key=CurrentTheoremNumber
            then (PRINTLINES T "Proposition" key status T)
                 (if keyIst and key=CurrentTheoremNumber
                     then (PRINTLINES T "Proof structure is:" T)
                          (PrintProof (ASSOC key proofList)
                                      0)
                          (TERPRI)))
        (RETURN status))
```

(UpdateProofList

```
  [LAMBDA (proofList alreadyChecked)                    (* edited: " 6-Apr-79 08:55")
    (for x in proofList when x:2='pending and x:1 ~MEMB alreadyChecked
       do (alreadyChecked← <x:1 ! alreadyChecked>)
          (alreadyChecked←(UNION (UpdateProofList x::4 alreadyChecked)
                                 alreadyChecked))
          (UpdateProof x:1 (for y in x::4 collect y:1)
                       TheoremList:ProofList))
    alreadyChecked])
```

(WhatRoots

```
  [LAMBDA (input)                                       (* edited: "18-SEP-78 17:06")
                                                        (* given a user input: all,roots,NIL, or 1 or
                                                        more numbers, ; returns list of predicate nos
                                                        to print)
    (if (NLISTP input)
        then input← <input>)
    (if input:1='roots or input:1='root or input:1=NIL
        then (Roots)
      elseif input:1='all
        then (for i to TheoremList:NumberPreds when (ASSOC i TheoremList:ProofList) collect i)
      elseif input:1=T
        then <(RootOf CurrentTheoremNumber) >
      else (if (NUMBERP input:1)
               then input
             else (PRINTLINES T input "?" T)
                  (ERROR!)))
```

(leaves

```
  [LAMBDA (pl status)                                   (* edited: " 6-Apr-79 08:48")

        (* given a list of proof lists, return list of #s of all nodes with the right status. We dont look below an
        okay node.)


    (PROG (n stat lower)
          (n←pl:1)
          (stat←pl:2)
          (RETURN (if stat MEMB status
                      then <n>
                    else lower←(SubProofs pl)
                         (for i in lower join (leaves i status])
```

(nextis

```
  [LAMBDA (given)                                       (* edited: " 6-Apr-79 08:48")
    (PROG [from spl (opens '(new untried assumed]
          (from←(RootOf (if given
                            else CurrentTheoremNumber)))
          (spl←(ASSOC from TheoremList:ProofList))
          (if spl
              then (RETURN (if spl MEMB opens
                               then <from>
                             else (REMOVEDUPLICATES (for i in (SubProofs spl) join (leaves i opens]
            else (PRINTLINES T from "?" T])
```

(retrycaller

```
  [LAMBDA (fn retryargs)                                (* edited: " 8-SEP-78 17:16")
    (PROG ((pos (for i from 1 as i in (ARGLIST fn) thereis i='retrycom)))
          (if pos
              then (PRINTLINES T fn (PrettyPrint retryargs T)
                               ";" T)
                   (APPLY fn <!!(to pos-1 collect NIL)
                               retryargs>)
```

```
            else (ERROR "retry can't find arg " fn))
)
(DECLARE: DONTCOPY
  (FILEMAP (NIL (672 13849 (AddPredicate 684 . 1736) (CheckAndSplit 1740 . 3041) (CheckForPropositionStatus 3045
 . 3384) (CheckForSuccess 3388 . 3986) (FetchLog 3990 . 4434) (GetPredicate 4438 . 4678) (GetProof 4682 . 4895 -
) (Log 4899 . 5103) (PrintBoth 5107 . 5573) (PrintBoth1 5577 . 6202) (PrintProof 6206 . 6656) (PrintProofs
6660 . 6978) (RootOf 6982 . 7826) (Roots 7830 . 8648) (SubProofNumbers 8652 . 8988) (SubProofNumbers1 8992 .
9443) (SubProofs 9447 . 9744) (UpdateProof 9748 . 10748) (UpdateProofList 10752 . 11264) (WhatRoots 11268 .
12168) (leaves 12172 . 12717) (nextis 12721 . 13341) (retrycaller 13345 . 13846)))))
STOP
```