An example of a RK3 method used in WRF (Wicker and Skamarock, 2002)

$$\phi^{*} = \phi + \frac{\Delta t}{3} f(\phi)$$
$$\phi^{**} = \phi + \frac{\Delta t}{2} f(\phi^{*})$$
$$\phi^{+} = \phi + \Delta t \, f(\phi^{**})$$

- two level time stepping (from n to n+1)
- second evaluation centered (i.e. given at n+1/2)
- 3 evaluations of the RHS
- only the last RHS evaluation contains the vertical mixing.

Stage 1 ***starting***:

U* =Un ; ***e3****= e3n ; ***tra**** = tra_n

rhs of e3 and dynamics:

call compute_rhs_dyn(U*, tra*,e3*, **rhs_uv**, stage=1)
call compute_rhs_e3 (U*, e3* , **rhs_e3**)

time advance of e3 and dynamics

call advance_var(e3n, e3* , rhs_e3, **e3(**stage=1**)**)
call advance_var(Un , e3(n_stage), rhs_uv, **U** **(**stage=1**)**)

rhs & time advance of tracers

call compute_rhs_tra(U* , tra*, **rhs_tra**, n_stage=1)
call advance_var (tra(n), e3 , rhs_tra , **tra(n_stage)**)

Stage 2 ***starting***: *(including diagnostic of **w***)*

e3* = combine_dyn_e3(e3(n,1), , n_stage=2)
U* = combine_dyn_u (e3(n,1), u(n,1) , e3* , n_stage=2)
tra* = combine_dyn_tra(tra(n,1) , n_stage=2)

rhs of e3 and dynamics

call compute_rhs_dyn(U*, tra*, e3* , **rhs_uv**, n_stage=2)
call compute_rhs_e3 (U* , e3* , **rhs_e3**)

time advance of e3 and dynamics

call advance_e3 (e3n, e3*, ,rhs_e3, **e3(**stage=2**)**)
call advance_dyn(un , e3(n_stage),rhs_uv, **U** **(**stage=2**)**)

rhs and time advance of tracers

call compute_rhs_tra(U* , tra*, **rhs_tra**, n_stage=2)
call advance_tra (tra_n, e3(n,2) , rhs_tra , **tra(n_stage)**)

Stage 3 ***starting***: *centered in n+1/2 and including the diagnostic of **w***)*

e3* = combine_dyn_e3(e3(n,2) , n_stage=3)
U* = combine_dyn_u (e3(n,2), u(0,2), e3*, n_stage=3)
tra* = combine_dyn_tra(tra(n,2) , n_stage=3)

rhs dynamics and its vertical average

call compute_rhs_dyn(U*, tra*, e3* , **rhs_uv**, n_stage=3)
rhs_u_ave = average(rhs_uv)

Integrate ssh & e3

call advance_ssh(rhs_uav, **ssh(n+1), u2d(n+1/2), u2d(n+1)**)
call compute_e3(ssh(n+1) **, e3(**n_stage=3**)**)

time advance of dynamics *(including mixing tendency)*

call advance_dyn(U(1), e3(n_stage=3), rhs_uv, **U(**n_stage=3**)**)
call correct_vertical_avergage(u2d(n+1) , **U(**n_stage=3**)**)

rhs and time advance of tracers *(including mixing tendency)*

call correct_vertical_avergage(u2d(n+1/2), **U***)
call compute_rhs_tra(U*, tra*, **rhs_tra**, n_stage=3)
call advance_var (tra_n, e3 , rhs_tra , **tra(**n_stage=3**)**)

Various Remarks and questions:

• Forward-Backward Scheme (FBS)
==>> straightforward simplification of above algorithm
requires the viscosity to be large (low resolution ocean)

• Requires, for all TRA and DYN tendency routines :
==>> pass velocity and/or T-S together with e3 in argument

• How to deal with adaptive, courant-dependent implicit vert. adv.
==>> implicit inversion at each stage ???

• How to deal with implicit top and bottom friction
==>> adaptation of Jérome's idea to variable e3 ?

• AGRIF compatibility
==>> swap from n to n+1 all done at the very end, sufficient ?

• z-tilda coordinate
==>> applied at all step (w* calculation) or only the last one?

• Do we all agree to remove Vector Form (VF)
==>> only available in Flux Form (FF)

pro
- significant reduction of the code
- e3t, u, v no more surface weighted quantities, simple average
- A Compensated Time-Space scheme (CTS) will need FF.
- monotonic momentum advection will also need need FF.

con
- only energy conserving scheme
- better shape of baroclinic instability in a channel….
- …

• Coriolis: FF, f-parameter defined at T-point. but issue with
its vertical average with step or partial step topography
(issue that exist also with Coriolis defined at F-point.

• Passive tracers: not inside the RK3 framework possible ?
==>>> single time integration (with possibly time coarsening)

• Pressure gradient : for sco, mix sco-zps and z-tilda coordinate. :
==>> pressure gradient using finit volume form ?

Proposed implementation strategy:

Initial target :

RK3 with WRF choice (known to work, but probably not optimal) as separate directory

preparation phase : (shared with the dev_merge_2017 (DM2017))

- Coriolis at T-point (Done and ready for DM2017)
- pass in argument U,T, e3 in all DYN/TRA (Done, and verified one by one in LF case. Put this in DM2017 ?)
- time-integration of (e3t T) (shared with DM2017)
- dynspg_ts : split of 2Δt/Δt calculation (shared with DM2017?)

RK3 directory only:

- suppress the Vector Form (share with DM2017 or not ?)
- implement and test
- Surface Module : forcing provided at n+1/2 only. adjust the sea-ice and iceberg time-step.

Work in parallel :

// • Adaptive Courant-number-dependent implicit vertical advection: 1st in MLF case, then try to find an optimize solution for RK3 needed to obtain the largest possible time-step.
// • dynspg_ts : move to a RK3 framework (cf Laurent/Rachid work)
// • dynspg_ts : add momentum non-linear terms ?
// • hpg : finite volume pressure gradient consistent with TEOS10
// • Coriolis : test the effect of a clean removal of barotopic Coriolis trend for the 2D RHS of external mode with step-like topography.
// • determine the optimal RK3 choice.

Second stage :

- incorporate the work done in //
- adaptation to passive tracer
- AGRIF compatibility
- and a lot of tests…