

Reference: J3/97-256

The proposed syntax for identifying initial and final procedures for a type is that they be type-bound procedures with “names” (INITIAL) and (FINAL) respectively. Note that the parentheses are part of these “names”.

5 Example:

```
TYPE mytype
  REAL :: component
CONTAINS
  PROCEDURE :: (initial) => my_initial_sub
10  PROCEDURE :: initial => explicit_sub
  PROCEDURE :: (final) => my_final_sub
END TYPE mytype

TYPE(mytype) :: a_var
15
! In effect, a
!   CALL my_initial_sub(a_var)
! is generated automatically at the beginning of execution
! this scoping unit and
20 !   CALL my_final_sub(a_var)
! at the end of execution. There is no automatic generation of
!   CALL explicit_sub(a_var)
! because INITIAL (without parentheses) is not a special name, so you
! would need to write
25 !   CALL a_var%initial
! explicitly to get this effect.
```

[The “names” (INITIAL) and (FINAL) are not merely illustrative syntax, but the subgroup is open to considering alternative special “names”.]

Ω