Subject: Four problems with the interaction of assignment and parameterized derived types
From: Van Snyder

1 Introduction

There are at least four problems with the conspiracy of intrinsic assignment, polymorphism, and parameterized derived types.

At [135:33-35], the criterion to use intrinsic assignment depends on not selecting defined assignment, but this is specified to depend on the dynamic type of the *expr*. Generic resolution depends on the declared type. It further depends on the nonexistence of a defined assignment for the types, not the nonaccessibility of it.

According to [137:36-137:2], intrinsic assignment of objects of derived type ultimately devolves to intrinsic assignment of objects of intrinsic type. We have reasonable rules for those assignments. The type parameters of *variable* and *expr* are, however, required at [136:13] to be equal. This means that the useful rules for intrinsic assignment of objects of intrinsic type are completely inaccessible by the time the entities of intrinsic type that ultimately compose derived type objects are assigned.

This gives rise to curious anomalies:

```
type T1(K)
  kind :: K
  real(k) :: C1
end type T1
type(t1(kind(0.0e0))) :: RV
type(t1(kind(0.0d0))) :: DV
dv = rv                         ! illegal
dv%c1 = rv%c1                   ! allowed

type T2(N)
  character(len=n) :: C2
end type t2
type(t2(10)) :: SV
type(t2(20)) :: LV
lv = sv                         ! illegal
lv%c2 = sv%c2                   ! allowed
```

At [136:21-22], the rules for the conformance of types in intrinsic assignment refer to table 7.9, which is silent concerning whether the types are the declared type or the dynamic type.

At [135:33-35] the dynamic types of the *variable* and *expr* are required to be the same, but the *variable* is not permitted to be polymorphic. If the dynamic types are required to be the same, there is no problem allowing the *variable* to be polymorphic.

2 Edits

Edits refer to 00-007r3. Page and line numbers are displayed in the margin. Absent other instructions, a page and line number or line number range implies all of the indicated text

is to be replaced by immediately following text, while a page and line number followed by + (-) indicates that immediately following text is to be inserted after (before) the indicated line. Remarks for the editor are noted in the margin, or appear between [and] in the text.

---

(2) There is no accessible defined assignment for the declared types and kind type parameters of the *variable* and the *expr*, and the dynamic type of the *expr* is the same as the dynamic type of the *variable*. 135:33-35

[Editor: "same ... *variable*" ⇒ "same dynamic type as the dynamic type of *variable*".] 136:13