

HOMME Coding Standards

HOMME Coding Standards

This page provides a minimal set of standards (requirements) placed on all code found within the HOMME. Since this list was put together well after HOMME code development began, not all HOMME code satisfies the below requirements. Existing HOMME routines will be brought into compliance over time. Modifications to existing or the addition of new code will be required to satisfy the below requirements. HOMME coding standards are expected to evolve over time and so users are encouraged to check back frequently. Users may also submit their coding standard suggestions to us.

- HOMME Coding Do's
 - Constants
 - Use globally defined constants from the constants module (physical_constants.f90), not local (routine specific) definitions.
 - Explicitly specify precision of local numeric constants (e.g., use 10.0_r_kind instead of 10.0)
 - Declarations
 - All variables must be explicitly typed - to ensure this all routines or modules must include "implicit none"
 - All module declarations must explicitly specify which module components (variables or routines) are used within the given routine (e.g., use constants, only: zero, one, two)
 - All integer and real variables must be typed using type definitions in the kinds module (kinds.f90)
 - The INTENT attribute must be specified for all arguments passed into and out of procedures (subroutines and functions)
 - Documentation
 - Document all changes, not only around modified code but also in the documentation block (doc block) at the head of each HOMME subroutine or function. Look at existing subroutines for examples.
 - Ensure input/output argument documentation in doc block is consistent with additions/deletions from calling list
 - Free format code
 - Maintain consistent and appropriate indenting of code. Use of the emacs F90 mode is encouraged.
 - Do not use all uppercase or mixed upper/lower case for code or comments. All uppercase may be used for comments when the intent is to draw the attention of readers to critical information regarding the code block in question.
 - Modules
 - All module contents should be explicitly declared public or private
- HOMME Coding Don'ts
 - Do not neglect, ignore, or violate the above "do's".
 - Do not use, *if at all possible*, machine (compiler) specific functions or calls. Doing so limits portability of the code.
 - Do not leave in code that was used for debugging purposes once the debugging is complete
 - Do not label user changes with comments that include the user's name or initials. For example, do not include a comment like this: "JPE change below" .