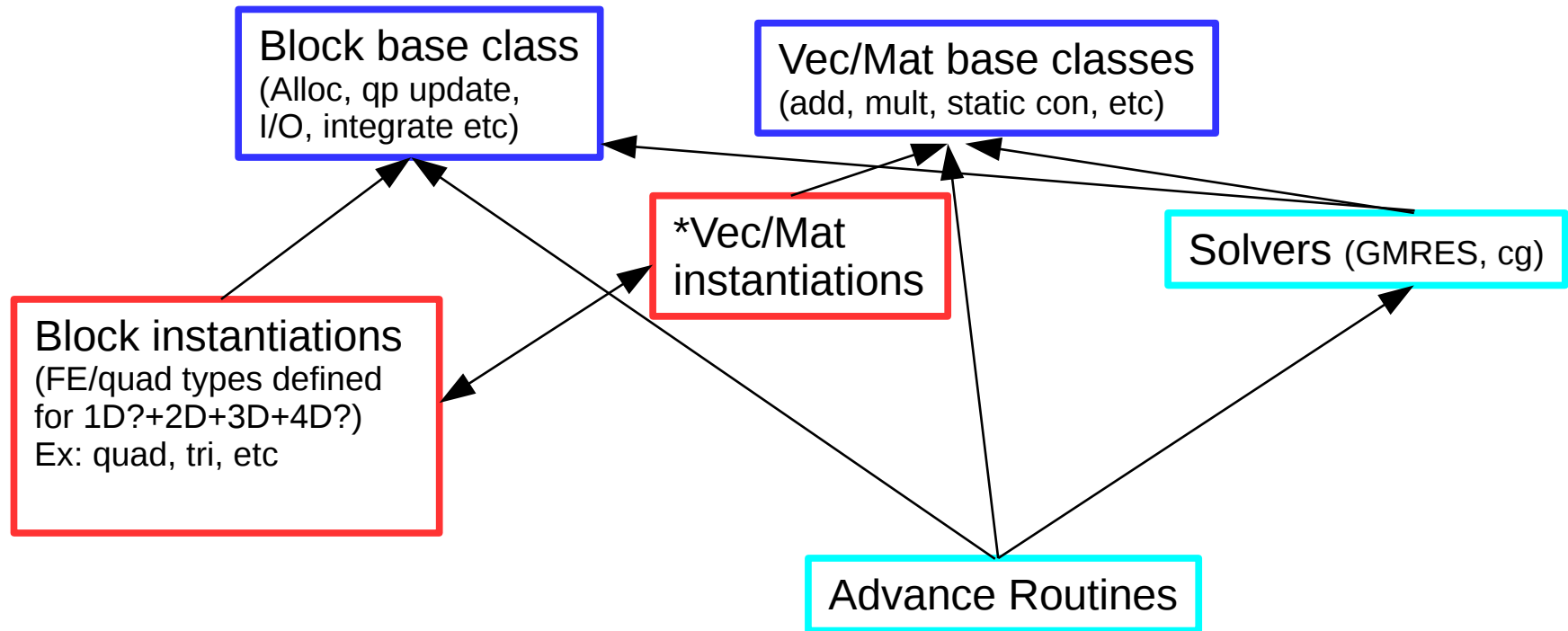


Discussion of NIMROD infrastructure

- CTTS objective: faster kinetic computations that include static condensation in phase space.
- Plan: implement a 3D FE field type and associated routines that follow the 2D FE structure (e.g. matrix/vector creation, `vector_type_mod`, `mat_vec`).
 - Effect of implementation: move pitch angle from “nqty” to “iz” or with quadrature points (“ig”).
 - Abstract types could make `iter_` routines agnostic to 2D or 3D finite elements.
- Is this a good time to address:
 - Abstract blocks and fields (synergistic with 3D FE fields)?
 - Vectorization?
 - Unit tests?
- Development should be incremental as much as possible (and tested as such).
- Dump-file compatibility should be maintained such that pre and post processing capabilities of existing code can be used (development focused on high computational-cost kernels).

NIMROD dependencies as abstract types



Open question: how could we deal with fields more efficiently?

At the same time as abstract block implementation: reorder arrays for vectorization

*Could be eliminated if we move from `arr{h,v,i}` to CSR format