The specification in 97-256 already provides for the possibility of INITIAL procedures having additional arguments that affect the value to which an object of derived type is initialized. The paper proposed the specification, syntax, and edits for invoking such an alternative INITIAL procedure.

Basic Facility

The form "=*derived-type-spec* ([*actual-arg-spec-list*])" is specially recognized in the interpretation of *initialization* inR505. The argument list specifies the arguments after the first in the invocation of an INITIAL procedure. This basic facility would provide support for declarations such as

```
TYPE(interval) :: interval_1 = interval("1.0")
TYPE(interval) :: zero_to_pi = interval("0.0","3.14159265358979323846264 3")
```

allowing symbolic "constants" to be created outside the defining module.

Elsewhere, the form "*derived-type-spec* ([*actual-arg-spec-list*])" would be an alternative form of *structure-constructor* (R447), acceptable even where the components are unknown, and implemented by invoking an alternative INITIAL procedure on an anonymous object of that type. Thus, one would be allowed to write code such as

```
my_interval = my_interval + interval("0.1")
```

providing a facility similar to literal constants outside the defining module.

Variation

Should ":=([*actual-arg-spec-list*])" be acceptable as an alternative to "=*derived-type-spec* ([*actual-arg-spec-list*])"?

Edits

<to be supplied later>

Ω