Compiler Support for the Fortran 2003 and 2008 Standards Revision 17

Ian D Chivers & Jane Sleightholme
Ian Chivers: Rhymney Consulting, London.
Jane Sleightholme: FortranPlus, London.
ian@rhymneyconsulting.co.uk
jane@fortranplus.co.uk

Introduction

This is a repeating article in Fortran Forum. The first version appeared in Fortran Forum in April 2007. The basis for the entries in the original list of features was a report by John Reid. An electronic version can be found at:

`ftp://ftp.nag.co.uk/sc22wg5/N1601-N1650/N1648.pdf`

If you are a compiler vendor and would like to be included in future versions of this table please email one of us with details and they will be added to the table and published in Fortran Forum.

Acknowledgements for the original article

An email was sent to the J3 list asking for information about compiler support for the new features of the Fortran 2003 standard. The following people contributed to the original article:

- Bill Long, Cray
- Joost VandeVondele
- Van Snyder
- Tobias Burnus and Brooks Moses, gfortran
- Andy Vaught, g95
- Robert Holmes, NAG

Thanks.

First appeared April 2007.

Revision 1 - August 2007

Two new compiler vendors were added. The information on the Intel compiler has been taken from the release notes that came with release 10 of the compiler. The information on the IBM entry has been taken from their web site. Ian Bush posted an article to comp.lang.fortran regarding this release (IBM XL Fortran Enterprise Edition for AIX, V11.1). Thanks Ian. See

`http://publib.boulder.ibm.com/infocenter/comphelp/v9v111/index.jsp?topic=/com.ibm.xlf111.aix.doc/getstart/new_features.htm`

for more information.

Intel and IBM were contacted to ask them to verify the information.

- Jim Xia of IBM corrected their entry, thanks Jim.
- Stan Whitlock of Intel corrected their entry, thanks Stan.

If there are any errors please notify us and we will correct them in the next version of this article.

Revision 2 - August 2008

Sun has been added and there have been a few corrections and updates to some of the other entries.

- Michael Ingrassia of Sun corrected their entry, thanks Michael.

Revision 3 - April 2009

The entries for Cray, gfortran (11 changes) , Intel (18 changes) and NAG (9 changes) have been updated.

Revision 4 - August 2009

We've had replies from Cray (Bill Long) , gfortran (Tobias Burnus), g95 (Andy Vaught), Intel (Stan Whitlock), Nag (Malcolm Cohen) and Sun (Robert Corbett).

We've also added two entries suggested by Richard Maine. Here is the text of the message we received from Richard.

I just got the latest Fortran Forum and noticed two somewhat related Fortran 2003 features that I personally think are important, but aren't reflected in your table of features. If convenient, they might be useful to add to the table.

1. Allocatable scalars. To me, this is an important feature for object orientation, and in particular for polymorphism. Basically, a polymorphic object has to be either a pointer or an allocatable (or a dummy argument, which is a bit restrictive). In my experiments with polymorphism, the polymorphic objects pretty much always naturally "wanted" to be allocatable. But the NAG compiler (which I was using at the time) didn't yet support allocatable scalars. This meant that I either needed to make all the polymorphic objects pointers or make them arrays (possibly of size 1). Neither of these alternatives was attractive at all. I found this a significant enough shortcoming to keep me from using the polymorphic features. Thus, I'd think this would be something people would want to know about a compiler if they planned to use polymorphism.

2. Allocatable character length. I think that allocatable character length is one of the biggest "sleeper" features of f2003. It wasn't even on the list of f2003 requirements, and thus sometimes doesn't show up in lists of new features. It just naturally arise from allocatable length parameters for parameterized derived types. It seemed like one should allow the same thing for the one intrinsic type with a length type parameter. And lo, when it was all put together, it seemed like this was finally a good way to do variable length strings in Fortran. It integrates with the rest of the language immensely better than iso_varying_string has any hope of doing. In fact, as I said, it integrates so well that it came about as a consequence of the integration of other features. Allocatable-length character strings act like so many people intuitively think of character strings, unlike the fixed-length character strings that we've had since f77.

Although this is related to allocatable scalars, in that you certainly want to be able to have allocatable character strings that are scalar, it is also a separate feature in that you can have allocatable scalars without necessarily allowing character length to be allocatable. It is also different in application, in that I see the main other usage of allocatable scalars as being for polymorphism, whereas allocatable character strings are not much related to polymorphism. It is also useful independent of parameterized derived types. I personally expect to see allocatable character strings used far more than parameterized derived types, even though it was the requirement for parameterized derived types that lead to allocatable character strings. I could almost see allocatable character strings as becoming the "normal" way that most character string variables are done.

Thanks Richard.

We've also added entries for the Fortran 2008 standard. The entries are based on the following document:

`ftp://ftp.nag.co.uk/sc22wg5/N1701-N1750/N1729.pdf`

The last change are entries for compilers that support the Fortran 95 standard, and a list of compilers that are no longer under development but did support Fortran 90, and finally compilers which are available but we have no information on.

We have included the above for completeness. Given the widely differing levels of compiler standard conformance today we wanted to make this information available to people choosing a compiler.

Thanks to everyone who has provided the data.

Revision 5 - August 2010

The IBM entry has been updated. See

`http://www-01.ibm.com/common/ssi/rep_ca/3/897/ENUS210-103/ENUS210-103.PDF`

The entry for gfortran has been updated. There is an entry for HP. The entry refers to the March 2010 release.

John Reid has also updated N1729.pdf and the latest version can be found at

`ftp://ftp.nag.co.uk/sc22wg5/N1801-N1850/`

Revision 6 - December 2010

The gfortran entry has been updated. Here is part of an email we received from Tobias Burnus.

- Hi, as the development of GCC has almost reached the end of Stage 1, I thought I could already update the F2003/F2008 conformance status for the December issue of ACM Fortran Forum. (Stage 1 allows for larger changes; it is followed by Stage 3 (!) which allows only smaller bug fixes, regression fixes and documentation updates.) Past experience suggests that 4.6.0 will be released next March as it won't be ready before Christmas - and it takes a while to fix the new issues reported during the Christmas break. (It is really a break as most developers are paid for GCC work ©, Ada, middle-end, target parts) and take off - only gfortran is purely developed in the spare time.).....gfortran 4.6 will presumably also allow to use REAL(16) (128 bit floating-point numbers) on x86, x86-64, and ia64 systems, which are emulated in software; so far only the real kinds 4, 8 and 10 (80bit FP) were supported on those systems. (This library inclusion had to be approved by the Free Software Foundation - but that problem seems to be mostly solved.)

For a complete list of what's new in 4.6 visit:

`http://gcc.gnu.org/gcc-4.6/changes.html`

Thanks Tobias.

Revision 7 - August 2011

The main driving force for the changes in this revision was an email from Stan Whitlock at Intel. Here is an extract from Stan's email.

I hope you will be updating the F2008 features list based on John Reid's updated article: ISO/N1828 - Features of F2008 - John Reid - latest adds features.

The feature list for Fortran 2008 used in the original article were taken from John Reid's earlier paper:

`ftp://ftp.nag.co.uk/sc22wg5/N1701-N1750/N1729.pdf`

The entries in the table are now taken from the contents of the N1828.pdf document. This is available at

`ftp://ftp.nag.co.uk/sc22wg5/N1801-N1850/N1828.pdf`

We have also added entries for two more Fortran compiler companies, Absoft (thanks Wood Lotz) and PGI (thanks Pat Brooks, Dave Norton and Brent Leback)

Revision 8 - December 2011

Corrections from Stan Whitlock

I have attached an updated spreadsheet with F2003 and F2008 tabs for Intel Fortran 12.1. There is little change over 12.0 but there are several typos. New text is in blue; editing directions are in red.

Damian's suggestion to add "generic procedure interfaces named the same as a type in the same module" as yet another F2003 feature is acceptable. I included that in the attached.

Thanks Stan.

Corrections from Malcolm Cohen. Here is the email from Malcolm.

You are probably aware of these, but anyway…

In the F2008 table, the subheading "Input/Output" has a spurious "N" in the PGI column.

Near the end of the table, there are two entries

"Null pointer or unallocated allocatable as an absent dummy argument"

and

"Null pointer as a missing dummy argument"; these would appear to be the same feature.

Also, the "generic resolution" line says "by pointer or allocatable attribute", it is missing any mention of "data object vs. procedure". (I think there is an argument for these two different extensions to generic resolution being mentioned separately.)

Thanks Malcolm, these have now been incorporated into the table.

The following was received from Tobias Burnus.

Dear Ian and Jane, dear all,

first, thanks for providing the helpful list of supported F2003/F2008 features. Unfortunately, coding for Fortran 2003 still requires such a list.

Whitlock, Stan wrote:

> I have attached an updated spreadsheet with F2003 and F2008 tabs for Intel Fortran 12.1. [...] Damian's suggestion to add "generic procedure interfaces named the same as a type in the same module" as yet another F2003 feature is acceptable. I included that in the attached.

Attached you find the modifications for gfortran; I also included an answer to Damian's item (unfortunately: "No" as of now).

I added additional items for the DTR 29113, which might a bit premature.

Most changes apply to GCC 4.6 (released: March 2011) *and* 4.7 (developer version, but builds widely used). Only one F2008 (all constants) and two DTR29113 items are better supported in 4.7 and marked such. (GCC 4.7 will be released around March/April 2012.)

Thanks Tobias.

The following was received from Pat Brooks

Greetings Ian and Greetings Jane,

Updated spreadsheet from PGI attached. Regarding Damian's request to add a item for generic procedure interfaces named the same as a type in the same module, yes please include it, and yes the PGI compiler supports this feature. You may want to verify that with Damian and please let us know if you hear any comments to the contrary.

Thanks and Best Regards, Pat Brooks

Thanks Pat, these have been incorporated.

Updates have also been received from Bill Long at Cray. Thanks Bill.

Revision 9 - April 2012

We received an email from Rafik Zurob at IBM in January and have corrected the IBM entry. To clarify the comment Rafik raised about compiler version number this is in the 2003 table and 2008 table as people wanted to know which versions of the compilers were being referred to in the tables, and is not a reference to the Fortran 2008 feature. Malcolm Cohen also spotted a couple of mistakes. Thanks to you both.

Revision 10 - August 2012

We have received a couple of emails regarding the table.

- Hi:, I was looking at your joint article in the latest issue of Fortran Forum. Regarding your table of features: It is fairly obvious that: Y=yes, N=no, but what on earth is "P"? Perhaps ? Thank you. Mike Milgram

We have corrected this issue. Thanks Mike.

- Hi Ian, I only spotted 2 things that need changing, as per attached. Malcolm Cohen.

Corrected - thanks Malcolm.

- Hi Ian and Jane - We announced the availability of XL Fortran V14.1 today. This release adds support for several Fortran 2008 features. Please find the updated spreadsheet attached. (In the F2003 tab, I added 13.1 to the compiler version since that's the first version with full Fortran 2003 compliance. I've updated the F2008 tab with the information for the 14.1 compiler.) Also, I noticed that the spreadsheet (and N1828.pdf) do not list support for the ERROR STOP statement. Should that be a separate item? Regards Rafik.

Thanks Rafik.

- Greetings Ian, Please find attached an updated spreadsheet for PGI. Also, as we mentioned last time, PGI supports Damian's suggestion for "Generic procedure interfaces named the same as a type in the same module". In the last report, our comments to that effect were published but the table wasn't updated. We'd appreciate it if you could please double check that for us. Let us know if we can provide any further clarification. Thanks and Best Regards, Pat Brooks

Thanks Pat.

Revision 11 - December 2012

We received emails from a variety of people including Wood Lotz at Absoft and Bill Long at Cray. The Cray compiler now supports the Fortran 2008 standard. Thanks to the contributors. The following document is being worked on at the moment.

- [1891 The new features of Fortran 2008 (Reid) - supersedes N1828]

Visit

`ftp://ftp.nag.co.uk/sc22wg5/N1851-N1900/`

for up to date information.

Revision 12 - April 2013

Minor corrections.

Revision 13 - August 2013

There are updates to several of the compilers. Thanks to Tobias Burnus, Wood Lotz, Polyhedron Software and Intel for providing the information.

The next revision will include an entry for the features introduced with ISO/IEC TS 29113:2012 which was published in 2012. Available at

`ftp://ftp.nag.co.uk/sc22wg5/N1851-N1900/N1942.pdf`

Tobias Burnus notified us about the document and an article is being prepared by Reinhold Bader which will form the basis for the entries in the table. Thanks to Tobias and Reinhold.

Revision 14 - December 2013

The major update in this edition is the inclusion of details of TS29113 and the original idea and request came from Tobias Burnus. The content of the table has been based on interchanges that took place between several people including Reinhold Bader, Bill Long and Malcolm Cohen. This edition of Fortran Forum also includes a paper by Reinhold Bader on this TS.

Nag have a draft of the TS.

`ftp://ftp.nag.co.uk/sc22wg5/N1901-N1950/N1942.pdf`

The full TS can be bought at

http://www.iso.org/iso/home/store/catalogue_tc/catalogue_detail.htm?csnumber=45136

The Absoft, Intel, Nag and PGI entries have also been updated.

Revision 15 - August 2014

John Reid has produced paper N1891 (March 2014) as an update to paper N1828 (May 2010), which is included in modified form in this edition of Fortran Forum. The table on Fortran 2008 conformance has been modified to reflect these changes. Additional changes have been made to the headings of this table in response comments made by-Malcolm Cohen of NAG and Rafik Zurob of IBM. The Cray, IBM, Intel, Nag, Oracle and PG entries have been updated in response information supplied by these vendors.

Bob Corbett has pointed out that the table of Fortran 2003 features includes ISO TR 15581 Allocatable Enhancements, and that the Fortran 2003 specification of these enhancements is incompatible with the specification in the TR. Some current Fortran implementations implement the old, non comforming semantics.

Thanks to everyone for their contributions.

Revision 16 - June 2015

Thanks to Stan Whitlock, Bill Long and Mark Leair for the updates, corrections and comments.

Revision 17 - November 2015

Paul Richard Thomas provided the updates for the gfortran entry to the 5.2 release. Thanks very much to Paul and all of the gfortran team.

Added summary information for each of the following tables

- Fortran 2003 features and conformance,
- Fortran 2008 features and conformance
- TS29113 features and conformance

providing totals by compiler vendor and feature.

Fortran 2003 Features	**Absoft**	**Cray**	**g95**	**gfortran**	**HP**	**IBM**	**Intel**	**NAG**	**Oracle**	**Path scale**	**PGI**
Compiler version number	**14**	**8.4.0**		**5.2**		**13.1**	**16.0**	**6.0**	**8.7, 32**	**4**	**15.5**
ISO TR 15580 IEEE Arithmetic	Y	Y	P	Y	Y	Y	Y	Y	Y	Y	Y
ISO TR 15581 Allocatable Enhancements	Y	Y	Y	Y	Y	Y	Y	Y	Y, 33	Y	Y
Data enhancements and object orientation											
Parameterized derived types	N	Y	N	N	N	Y	Y	P, 34	N	N	Y
Procedure pointers	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y
Finalization	N	Y	N	P, 31	N	Y	Y	Y	Y	N	Y
Procedures bound by name to a type	N	Y	N	Y	N	Y	Y	Y	Y	N	Y
The PASS attribute	Y	Y	N	Y	Y	Y	Y	Y	Y	N	Y
Procedures bound to a type as operators	N	Y	N	Y	N	Y	Y	Y	Y	N	Y
Type extension	N	Y	N	Y	N	Y	Y	Y	Y	N	Y
Overriding a type-bound procedure	N	Y	N	Y	N	Y	Y	Y	Y	N	Y
Enumerations	Y	Y	Y	Y	N	Y	Y	Y	Y	N	Y
ASSOCIATE construct	N	Y	N	P	N	Y	Y	Y	N	N	Y
Polymorphic entities	N	Y	N	Y	N	Y	Y	Y	Y	N	Y
SELECT TYPE construct	N	Y	N	Y	N	Y	Y	Y	Y	N	Y
Deferred bindings and abstract types	Y	Y	N	Y	N	Y	Y	Y	Y	N	Y
Allocatable scalars, 12		Y		Y	N	Y	Y	Y	Y	N	Y
Allocatable character length, 12		Y		P	N	Y	Y	Y	Y	N	Y
Miscellaneous enhancements											
Structure constructors	Y	Y	Y	Y	N	Y	Y	Y	Y	N	Y

Fortran 2003 Features	**Absoft**	**Cray**	**g95**	**gfortran**	**HP**	**IBM**	**Intel**	**NAG**	**Oracle**	**Path scale**	**PGI**
Compiler version number	**14**	**8.4.0**		**5.2**		**13.1**	**16.0**	**6.0**	**8.7, 32**	**4**	**15.5**
Generic procedure interfaces named the same as a type in the same module				P,31		Y	Y	Y	Y	N	Y
The allocate statement	P	Y	P	Y	N	Y	Y	Y	Y	N	Y
Assignment to an allocatable array	N	Y, 3	N	Y	Y	Y	Y, 2	Y	Y	N	Y,2
Transferring an allocation, 18	N	Y	N	Y	N	Y	Y	Y	Y	N	Y
More control of-access from a module	Y	Y	N	Y	N	Y	Y	Y	Y	N	Y
Renaming operators on the USE statement	Y	Y	Y	P	N	Y	Y	Y	Y	N	Y
Pointer assignment	Y	Y	Y	Y	N	Y	Y	Y	Y	N	Y
Pointer INTENT	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
The VOLATILE attribute	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
The IMPORT statement	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Intrinsic modules	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Access to the computing environment	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Support for international character sets	P, 19	P, 19	Y	Y	N	P	P, 19	Y	N	N	P,1 9
Lengths of names and statements	Y	Y	?	Y	Y	Y	Y	Y	Y	Y	Y
Binary, octal and hex constants	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Array constructor syntax	Y	Y	Y	Y	Y	Y	Y	Y	N	Y	Y
Specification and initialization expressions	Y	Y	Y	P	Y	Y	Y	Y	N	N	Y
Complex constants	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Changes to intrinsic functions	Y	Y	Y	P, 9	Y	Y	Y	Y	Y		Y

Fortran 2003 Features	**Absoft**	**Cray**	**g95**	**gfortran**	**HP**	**IBM**	**Intel**	**NAG**	**Oracle**	**Path scale**	**PGI**
Compiler version number	**14**	**8.4.0**		**5.2**		**13.1**	**16.0**	**6.0**	**8.7, 32**	**4**	**15.5**
Controlling IEEE underflow	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y
Another IEEE class value	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y
Input/output enhancements											
Derived type i/o	N	Y	N	N	N	Y	Y	N	N	N	Y
Asynchronous input/output	Y	Y	Y	Y,10	N	Y	Y	Y	Y, 10	N	Y
FLUSH statement	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y
IOMSG= specifier	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y
Stream access input/output	N	Y	Y	Y	N	Y	Y	Y	Y	N	Y
ROUND= specifier	Y	Y	P	Y	Y	Y	Y,20	Y	Y	N	Y
DECIMAL= specifier	Y	Y	Y	Y	Y	Y	Y,22	Y	Y	N	Y
SIGN= specifier	Y	Y	Y	Y	Y	Y	Y,21	Y	Y	N	Y
Kind type parameters of integer specifiers	N	Y	?	Y	N	Y	Y	Y	Y	N	Y
Recursive input/output	N	Y	Y	Y	N	Y	Y	Y	Y	Y	Y
Intrinsic function for newline character	N	Y	Y	Y	Y	Y	Y	Y	?	Y	Y
Input and output of IEEE exceptional values	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Comma after a P edit descriptor	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Interoperability with C											
Interoperability of intrinsic types	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Interoperability with C pointers	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Interoperability of derived types	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Fortran 2003 Features	**Absoft**	**Cray**	**g95**	**gfortran**	**HP**	**IBM**	**Intel**	**NAG**	**Oracle**	**Path scale**	**PGI**
Compiler version number	**14**	**8.4.0**		**5.2**		**13.1**	**16.0**	**6.0**	**8.7, 32**	**4**	**15.5**
Interoperability of variables	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Interoperability of procedures	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Interoperability of global data	Y	Y	Y	Y	Y	Y	Y	Y	Y	N	Y

Notes

Y	Yes
N	No
P	Partial
?	Not known
3	Can be disabled by an optional flag.
9	kind= of maxloc, minloc, shape missing
10	implemented as synchronous i/o
12	Suggested by Richard Maine
18	MOVE_ALLOC
19	SELECTED_CHAR_KIND only
20	plus RC,RD,RN,RP,RU,RZ
21	plus BLANK=,DELIM=,PAD=,SIZE=
22	plus DC,DP
30	only for output
31	Partial in the 4.9 development version
32	Included in Oracle Solaris Studio 12.4
33	Implemented as specified in the Fortran 2003 standard, not as in ISO TR 15581
34	Only kind type parameters

	Fortran 2008 Features	Absoft	Cray	g95	gfortran	HP	IBM	Intel	NAG	Oracle	Path scale	PGI
	Compiler version number	**14**	**8.4.0**		**5.2**		**15.1**	**16.0**	**6.0**	**8.7, 32**	**4**	**15.5**
2	**Submodules**	N	Y		N, 300	N	Y	Y	N	N	N	N
3	**Coarrays**	N	Y	P	Y, 301	N	N	Y	N	N	N	N
4	**Performance enhancements**											
4.1	do concurrent	N	Y		P	N	N	Y	N	N	N	N
4.2	Contiguous attribute	N	Y		Y	N	Y	Y	Y	N	N	Y
5	**Data Declaration**										N	N
5.1	Maximum rank maximum rank + corank <=15	N	Y		N	N	Y	Y	Y	N	N	N
5.2	Long integers (18 digit or 64 bit)	Y, 100	Y		Y	N	Y	Y, 100	Y			Y
5.3	Allocatable components of recursive type	N	Y		N	N	N	N	N	N	N	N
5.4	Implied-shape arrays	N	Y		Y	N	Y	N	Y	N	N	N
5.5	Pointer initialization	N	Y		Y	N	N	N	N	N	N	N
5.6	Data statement restrictions lifted		Y		N		N	Y	N		N	N
5.7	Kind of a forall index	N	Y		N	N	Y	N	N	N	N	N
5.8	Type statement for intrinsic types TYPE (intrinsic type) specifier		Y		Y		Y	N	Y		N	N
5.9	Declaring type-bound procedures		Y		Y		Y	Y	Y		N	N

	Fortran 2008 Features	Absoft	Cray	g95	gfortran	HP	IBM	Intel	NAG	Oracle	Path scale	PGI
	Compiler version number	**14**	**8.4.0**		**5.2**		**15.1**	**16.0**	**6.0**	**8.7, 32**	**4**	**15.5**
5.10-1 ,1	value attribute is permitted for any nonallocatable nonpointer noncoarray		Y		Y		P 108	Y				N
5.10-2	in a pure procedure the intent of an argument need not be specified if it has the value attribute		Y		Y			N				N
6, 2	**Data Usage**											
4.3, 3	Simply contiguous arrays rank remapping to rank>1 target	N	Y		Y	N	Y	Y	N	N	N	Y
6.1	Omitting an allocatable component in a structure constructor		Y		N		N	Y	Y		N	N
6.2, 1	Multiple allocations with source=		Y		Y		N	Y	N			N
6.3-1, 2	Copying the properties of an object in an allocate statement		Y		N, 209		Y	Y	Y		N	Y
6.3-2	MOLD= specifier for ALLOCATE		Y		Y		Y	Y	Y		N	N
6.3-3	copying the bounds of a source array in an allocate statement		Y		N, 210		Y	Y	Y		N	Y
6.4	Polymorphic assignment	N	Y		P, 211	N	N	N	Y	N	N	N

	Fortran 2008 Features	Absoft	Cray	g95	gfortran	HP	IBM	Intel	NAG	Oracle	Path scale	PGI
	Compiler version number	**14**	**8.4.0**		**5.2**		**15.1**	**16.0**	**6.0**	**8.7, 32**	**4**	**15.5**
6.5	Accessing real and imaginary parts	N	Y		N	N	Y	N	N	N	N	Y
6.6, 2	Pointer function reference is a variable	N	Y		P, 201	N	N	N	N	N	N	N
6.7	Elemental dummy argument restrictions lifted		Y		Y		N	N	N			N
7	**Input/Output**											
7.1	Finding a unit when opening a file	N	Y		Y	N	Y	Y	Y	N	N	Y
7.2	g0 edit descriptor	N	Y		Y	N	N	Y	Y	N	N	N
7.3	Unlimited format item	N	Y		Y	N	N	Y	Y	N	N	N
7.4	Recursive i/o	N	Y		Y	N	N	Y	Y	Y	Y	Y
8	**Execution control**											
8.1	The block construct	N	Y		Y	N	Y	Y	Y	N	N	N
8.2	Exit statement	N	Y		Y	N	Y	P	Y	N	N	N
8.3	Stop code	N	Y		Y	N	Y	Y	Y	N	N	Y
8.4, 2	ERROR STOP				Y			Y				
9	**Intrinsic procedures and modules**											
9.1	**Bit processing**											
9.1.1	Bit sequence comparison	Y	Y		Y	N	N	Y	Y	N	N	N
9.1.2	Combined shifting	Y	Y		Y	N	Y	Y	Y	N	N	N

	Fortran 2008 Features	Absoft	Cray	g95	gfortran	HP	IBM	Intel	NAG	Oracle	Path scale	PGI
	Compiler version number	**14**	**8.4.0**		**5.2**		**15.1**	**16.0**	**6.0**	**8.7, 32**	**4**	**15.5**
9.1.3	Counting bits	Y	Y		Y	N	Y	Y	Y	N	P, 212	P, 214
9.1.4	Masking bits	N	Y		Y	N	Y	Y	Y	N	N	N
9.1.5	Shifting bits	Y	Y		Y	N	Y	Y	Y	N	N	N
9.1.6	Merging bits	Y	Y		Y	N	Y	Y	Y	N	N	N
9.1.7	Bit transformat ional functions	N	Y		Y	N	N	Y	Y	N	N	N
9.2	Storage size	N	Y		Y	N	N	Y	Y	N	N	Y
9.3, 2	Optional argument radix added to selected real kind	N	Y		Y	N	Y	Y	Y	N	N	N
9.4	Extensions to trigonometric and hyperbolic intrinsic functions	Y	Y		Y	N	Y	Y	Y	N	N	Y
9.5	Bessel functions	Y	Y		Y	N	N	Y	N	N	N	Y
9.6	Error and gamma functions	Y	Y		Y	N	Y	Y	Y	N	P	Y
9.7	Euclidean vector norms	N	Y		Y	N	N	Y	Y	N	N	N
9.8	Parity	N	Y		Y	N	N	Y	Y	Y	N	N
9.9	Execute command line	Y	Y		Y	N	Y	Y	Y	N	N	N
9.10	Optional back argument added to maxloc and minloc	N	Y		N	N	Y	N	N	N	N	N
9.11	Find location in an array	?	Y		N	N	Y	N	N	N	N	N
9.12	String comparison		Y		Y		N	Y	Y		N	Y
9.13	Constants	N	Y		P, 203	N	Y	Y	Y	N	N	Y
9.14-1 , 4	COMPILER _VERSION	N	Y		N	N	Y	N	Y	N	N	N

	Fortran 2008 Features	Absoft	Cray	g95	gfortran	HP	IBM	Intel	NAG	Oracle	Path scale	PGI
	Compiler version number	**14**	**8.4.0**		**5.2**		**15.1**	**16.0**	**6.0**	**8.7, 32**	**4**	**15.5**
9.14-2, 4	COMPILER_OPTIONS	N	Y		N	N	Y	N	Y	N	N	N
9.15	Function for C sizeof		Y		Y		Y	Y				Y
9.16, 1	Added optional argument for ieee_selected_real_kind		Y		Y		Y	N				Y
10	**Programs and procedures**											
10.1	Save attribute for module and submodule data		Y		Y		Y	Y	Y		N	N
10.2	Empty contains section	Y	Y		Y	N	Y	Y	Y	N	N	N
10.3	Form of the end statement for an internal or module procedure		Y		Y		Y	Y	Y		N	Y
10.4	Internal procedure as an actual argument or pointer target	N	Y		Y	N	Y	Y	Y	N	N	N
10.5	Null pointer or unallocated allocatable as an absent dummy argument	N	Y		Y	N	Y	Y	Y	N	N	N
10.6	Non pointer actual for pointer dummy argument		Y		N			P	Y			N
10.7-1	generic resolution by procedureness	N	Y		N	N	Y	Y	Y	N	N	N
10.7-2	Generic resolution by pointer versus allocatable	N	Y		Y	N	Y	Y	Y	N	N	N

	Fortran 2008 Features	Absoft	Cray	g95	gfortran	HP	IBM	Intel	NAG	Oracle	Path scale	PGI
	Compiler version number	**14**	**8.4.0**		**5.2**		**15.1**	**16.0**	**6.0**	**8.7, 32**	**4**	**15.5**
10.8	Impure elelmental procedures	N	Y		Y	N	Y	Y	Y	N	N	Y
10.9	Entry statement becomes obsolescent	Y	Y		Y	N	Y	Y	Y	N	N	Y
11	Source form											
11.1	Semicolon at line start	Y	Y		Y		Y	Y	Y		Y	Y

Notes

1 New

2 Renamed

3 Moved from 4.3

4 Missing in earlier version

32 Included in Solaris Studio 12.4

100 INTEGER (KIND=8)

201 Implemented in 6.0

203 int and real, and coarray

209 Implemented in 6.0

210 gfortran via allocate but not via intrinsic assignment

212 Pathscale, counting bits, not trailz

213 Waiting for update from IBM.

214 leadz, popcnt, and poppar supported. No trailz

300 Supported in 6.0

301 Single image support since 4.6. Multi-image support using OpenCoarrays (including the Fortran 2015 collective subroutines) since 5.1, except allocatable or pointer components of derived type coarrays.

TS29113 Compiler conformance table	**Absoft**	**Cray**	**g95**	**gfortran**	**HP**	**IBM**	**Intel**	**Nag**	**Oracle**	**Path scale**	**PGI**
Compiler version number	**14.0**	**8.4.0**		**5.0**		**15.1**	**16.0**	**6.0**	**8.7**	**4.0**	**15.5**
TS conformant iso_fortran_binding.h	N	Y	N	N	N	Y	Y	N	N	N	N
CFI_CDESC_T macro	N	Y	N	N	N	Y	Y	N	N	N	N
CFI_* functions declared in iso_fortran_binding.h	N	Y	N	N	N	Y	Y	N	N	N	N
Assumed rank	N	Y	N	P	N	Y	Y	N	N	N	N
Assumed type or TYPE(*)	N	Y	N	Y	N	Y	Y	N	N	N	N
Pass scalar to TYPE(*) , DIMENSION(*)	N	Y	N	N	N	N	Y	N	N	N	N
Non-interoperable array for C_LOC C_F_POINTER	N	Y	N	N	N	N	Y	N	N	N	N
Non-interoperable function for C_FUNLOC C_F_PROCPTR	N	Y	N	N	N	N	Y	N	N	N	N
New semantics for ASYNCHRONOUS attribute	N	Y	N	N	N	Y	Y	N	N	N	N
RANK intrinsic function	N	Y	N	N	N	Y	Y	N	N	N	N
Changes to-SHAPE, SIZE, and UBOUND	N	Y	N	N	N	Y	Y	N	N	N	N
Assumed shape dummy arguments for BIND(C)	N	Y	N	N	N	N	Y	N	N	N	N
Allocatable dummy arguments for BIND(C)	N	Y	N	N	N	Y	Y	N	N	N	N
Pointer dummy arguments for BIND(C)	N	Y	N	N	N	Y	Y	N	N	N	N
Optional dummy arguments for BIND(C)	N	Y	N	N	N	Y	Y	N	N	N	N
Assumed-length character arguments for BIND(C)	N	Y	N	N	N	N	Y	N	N	N	N

Summary by compiler vendor

Fortran 2003											
	Absoft	Cray	g95	gfortran	HP	IBM	Intel	NAG	Oracle	Pathscale	PGI
	14	8.4.0		5.2		13.1	16	6	8.7, 32	4	15.5
Yes	39	56	33	44	31	57	57	56	51	23	57
Partial	0	1	3	8	0	1	1	1	0	0	1
No	16	0	17	6	26	0	0	1	6	34	0
Unknown	3	1	5	0	1	0	0	0	1	0	0
Sum	58	58	58	58	58	58	58	58	58	57	58
Fortran 2008											
Yes	12	65	0	45	0	42	48	46	2	2	20
Partial	0	0	1	2	0	2	2	0	0	1	0
No	34	0	0	12	47	20	13	15	44	53	42
Unknown	18	1	65	0	19	2	2	5	20	9	3
Sum	64	66	66	63	66	66	65	66	66	65	65
TS 29113											
Yes	0	16	0	0	0	11	16	0	0	0	0
Partial	0	0	0	0	0	0	0	0	0	0	0
No	16	0	16	16	16	5	0	16	16	16	16
Blank	0	0	0	0	0	0	0	0	0	0	0
Sum	16	16	16	16	16	16	16	16	16	16	16

Summary by feature

F2003						**F2008**						**TS 29113**				
Feature No.	Y	P	N	U		Feature No.	Y	P	N	U		Feature No.	Y	P	N	U
1	10	1	0	0		2	3	0	5	1						
2	11	0	0	0		3	2	1	6	0		1	3	0	8	0
3	4	1	5	0		4.1	2	1	6	1		2	3	0	8	0
4	10	0	1	0		4.2	6	0	3	1		3	3	0	8	0
5	6	0	4	0		5.1	4	0	5	1		4	3	0	8	0
6	7	0	3	0		5.2	3	0	1	5		5	3	0	8	0
7	9	0	2	0		5.3	1	0	8	1		6	2	0	9	0
8	7	0	3	0		5.4	4	0	5	1		7	2	0	9	0
9	7	0	3	0		5.5	1	1	7	1		8	2	0	9	0
10	7	0	3	0		5.6	2	0	5	4		9	3	0	8	0
11	9	0	2	0		5.7	2	0	7	1		10	3	0	8	0
12	5	1	4	0		5.8	4	0	3	4		11	3	0	8	0
13	7	0	3	0		5.9	5	0	2	4		12	2	0	9	0
14	6	1	3	0		5.10-1,1	3	1	1	6		13	3	0	8	0
15	8	0	3	0		5.10-2	2	1	0	8		14	3	0	8	0
16	7	0	2	2		4.3, 3	5	0	4	1		15	3	0	8	0
17	6	1	2	2		6.1	3	0	4	4		16	2	0	9	0
18	9	0	2	0		6.2, 1	3	0	3	5						
19	5	1	1	4		6.3-1,2	5	0	1	4						
20	8	1	2	0		6.3-2	5	0	2	4						
21	7	1	2	0		6.3-3	5	0	1	4						
22	7	0	3	0		6.4	2	0	6	1						
23	8	0	3	0		6.5	3	0	6	1						
24	8	1	2	0		6.6, 2	1	0	7	1						

F2003						**F2008**						**TS 29113**				
25	9	0	2	0		6.7	2	0	4	5						
26	11	0	0	0		7.1	6	0	3	1						
27	11	0	0	0		7.2	4	0	5	1						
28	11	0	0	0		7.3	4	0	5	1						
29	11	0	0	0		7.4	7	0	2	1						
30	11	0	0	0		8.1	5	0	4	1						
31	4	4	3	0		8.2	4	1	4	1						
32	10	0	0	1		8.3	6	0	3	1						
33	11	0	0	0		8.4, 2	2	0	0	9						
34	10	0	1	0		9.1.1	5	0	5	1						
35	8	1	2	0		9.1.2	6	0	4	1						
36	11	0	0	0		9.1.3	6	0	2	1						
37	9	1	0	0		9.1.4	5	0	4	1						
38	9	0	2	0		9.1.5	6	0	4	1						
39	9	0	2	0		9.1.6	6	0	4	1						
40	4	0	6	0		9.1.7	4	0	5	1						
41	9	0	2	0		9.2	5	0	4	1						
42	10	0	1	0		9.3, 2	5	0	4	1						
43	10	0	1	0		9.4	7	0	3	1						
44	8	0	2	0		9.5	5	0	5	1						
45	8	2	1	0		9.6	7	1	2	1						
46	10	0	1	0		9.7	4	0	5	1						
47	10	0	1	0		9.8	5	0	4	1						
48	6	0	3	1		9.9	6	0	4	1						
49	9	0	1	0		9.1	2	0	7	1						
50	9	0	0	1		9.11	2	0	7	1						
51	11	0	0	0		9.12	5	0	2	4						
52	11	0	0	0		9.13	5	0	3	1						

F2003						**F2008**						**TS 29113**				
53	11	0	0	0		9.14-1, 4	3	0	6	1						
54	11	0	0	0		9.15	5	0	0	6						
55	11	0	0	0		9.16, 1	4	0	1	6						
56	11	0	0	0		10.1	5	0	2	4						
57	11	0	0	0		10.2	6	0	4	1						
58	10	0	1	0		10.3	6	0	1	4						
						10.4	5	0	4	1						
						10.5	5	0	4	1						
						10.6	2	1	2	6						
						10.7-1	4	0	5	1						
						10.7-2	4	0	5	1						
						10.8	6	0	3	1						
						10.9	7	0	3	1						
						11.1	8	0	0	3						