NEW PRESENTATION FOR OBSHOR OBSERVATION INTERPOLATOR.

YESSAD Karim.

September 21, 2011

Version 4.

1 Introduction and purpose.

Observation horizontal interpolator is done under routines **COBSALL**. Horizontal interpolations are done on upper air fields but also on some surface fields.

In March 2009, Ludovic Auger (CNRM/GMAP) pointed out difficulties to introduce new surface fields in this interpolator. Significant cleanings have been done in CY37 and CY38, and we focus on work remaining to be done.

We can briefly sum-up the remaining issues of the current code:

- There is currently no checking that a surface field which is allocated in the GOMS structure, is a field which is allocated in the surface dataflow.
- Structure of GOMS surface fields does not match the structure of surface fields, and because of that it is difficult to introduce new surface fields in the GOMS dataflow. Apart from the GOMS structures issues themselves, there are features in the **MPOBSEQ..** routines which make very difficult to introduce new surface fields.
- There is currently no possibility to call the observation interpolator without calling **GETDB** nor **PUTDB** (missing an idealised configuration with an idealised list of interpolation points defined in an ARPEGE/IFS set-up routine, no call of ODB routine at all).

The basis of this study is cycle CY38.

2 Organigramme of the pieces of code to be adapted.

Not detailed. All pieces of code are under routines **COBSALL**, **COBSALLTL** and **COBSALLAD**. Additionally additional checkings are required in **SUGOMS**.

3 Surface fields: store in GOMS fields which have been allocated in the surface dataflow.

In cycle CY38, there is no checking that a surface field stored in the GOMS structure has been allocated in routine **SU_SURF_FLDS**. There are now logical variables saying if surface variables are active.

Proposal:

- In routine **SUGOMS**, use attributes of YSP_ACT_[group] and YSD_ACT_[group] to allocate surface GOMS (we should not allocate surface GOMS for fields which have not been allocated in **SU_SURF_FLDS**).

Target: CY39. This work must be preferably done at ECMWF to avoid merging issues.

4 Halo management aspects.

All necessary cleanings have been completed in CY38; nothing remaining to do.

5 GOMS structures and MPOBSEQ... routines features issues.

In cycle CY38, structure of GOMS surface fields does not match the structure of surface fields, and because of that it is difficult to introduce new surface fields in the GOMS dataflow. Apart from the GOMS structures issues themselves, there are features in the **MPOBSEQ..** routines which make very difficult to introduce new surface fields.

Work to solve GOMS structures and **MPOBSEQ...** routines features issues can be shared into two steps.

- Some preliminary cleanings in the **MPOBSEQ...** routines. At this stage we do not change the GOMS structures themselves. This work is designed to reduce the risks of bugs and to make easier future developments in the **MPOBSEQ...** routines.
- Rewrite GOMS structures, in order that GOMS match model structures. This is the case in particuliar for constants, surface fields and model errors statistics where model structures and GOMS structures are completely different.

Proposal for first step (details are given for the direct code):

- Put paragraph headers in the different **MPOBSEQ...** routines to separate the different groups of variables:
  - Upper-air variables (GMV, GFL): fill or read YGOM%YUA and YGOM%YUA_2D.
  - GMVS variables, GFLS (=surface GFL) variables, constants, surface variables: fill or read YGOM%YS and YGOM%YS_2D, YGOM%YEC and YGOM%YEC_2D.

  - Surface variables and model errors statistics used in CANARI: fill or read YGOM%YCANA and YGOM%YCAN.

- Add a new attribute L[variable] in types TYPE_GOMS, TYPE_GOMS_2D, TYPE_GOMEC, TYPE_GOMEC_2D, TYPE_GOMCAN, TYPE_GOMCAN_ALT, as it is already done in TYPE_GOMUA (routine **GOMS_MIX**). This attribute says if the corresponding GOM variable must be written/read. Set this attribute in **SUGOMS** to the right value. If possible, for surface fields, use the new variable introduced in **SU_SURF_FLDS** and **SURFACE_FIELDS_MIX**, in order that the allocated GOMS surface fields are also allocated ones in **SU_SURF_FLDS**.
- In **MPOBSEQ_PACK** and **MPOBSEQ**, memory transfers must be rewritten in order to be done according to the new attribute Y(D)GOM..(INUM)%L[var] and no longer according to tests using variables like LECMWF, LCANARI, LCASIG, etc... Only key LDMVO5 can still be used in the **MPOBSEQ...** routines. Management of index KBP must be done step by step (add 1 or NFLEVG to KBP after each memory transfer); features like KBP=KBP+8 or PBUFS(KBP+12) should be completely removed. For more details about 1-level fields (ignoring key LDMVO5 to simplify):
  - in **UNPACK_RECV**, memory transfers must look like:

```
IF (Y(D)GOM..(INUM)%L[var]) THEN
  Y(D)GOM..(INUM)%[var]=ZBUFR(KBP)
  KBP=KBP+1
ENDIF
```

  - in **PACK_SEND**, memory transfers must look like:

```
IF (Y(D)GOM..(INUM)%L[var]) THEN
  PBUFS(KBP)=PBF(IOBS,YOBB1%MPTR[var])
  KBP=KBP+1
ENDIF
```

- At the end of this work the **MPOBSEQ..** routines should not use any longer variables like LECMWF, LCANARI, LCASIG, LSOLV, LDIRCLSMOD, LOBSTL.
- Calculation of local variable IBUFLENP done in the different **MPOBSEQ...** routines must be moved in a new subroutine **SET_NBUFLENP** (with an option allowing to compute the IBUFLENP required in the adjoint code). In a first stage of the work we can call **SET_NBUFLENP** in the **MPOBSEQ...** routines; later this calculation can be done from a set-up routine (call **SET_NBUFLENP** in **SUGOMS**). **SET_NBUFLENP** must use the attributes Y(D)GOM..(INUM)%L[var] and no longer variables like LECMWF, LCANARI, LCASIG, LSOLV, LDIRCLSMOD, LOBSTL.
- A similar work must be done in the TL and adjoint code.

Proposal for second step: split GOMS according to the current model data structures.

- GMV and GFL: attribute YGOM%YUA can be kept unchanged, or split it into YGOM%YGMV and YGOM%YGFL.
- GMVS: create attribute YGOM%YGMVS and use it instead of YGOM%YS.
- GFLS: create attribute YGOM%YGFLS and use it instead of YGOM%YS.
- Constants: create attribute YGOM%YCONST and use it instead of YGOM%YS or YGOM%YEC.
- Surface fields: create attribute YGOM%YSURF (or even split it into YGOM%YSP for prognostic surface fields and YGOM%YSD for diagnostic surface fields) and use it instead of YGOM%YS, YGOM%YEC and YGOM%YCAN. Introduce the different groups (SOILB, VDIAG, VARSF, etc..) inside YGOM%YSURF.
- Model errors statistics for CANARI: create YGOM%YMESU (3D fields) and YGOM%YMES (2D fields) and use them instead of YGOM%YCANA and YGOM%YCAN.
- These GOM buffers must have 2D-versions appended by _2D.
- Remove the old GOM structures.
- Root names to be retained for constants, surface fields and model errors statistics: use the same roots everywhere for fields where there is no possible ambiguity, according to a standard naming given in documentation (IDVNAM). For example use root [OROG] everywhere for orography.

Target:

- First step: could be done for CY39 (preferably at ECMWF, to avoid severe merging issues).
- Second step: no target given, ECMWF is more experienced to do that. A lot of side effects are expected (several tens of routines use GOMS structures) and that may be a huge amount of work. Priority depends on the number of new surface fields expected to enter the GOMS in the next following years; if this number is no more than 5 and if there is no ambiguity about the choice YGOM%YS/YGOM%YEC/YGOM%YCAN, the current GOMS structure can still be used; on the contrary if several tens of surface fields are expected to enter the GOMS, with an impossibility to choose among YGOM%YS/YGOM%YEC/YGOM%YCAN, action to use new GOMS structures will become stringent.

  An action must be done to list the potential additional surface fields which may enter the GOMS in the next ten years (2011-2020), and which team will contribute (METEO-FRANCE, ECMWF, ALADIN partners, HIRLAM partners).

6 Specific adjoint code issues.

The adjoint code is less flexible than the direct code about the choice of interpolations. It uses optimisations, which currently assume that all fields use the same interpolator (contrary to what is done in the direct and TL codes). It is necessary to know the level of compromise between flexibility and optimisations to decide how to write properly this piece of code (and in particuliar **SLINTAD**).

The questions which should be answered are:

- in the next years, do will stick to interpolate all the fields with the same interpolator, or do we want to make possible the use of different interpolators?
- is it important for optimisation aspects to do one call to **LAIDDIOBSAD/LAIDLIOBSAD** for all fields, or can we allow to do this unique call for upper-air fields, and to have individual interpolations for surface fields?
- is it expected to have a subset of fields which must be interpolated with different weights (taking account of the land-sea mask for example)?

According to the answers to these questions, the way to write **SLINTAD** and to allocate PB1 will not be exactly the same one.

Assuming that we keep the same interpolator for all fields and one call to **LAIDDIOBSAD/LAIDLIOBSAD** interpolating all fields, there is nothing more to do in **SLINTAD** for the time being, after the work which entered CY38. Some comments and checkings giving some restrictions of use (part 3 of **SLINTAD**) must be checked to see if they are still relevant.

7 Test configuration to validate the observation interpolator.

This is out of the scope of this paper but it would be desirable as soon as possible to have a test configuration, not calling ODB routines, using an idealised list of interpolation points defined in an ARPEGE/IFS set-up routine, to test the observation interpolator (direct code, and also if possible TL and AD codes).

8 Conclusion and perspectives.

We have tried to list a series of separate tasks remaining to do after what already entered CY38. Some of them can be undertaken as soon as possible, to enter CY39. The less stringent (and most difficult task) is to rewrite the GOMS structures, and a working plan must be known about the new surface fields which will enter the GOMS in the next ten years.

9 References.

- (IDVNAM) Yessad, K., 2011: recommended variable naming in ARPEGE/IFS. Internal note.