Date: 15 February 1997
To: X3J3
From: Van Snyder
Subject: Swap operation

The capability advocated in Section 21 of X3J3/97-114r2, a SWAP operation, was added to the "C" list during X3J3 meeting 140. Illustrative editorial changes that apply to ISO/IEC 1539:1991(E) were suggested in X3J3/97-114r2, but are not reproduced here.

Section 21 of 97-114r2: A SWAP operator would be useful

Something I'd like to see that is simple and not on your list is a swap operator. Thus

```
A:=:B ! or ??? swaps the contents of A and B.
A<=>B ! or ??? swaps the pointers A and B.
```

This would be allowed anywhere that both A=B and B=A (or A=>B and B=>A) are allowed. One could also restrict A and B to be of the same type without losing any important functionality. This is

- Easy to implement,
- Easy to describe,
- Has negligible interaction with the rest of the language,
- Makes some code more clear (One needn't wonder "is that temp. variable ever used someplace else?"),
- Has frequent use,
- Is more likely to be optimized by a simple compiler.

It should be indivisible when applied to scalars of intrinsic "atomic" types (not including `COMPLEX` or `CHARACTER` with length > 1), so it could be used for "test and set" in multiprocessor systems, and it could be indivisible for each component of composite objects (but the latter isn't as important).