****

J3/02-323
Date: November 2002
To: J3
From: Aleksandar Donev
Subject: Interoperability of strings and Typing of C_PTR
Reference: J3-007R3

****

Interoperability of strings

Fortran strings (i.e. character scalars with length different then 1) are at present non interoperable with C strings (i.e. character arrays), even if their kind is C_CHAR. But they are interoperable if the length is one. This makes no sense. One should be able to for example apply C_LOC to a Fortran string.

I am not sure when and why the decision in the current draft was made, but think it can be changed.

Typing of C_PTR

The Fortran 2002 draft makes two blunders concerning interoperability with C pointers via the C_PTR argument:

1. Both data and function pointers are made interoperable with the same type C_PTR. The C standard makes careful distinctions between data and function pointers and they are not mixable (not even via casts), and there are implementations (say in embedded systems) where function pointers are short (say 16 bits) since the executable fits in a small piece of the memory, while data pointers are long (32 bits) since there is more memory available.

A valid option is to split C_PTR into two types, C_DATA_PTR and C_PROC_PTR, or something like that. This can actually be done at present, though at this point in time I would not put a high priority on it.

2. C pointers from the perspective of Fortran 2002 are non-typed. The justification is that all C implementations have the same physical representation for all pointers, regardless of type. However, making all C pointers untyped makes it hard to do compile-time checks and to read codes. An alternative would have been to have typed C pointers, say:

```
INTEGER(C_INT), C_PTR :: my_pointer
INTEGER(C_INT) :: my_int
REAL(C_FLOAT) :: my_float

my_pointer=C_LOC(my_int) ! OK
my_pointer=C_LOC(my_float) ! Rejected at compile time
```

The problem with this is dealing with C's pointers to pointers.
I do understand that there is nothing that can be done about this issue at this point.