

Table of Contents

1	Fortran 90, 95, 2003, 2008, 77 Information	2
1.1	The development and evolution of Fortran	2
1.1.1	The new features of Fortran 90	2
1.1.2	The new features of Fortran 95	3
1.1.3	ISO TR 15580, 15581	3
1.1.4	Fortran 2003	3
1.1.5	Fortran 2008	4
1.1.6	Current Status	4
1.2	Introduction to Programming with Fortran: With Coverage of Fortran 90, 95, 2003 and 77 5	
1.3	Introducing Fortran 95: With coverage of ISO TR 15580 and TR 15581	5
1.4	Fortran Resource file.....	5
1.5	Fortran 90 List at JISCmail.....	5
1.6	The ACM Fortran Forum Newsletter	6
1.7	BCS Fortran Specialist Group.....	6
1.8	comp.lang.fortran.....	6
1.9	Object Oriented Programming In Fortran 90 and 95	6
1.10	Computational Science Education Paper: Comparison of Fortran 77, Fortran 90, C and C++	7
1.11	Interactive Fortran 77: A Hands on Approach, Ian Chivers and Jane Sleightholme. 7	7

1 Fortran 90, 95, 2003, 2008, 77 Information

Ian Chivers and Jane Sleightholme

Last updated December 2007

The pdf version of this document can be found at:

http://www.chiversandbryan.co.uk/fortran/files/fortran_information.pdf

1.1 The development and evolution of Fortran

Fortran was one of the first high level languages developed and widely adopted by the academic and scientific community. It was developed over a three year period (1954-1957) by a team at IBM lead by John Backus. Fortran stands for FORMula TRANslation and was used mainly by people with a scientific background for solving problems with a significant arithmetic content.

By 1966 and the first standard it was widely used, easy to teach, had demonstrated the benefits of subroutines and independent compilation, was relatively machine independent and often had very efficient implementations.

By the 1970s it had started to show its age in relation to some of the other languages that had emerged and there was pressure on the X3J3 committee to incorporate changes to bring it more into line with mainstream language development. It was standardised in 1978 (even though the next version was called Fortran 77) and whilst the changes that were made were on the welcome side many felt that they had not gone far enough.

Other languages emerged and established themselves. Pascal, Ada, Modula 2, C, C++ all became rivals to Fortran in the scientific and academic communities.

Fortran was next standardised in 1991 (yet again out by one) and called Fortran 90. This was major improvement and the changes are given below.

Work continued and 1996 saw the publication of the 1995 standard. Details are given below.

There are also two ISO Technical Reports that have been published on IEEE Floating Point Arithmetic and Allocatable Attributes. Details of these are given later.

Work on Fortran 2003 is now complete and details are given below.

Work on Fortran 2008 is nearing completion and details are given below.

1.1.1 The new features of Fortran 90

The new language features include:

- free source form:
 - names can be up to 31 characters in length
 - blanks are significant
 - lines up to 132 characters in length
 - up to 39 continuation lines
 - ; as statement separator for multiple statements per line
 - ! as comment symbol
 - include option for source text from files.
- modern control structures
 - Fortran 90 has a modern DO statement, with CYCLE and EXIT options and the control part of the DO can be conventional iteration, WHILE or no control clause. There is also a CASE statement.

- specification of numeric precision
There is now a clean way to control numeric precision.
- whole array processing
It is now possible to treat arrays as whole objects and simply write
 $A=B*\text{SIN}(A)$
where A and B are arrays.
- dynamic behaviour, including allocate, deallocate, pointers, recursion
- user defined data types
- modules, and with them come
operator overloading
generic procedures

1.1.2 The new features of Fortran 95

The next revision was finalised in 1996. It was a relatively small change compared to the changes between the Fortran 77 and Fortran 90 standard. There were some major features and some minor corrections and clarifications. Some of these are to keep Fortran in step with the work in the HPF area.

The major features include

- FORALL statement and construct
- pure and elemental user defined subprograms
- initial association status for pointers
- implicit initialisation of derived type objects

Minor features include

- new intrinsic function NULL
- new intrinsic function CPU_TIME
- automatic deallocation of allocatable arrays
- SIGN can distinguish between +0 and -0
- comments in namelist input data
- references to pure functions in specification expressions
- changes to some intrinsic functions,

1.1.3 ISO TR 15580, 15581

Details of the work in the areas of IEEE Floating Point Arithmetic and Allocatable Attributes can be found at NAG. Their address is:

<http://www.nag.co.uk/nagware/NP/doc/TR.asp>

1.1.4 Fortran 2003

The following is a verbatim extract from the standard.

Fortran 2003 contains several extensions to Fortran 95; among them are:

- Derived-type enhancements: parameterized derived types (allows the kind, length, or shape of a derived types components to be chosen when the derived type is used), mixed component accessibility (allows different components to

have different accessibility), public entities of private type, improved structure constructors, and finalizers.

- Object oriented programming support: enhanced data abstraction (allows one type to extend the definition of another type), polymorphism (allows the type of a variable to vary at runtime), dynamic type allocation, SELECT TYPE construct (allows a choice of execution flow depending upon the type a polymorphic object currently has), and type-bound procedures.
- The ASSOCIATE construct (allows a complex expression or object to be denoted by a simple symbol).
- Data manipulation enhancements: allocatable components, deferred type parameters, VOLATILE attribute, explicit type specification in array constructors, INTENT specification of pointer arguments, specified lower bounds of pointer assignment and pointer rank remapping, extended initialization expressions, MAX and MIN intrinsics for character type, and enhanced complex constants.
- Input/output enhancements: asynchronous transfer operations (allows a program to continue to process data while an input/output transfer occurs), stream access (allows access to a file without reference to any record structure), user specified transfer operations for derived types, user specified control of rounding during format conversions, the FLUSH statement, named constants for preconnected units, regularization of input/output keywords, and access to input/output error messages.
- Procedure pointers.
- Scoping enhancements: the ability to rename defined operators (supports greater data abstraction) and control of host association into interface bodies.
- Support for IEC 60559 (IEEE 754) exceptions and arithmetic (to the extent a processor's arithmetic supports the IEC standard).
- Interoperability with the C programming language (allows portable access to many libraries and the low-level facilities provided by C and allows the portable use of Fortran libraries by programs written in C).
- Support for international usage: (ISO 10646) and choice of decimal or comma in numeric formatted input/output.
- Enhanced integration with the host operating system: access to command line arguments and environment variables, and access to the processor's error messages (improves the ability to handle exceptional conditions).

A very readable article by John Reid, the Working Group 5 convenor can be found at below.

http://kcl.ac.uk/kis/support/cit/fortran/john_reid_new_2003.pdf

We've updated the original document so that the web links in this document are active, i.e. can be clicked on.

1.1.5 Fortran 2008

The February 2008 joint J3/WG5 meeting should finalise details about this standard.

1.1.6 Current Status

People are gradually moving from Fortran 77 to the more modern versions. Most compilers today support the Fortran 95 standard.

As standard Fortran 77 is a complete subset of modern Fortran it is possible to simply recompile your old code with a new compiler and develop code in Fortran 90 and 95.

This protects your investment in your old, working Fortran 77 code.

The new features of the language bring it up to date and increase the range of problems that can be solved easily. The language provides a very good framework for modern software development. The separation of the implementation from the design is a very powerful tool for reliable software development. We see in the language the first steps to object orientation with encapsulation and a limited support for polymorphism, without the steep learning curve demanded by C++.

1.2 Introduction to Programming with Fortran: With Coverage of Fortran 90, 95, 2003 and 77

Springer link is at

<http://www.springer.com/uk/home/generic/search/results?SGWID=3-40109-22-52482633-0>

Program examples can be found at

http://chiversandbryan.co.uk/fortran/f2003_book_examples/

and

<http://chiversandbryan.co.uk/fortran/examples/>

has details of zip and tar versions of the examples.

1.3 Introducing Fortran 95: With coverage of ISO TR 15580 and TR 15581 Springer Verlag, ISBN 1-85233-276-X

The Springer site has details at:

<http://www.springer.com/uk/home/generic/search/results?SGWID=3-40109-22-2095720-0>

Program examples can be found at:

http://chiversandbryan.co.uk/fortran/f95_book_examples/

1.4 Fortran Resource file

This is our primary Fortran Resource file. The original basis for this document was Mike Metcalf's Fortran Information File. The next input came from people on comp-fortran-90. Details of how to subscribe or browse this list can be found in this document. If you have any corrections, additions, suggestions etc to make please contact us and we will endeavour to include your comments in later versions. Thanks to all the people who have contributed.

One web address for the file is:

http://www.chiversandbryan.co.uk/fortran/files/fortran_resources.pdf

Versions can also be found at King's College and the JISCMail site.

http://www.kcl.ac.uk/kis/support/cit/fortran/pdfs/fortran_resources.pdf

and

<http://www.jiscmail.ac.uk/cgi-bin/quicktype.cgi?m>

in the files area.

1.5 Fortran 90 List at JISCMail

<http://www.jiscmail.ac.uk/lists/comp-fortran-90.html>

comp-fortran-90 home page.

This is a list dedicated to the latest developments with the Fortran language. In particular it looks at the developments that have taken place since the publication of the Fortran 90 standard, including HPF.

If you want more information then follow the links. If you have any comments or suggestions to make regarding this list then please mail us as we are the primary list owners.

1.6 The ACM Fortran Forum Newsletter

The Fortran Forum is a publication of the Special Interest Group in Programming Languages (SIGPLAN) of the Association for Computing Machinery (ACM). Founded by Loren Meissner, the ACM Fortran Forum is published three times a year. Visit

http://portal.acm.org/browse_dl.cfm?coll=portal&dl=ACM&idx=J286&linked=1&part=newsletter

The following address

<http://www.sigplan.org/>

has details on how to subscribe.

If you would like to submit an article please contact Ian Chivers as he is the current editor.

1.7 BCS Fortran Specialist Group

The Group was established in 1970 to provide an open forum for Fortran users. Its main aims are

- to disseminate information about Fortran and its application in various fields, to provide a platform for discussion of users' needs and requirements in future versions of Fortran,
- to encourage the development of the language in collaboration with national and international standardization bodies,
- to promote the use of the Fortran language.

Visit

<http://www.fortran.bcs.org/index.php>

for more information.

1.8 comp.lang.fortran

This is a usenet newsgroup for discussion of Fortran. Google has the group at

<http://groups.google.co.uk/group/comp.lang.fortran/topics?hl=en>

High activity, 1290 subscribers as of December 2007.

1.9 Object Oriented Programming In Fortran 90 and 95

The first reference is a book by Ed Akin called Object Oriented Programming via Fortran 90/95.

Here is a synopsis from Amazon

Learn how to write technical applications in a modern object-oriented approach, using Fortran 90 or 95. This book will teach you how to stop focussing on the traditional procedural abilities of Fortran and to employ the principles of object-oriented programming to produce clear, highly efficient executable codes. Get ready now to take advantage of all the features of the finalized, fully object-oriented Fortran 200X! In addition to covering the OOP methodologies the book also covers the basic foundation of the language and good programming skills, making the book valuable also as a good migration tool for experienced Fortran programmers who want to smoothly pick up the OOP paradigm. The author highlights com-

mon themes by using comparisons with Matlab and C++ and uses numerous cross-referenced examples to convey all concepts quickly and clearly. Complete code for the examples is included on the accompanying CD-ROM

Here are links to work carried out in the US at UCLA under contracts from US-DOE and NSF, and at Jet Propulsion Laboratory, CIT under contract within NASA. The work looks at research in oo programming using Fortran 90.

<http://www.cs.rpi.edu/~szymansk/oof90.html>

Home page for this work

1.10 Computational Science Education Paper: Comparison of Fortran 77, Fortran 90, C and C++

We have put up a postscript version of a paper by the Computational Science Education Project. This paper compares Fortran 77, Fortran 90, C and C++ using five criteria. The criteria are numerical robustness, data parallelism, data abstraction, object oriented programming and functional programming.

This is an essential read for anyone wanting to know about the strengths and weaknesses of the above four languages in these areas.

Fortran 90 and Computational Science

Copyright ©) 1991, 1992, 1993, 1994, 1995 by the Computational Science Education Project.

This electronic book is copyrighted, and protected by the copyright laws of the United States. This (and all associated documents in the system) must contain the above copyright notice. If this electronic book is used anywhere other than the projects's original system, CSEP must be notified in writing (email is acceptable) and the copyright notice must remain intact.

The paper provides a brief overview of Fortran 90 before comparing C, C++, Fortran 77 and Fortran 90 using five criteria. The criteria are numerical robustness, data parallelism, data abstraction, object oriented programming and functional programming. They rank the four languages within each criteria. Possibly a little technical for someone with only a limited C or Fortran 77 background.

Highly recommended for anyone wanting to bring themselves up to date with current language thinking.

Postscript version of the above paper.

1.11 Interactive Fortran 77: A Hands on Approach, Ian Chivers and Jane Sleightholme.

Copyright on this book has reverted to Jane and I so we are making it available electronically. Further details are given below.

If you need further information on any of the above then please contact Ian Chivers or Jane Sleightholme.

Interactive Fortran 77: A Hands on Approach Ian Chivers and Jane Sleightholme.

Copyright ©) Ian D Chivers and Jane Sleightholme.

Legal comments

Unless otherwise specified, Ian D Chivers and Jane Sleightholme hold all rights, including copyright and retains such rights. This work may be distributed in its entirety provided the work is distributed as a whole with this copyright notice intact.

This work may not be distributed in hard copy or other machine readable form, redistributed, transmitted or translated without prior written authorization from Ian D Chivers and Jane Sleightholme.

Commercial use can only be allowed by specific license agreements. The accuracy of this document cannot be guaranteed. Ian D Chivers and Jane Sleightholme make no warranty, either express or implied, with respect to the use of any information and assumes no liabilities for loss or damage, whether such loss or damage is caused by error or omission.

If this electronic book is made available anywhere other than the original system, Ian Chivers or Jane Sleightholme must be notified in writing (email is acceptable) and the copyright notice must remain intact.

The following directory has several Fortran related files:

<http://www.chiversandbryan.co.uk/fortran/files/>

and the address of the Fortran 77 book is:

<http://www.chiversandbryan.co.uk/fortran/files/f77book.pdf>