

Title: Interval and IEEE Arithmetic

Submitted by: Germany (DIN)

Status: Being developed for 2000 Revision

References: N1148, S8, S9, INTLIB_90 (Kearfott), FORTRAN-XSC (Walter)

Basic Functionality: Provide a standardized interface to all features of the IEEE Standard for Binary Floating-Point Arithmetic (ANSI/IEEE 754-1985) and efficient support for an interval data type including interval operations, functions and I/O.

This includes the following tasks (not a complete list):

- Define a Fortran binding to IEEE primitives and features
- Introduce the abstract data type INTERVAL
- Require containment/enclosure of solutions (or solution sets)
- Support elementary arithmetic operations
- Support intrinsic functions of Fortran
- Keep it rather simple and unrestrictive
- Do not require high/optimal accuracy, except maybe as an option

Rationale: A growing community of engineers and scientists involved in scientific computing is recognizing the need for better hardware and software support for IEEE and interval arithmetic. Many years of experience with interval software have made it clear that a standard for the fundamental tools would greatly improve portability and efficiency.

Interval arithmetic provides a way for users to have confidence in their computations by calculating guaranteed enclosures of the solution or solution set of a problem. There are many applications that require the numerical reliability provided by interval methods.

IEEE conformance is supposed to be a property of a complete programming system (see first 3 sentences of the IEEE standard), but at present it is only a property of certain hardware processors (if any). To date, no scientific programming system achieves full compliance with the IEEE standard at the programmer's level.

Estimated Impact: Greatly improves portability and runtime speed of interval programs and numerical programs that require direct access to IEEE features such as arithmetic operations with directed roundings, setting the rounding mode, supporting special IEEE values (e.g. infinities, NaNs, ...), accessing the exception and trap flags, controlling the rounding during data conversion, and more.

Does not impact existing code, but existing interval code may be rewritten to conform to the newly standardized interface.

Detailed Specification: see S9 and documentation of INTLIB_90 (Kearfott) and FORTRAN-XSC (Walter)

History: A preliminary version of this requirement was submitted to X3J3 by William Walster and Keith Bierman from SUN on Aug. 29, 1995. The DIN working group for Fortran has made this a German requirement for the future development of Fortran. At the last international conference on Scientific Computing, Computer Arithmetic and Validated Numerics (SCAN 95) in September 1995, a group of more than 50 participants signed a petition to WG5 asking for the development of a standard for Interval Analysis in Fortran. Many experimental and several commercial packages for interval mathematics have been developed over the past 25 years.