The Next Fortran Standard

- Soon, in Order to
  - Give Users & Vendors Confidence that Investment in Fortran Software Is Sound & Has a Future
  - Maintain & Extend Fortran's Lead in Efficiency, Clarity-of-Expression, Ease-of-Use & Portability
- Examine Only Those New Features That May Be Addressed Quickly
  - Specifications & Syntax Done in About a Year
  - Keep the New Features Small & Independent

A Small Revision Is Due

- A Small Revision Might Be Able to
  - Address Some Features J3 Considered But Did Not Implement in F2000
  - Standardize Some Industry Standard Features
  - Address Some Portability Issues
  - Address Some Efficiency Issues
  - Address Some Ease-of-Use Issues
  - Add Some New & Extended Intrinsic Procedures
  - Add Some New & Extended Intrinsic Modules

Features J3 Did Not Do In F2000

- Initializers (Constructors)
- Automatic Attribute of Procedure Local Variables
  - Does an Object's Initializer or Finalizer Execute?
- Exit from Named Conditional Blocks
- Allow Non-nested Constructs of the Same Name
  - The Scope of the Name Should Be the Construct
- Return a Logical Unit Which May Be Opened

Industry Standard Features

- Do Until Loop Control
- HPF Ceiling Division & Ceiling Remainder
- A Way to Execute External Programs
- A Way to Flush a Unit Opened for Writing
- A Way to Seek a File Opened for Reading
- Variable Repeat Format Specifiers (*a la* VAX)
- A Way to Set File Buffer Sizes (etc.?)

Portability Features

- Maximum Integer a Real Kind Specifies Exactly
- Largest & Smallest Character Codes
- Selection of Logical Kinds (*How?*)
  - Add **`selected_log_kind()`**? *-or-*
  - A Logical Kind for Each Integer Kind ?
- Set Default Kind (via Implicit Statement?)
  - Mimic Today's -r8 & -i8 etc. Compiler Flags
- Generic Procedures (**`kind(*)`** ?)

Efficiency Features

- Swap the Values of Two Similar Objects
- Intent "Signature" & "Scratch" Arguments
- Pointers That Point Only to Pre-Specified Targets or Only to Protected or Intent(In) Targets
- Pointers That Point Only to Contiguous Storage
- Declare When Vector Indices Are Permutations
- Declare When a Set of Assumed Shape Dummy Argument Arrays Will All Have the Same Extents

Ease-of-Use Features

- Generic Dummy Arguments (**`type(*)`**)
  - May Appear <u>Only</u> in Interfaces
- A Way to Treat a Real Object as Complex (& *vv*)
- User Procedures of Indefinite Number of Similar (Same TK) Scalar Intent(In) Actual Arguments
- Local Variable in Place of a Non-Present Optional Argument & Specifying a Default Value
- Assertions for Debugging & Documentation

New & Extended Intrinsic Procedures

- Compute GCD & LCM (Two Integers)
- Compute Hypotenuse (Indefinite # of Real)
- Object Storage Size & Storage Availability
- Tell When a Pointer Points to Contiguous Storage
- Tell When a Vector Index Is a Permutation
- Time & Date Arithmetic (Use O/S Time Type ?)
- Indicate First Difference Between Two Strings
- Categorize a Character (UC, LC, digit, etc.)

New & Extended Intrinsic Modules

- A Use "All Names Except" Clause
- A New File Manipulation Intrinsic Module
- Add All Processor Dependent Values to the **`ISO_FORTRAN_ENV`** Intrinsic Module
- Put All Fortran Standard Specified Quantities in a New **`ISO_FORTRAN_STD`** Intrinsic Module
- Add Frequently Used Real Constants via a New **`ISO_MATH_CONSTANTS`** Intrinsic Module