

Re: IMPLICIT NONE (F2k8+/-)

Source: <http://coding.derkeiler.com/Archive/Fortran/comp.lang.fortran/2005-10/msg00716.html>

- *From:* "Gary L. Scott" <garyscott@xxxxxxx>
 - *Date:* Sun, 23 Oct 2005 15:58:34 -0500
-

Edward N Bromhead wrote:

"Gary L. Scott" <garyscott@xxxxxxx> wrote in message
news:111n80341kkhtbd@xx

IMPLICIT NONE should finally be made default
beginning at F2k8. Compiler switches can be
specified to get the older behavior. There is
absolutely no good reason to continue this way.
--

Gary Scott

The assertion that there is absolutely no good reason to
continue with a convention that has served Fortran for 50
years (nearly), is itself, no good reason for change. Plenty
of code has been written, and plenty of good work done, using
implicit typing. Those of us who like implicit typing find
that we can instantly recognize the type of a variable from
its initial letter, provided that explicit typing has not been
turned on.

There is ample evidence in very large codes, that inhibition of implicit
typing is VERY beneficial. I agree with you that smallish applications
such as "engineering tools" with 1 or a very small number of developers
are unlikely to benefit significantly. However, large teamed projects
definitely do benefit. It is just plain wishful thinking to believe that
having the compiler perform additional checks that you get without
implicit typing is of benefit to a majority of large scale applications.

Re: IMPLICIT NONE (F2k8+/-)

The argument that this leads to errors because of mistyped variable names is in practice not so good as it looks. All arrays have to be dimensioned somewhere, so that a mistyped (in the sense of key punching) array name shows up during compilation or linking, often/usually as a call to a non-existent function. The type of user-defined functions is also given at the outset of the code for that function. We have to conclude that only single variables are susceptible to mistyping and not being trapped. Just how many single variables are there in an individual subprogram? So many that a simple listing of all the variables used in a given subprogram won't show, instantly, that such an error exists? The IBM 1130 compiler could do that, and computer science has marched on since 1970! I rarely use more than a few single variables in a subprogram, and get to know their names intimately.

IMPLICIT NONE is the case that could be selected by a compiler switch, as the user of this option will have declared the type of every variable explicitly, and other than the case where the programmer has made an error, it is immaterial whether or not it is used. In other words, IMPLICIT NONE is a redundant statement in code that is correct!

The present arrangement permits all preferences to be served. That seems to me to be an ideal compromise, and an excellent reason not to change.

Again, it impacts nobody in any significant way, just add a compile time switch. Nobody will be inordinately placed in hardship. It isn't even big enough of a deal to add a switch as to even argue against it.

Eddie Bromhead

--

Gary Scott
<mailto:garyscott@xxxxxxx>

Fortran Library: <http://www.fortranlib.com>

Re: IMPLICIT NONE (F2k8+/-)

Re: IMPLICIT NONE (F2k8+/-)

Support the Original G95 Project: <http://www.g95.org>

-OR-

Support the GNU GFortran Project: <http://gcc.gnu.org/fortran/index.html>

Why are there two? God only knows.

If you want to do the impossible, don't hire an expert because he knows it can't be done.

-- Henry Ford

.