NIMROD abstraction to enable development

Jacob King, Eric Howell, Scott Kruger (Tech-X)
Brian Cornille, Carl Sovinec (U. Wisconsin-Madison)
Eric Held, Andy Spencer (Utah State)

CTTS Meeting
Sherwood, April 22nd 2018
Auburn, AL

What is abstraction? Why should we do it?

With abstraction we can rewrite hard-coded blocks loop from this:

```
DO ibl=1,nrbl
  rb(ibl)%be%fs …
ENDDO
DO ibl=nrbl+1,nbl
  tb(ibl)%be%fs …
ENDDO
```

To this:

```
DO ibl=1,nbl
  bl(ibl)%be%fs …
ENDDO
```

Concrete type of bl is determined at runtime

- NIMROD is *already* designed/structured in an abstract way -- changes aren’t that radical
- Beyond cleaning up the code: plan to add new capabilities or optimization

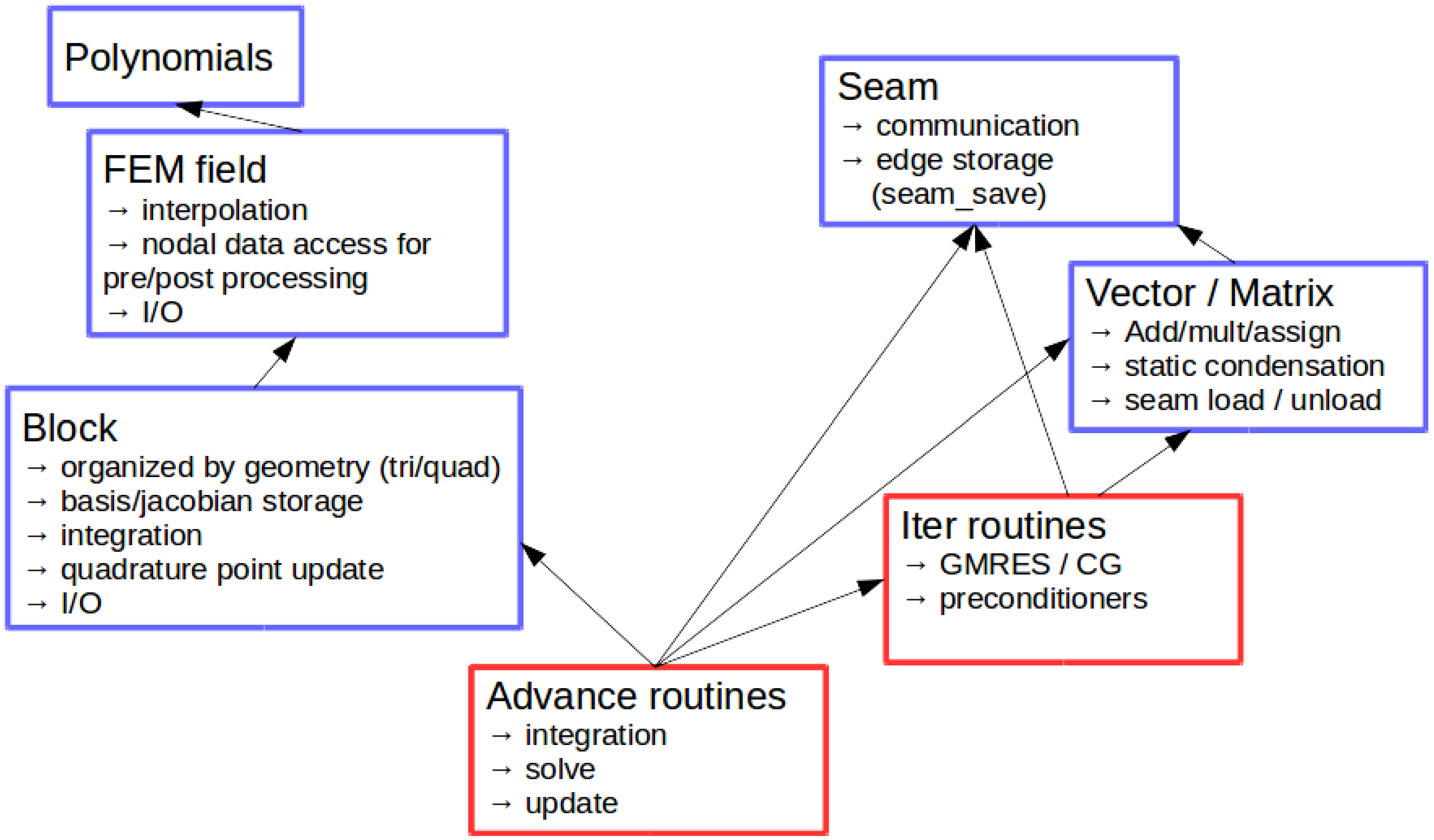
What are the goals of this effort?

- Goals are oriented towards physics objectives
- Optimization for continuum kinetic (CK) computations
  - Many CTTS CK-related planned -- NTMs, RWMs and energetic particles
  - Other CK-related efforts: RMPs, neoclassical flows/current
- Flexibility with meshing
  - E.g. tokamak calculation with structured closed-flux region grid coupled to unstructured grid that conforms to the limiter/divertor geometry
  - Potentially significant for disruption computations (VDEs, ideal and locked modes)
- Flexibility with basis functions
  - E.g. H(curl) and H(div) elements
  - Enable exploration of additional numerical methods

Fortran 90 or 2008/2013?

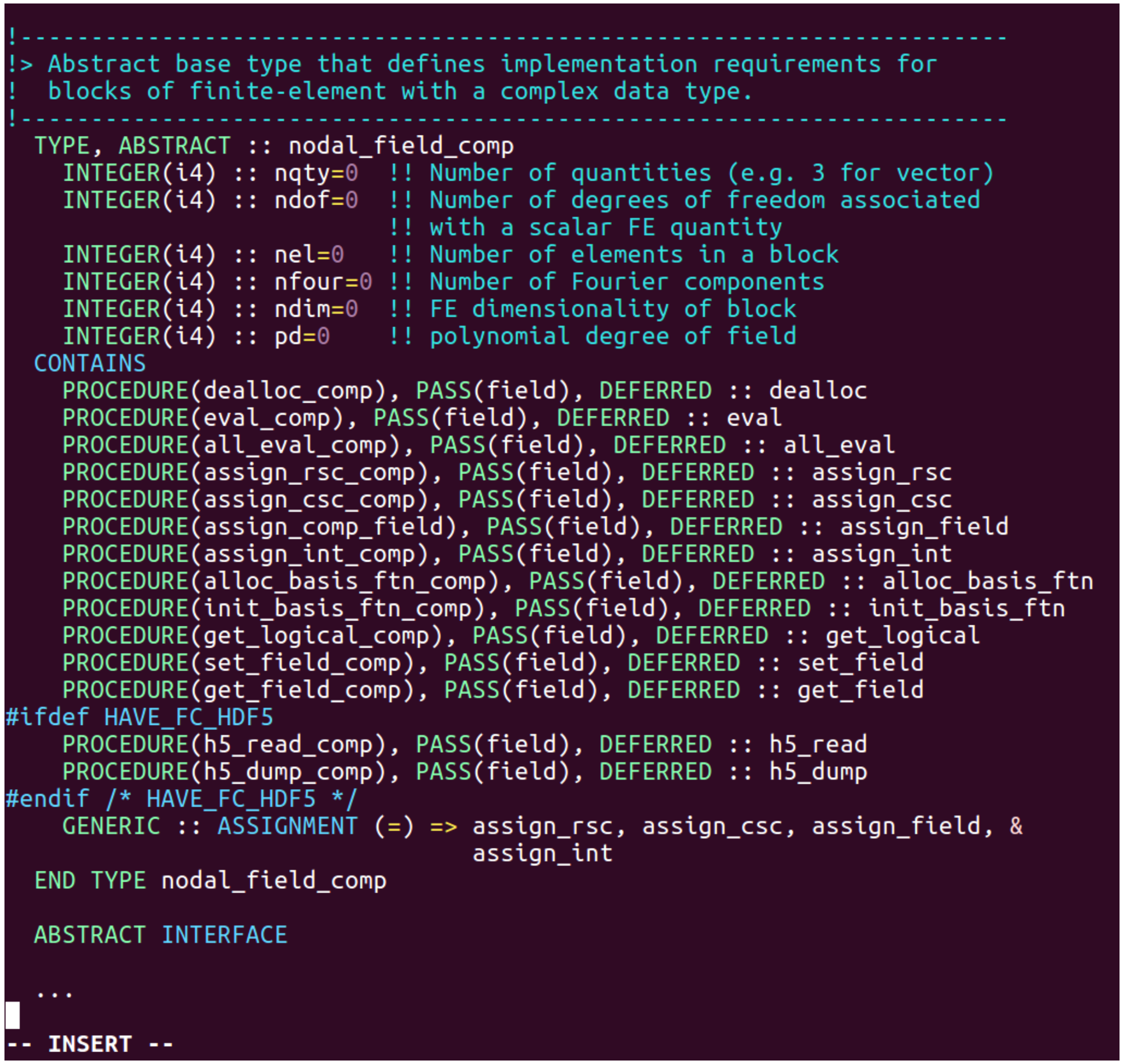
- Abstraction can be done in Fortran 90
- Fortran 2008/2013 adds useful object oriented functionality:
    - Blocks/Fields are cleanly described as self-contained objects
    - Use of abstract base class defines an interface that instantiations must conform to
    - “Program to interface, not an implementation”
    - Easier to maintain/extend/test code base
    - Consider an interface to “fields” (presently H1 lagrange_quad):
        - Basic functionality: interpolation of basis (evals)
            - Requires polynomial evaluation
        - Extended functionality: initialization (alloc/dealloc/set values at nodes)
- But don’t use object oriented program if not needed!
    - Abstract interfaces (callback) may be more appropriate
    - Present example: integrand routines
    - Planned example: generalized iterative solver routines (GMRES/CG)

Flow-chart overview of NIMROD design



Blue -- abstract types
Red -- abstract interfaces

What does an “abstract” interface look like?



```
!-----------------------------------------------------------------------
!> Abstract base type that defines implementation requirements for
!  blocks of finite-element with a complex data type.
!-----------------------------------------------------------------------
  TYPE, ABSTRACT :: nodal_field_comp
    INTEGER(i4) :: nqty=0  !! Number of quantities (e.g. 3 for vector)
    INTEGER(i4) :: ndof=0  !! Number of degrees of freedom associated
                           !! with a scalar FE quantity
    INTEGER(i4) :: nel=0   !! Number of elements in a block
    INTEGER(i4) :: nfour=0 !! Number of Fourier components
    INTEGER(i4) :: ndim=0  !! FE dimensionality of block
    INTEGER(i4) :: pd=0    !! polynomial degree of field
  CONTAINS
    PROCEDURE(dealloc_comp), PASS(field), DEFERRED :: dealloc
    PROCEDURE(eval_comp), PASS(field), DEFERRED :: eval
    PROCEDURE(all_eval_comp), PASS(field), DEFERRED :: all_eval
    PROCEDURE(assign_rsc_comp), PASS(field), DEFERRED :: assign_rsc
    PROCEDURE(assign_csc_comp), PASS(field), DEFERRED :: assign_csc
    PROCEDURE(assign_comp_field), PASS(field), DEFERRED :: assign_field
    PROCEDURE(assign_int_comp), PASS(field), DEFERRED :: assign_int
    PROCEDURE(alloc_basis_ftn_comp), PASS(field), DEFERRED :: alloc_basis_ftn
    PROCEDURE(init_basis_ftn_comp), PASS(field), DEFERRED :: init_basis_ftn
    PROCEDURE(get_logical_comp), PASS(field), DEFERRED :: get_logical
    PROCEDURE(set_field_comp), PASS(field), DEFERRED :: set_field
    PROCEDURE(get_field_comp), PASS(field), DEFERRED :: get_field
#ifdef HAVE_FC_HDF5
    PROCEDURE(h5_read_comp), PASS(field), DEFERRED :: h5_read
    PROCEDURE(h5_dump_comp), PASS(field), DEFERRED :: h5_dump
#endif /* HAVE_FC_HDF5 */
    GENERIC :: ASSIGNMENT (=) => assign_rsc, assign_csc, assign_field, &
                                 assign_int
  END TYPE nodal_field_comp

  ABSTRACT INTERFACE

  ...

-- INSERT --
```