Date: 5 March 1999
To: J3
From: Van Snyder
Subject: Syntax changes from 99-107

1 Introduction

Two syntax changes suggested in 99-107 are recommended by the /data subgroup.

1.1 Position of *private-sequence-stmts*

If the *private-sequence-stmts* were put after the *type-param-def-stmts* within a derived type definition, it would help one to remember that one has no control of accessibility of type parameters, and they're not affected by SEQUENCE. Changing the same BNF without any additional changes to syntax allows clarification of the description of three requirements elsewhere.

1.2 Change how *proc-binding* specifies null binding

The present syntax of the *proc-binding* requires one to create a procedure or procedure pointer in order to declare that there is no binding for a type-bound procedure. It's also dissimilar to syntax for procedure declarations and procedure component declarations.

2 Edits

Edits refer to 99-007. 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 the indicated line. Remarks for the editor are noted in the margin, or appear between [and] in the text.

2.1 Position of *private-sequence-stmts*

R422 *derived-type-def* **is** *derived-type-stmt* 39:18-23
[*data-component-part*]
[*type-bound-procedure-part*]
end-type-stmt

R422A *data-component-part* **is** [*type-param-def-stmt*] ...
[*private-sequence-stmt*] ...
[*component-def-stmt*] ...

[Editor: Replace "If a type ... *private-sequence-stmt*" by "If the *data-component-part* of a type definition statement contains a PRIVATE statement"] 47:31

[Editor: Replace 'PRIVATE statement that is a *private-sequence-stmt*" by "PRIVATE statement in a *data-component-part*"] 48:5-6

[Editor: Replace "the definition ... (4.5.1)" by "the *data-component-part* of the type definition does not include a PRIVATE statement (4.5.1)"] 340:37-38

2.2 Change how *proc-binding* specifies null binding

R439 *proc-binding*	**is** PROCEDURE [[, *binding-attr-list*] ::] ■ ■ *binding-name* [=> *procedure-name*] **or** PROCEDURE [([*proc-interface-name*])] ■ ■ [[, *binding-attr-list*] ::] ■ ■ *binding-name* => NULL()	42:4-5

Constraint: The procedure name specified by *procedure-name*, or *binding-name* if there is no *procedure-name*, shall be the name of an accessible module procedure or external procedure that has an explicit interface. If PASS_OBJ is specified, it shall have a scalar nonpointer nonallocatable dummy argument of type *type-name*. The first such dummy argument is called the passed-object dummy argument and shall be polymorphic if and only if *type-name* is extensible.

Constraint: The *proc-interface-name* shall be specified if and only if the NULL() binding is not overriding (4.5.3.2) an inherited (4.5.3.1) binding.

Constraint: The *proc-interface-name* shall be the name of an accessible explicit abstract interface. If *proc-interface-name* and PASS_OBJ are both specified, *proc-interface-name* shall have a scalar nonpointer nonallocatable dummy argument of type *type-name*. The first such dummy argument is called the passed-object dummy argument and shall be polymorphic if and only if *type-name* is extensible.

[Editor: Replace "*binding*" by "*procedure-name*"]	42:6
[Editor: Delete]	42:12-19
[Editor: Delete]	42:21-24