RESEARCH DEPARTMENT MEMORANDUM

To:	RD Scientific Staff and Consultants	
Copy:	DR, DF, HPS, HES, Alain Joly, John Hodkinson, François Bouyssel, Claude Fischer, Ryad El Khatib, Karim Yessad	
From:	Deborah Salmond et al.	
Date:	November 19, 2015	File: RD15-264
Subject:	**IFS Memorandum Cycle CY42R2**	

Cycle 42r2 was created in September 2015. This is an 'OOPS-refactoring' only cycle.

Contributors:

Alan Geer, Deborah Salmond, Gabor Radnoti, Mike Fisher, Olivier Marsden, Peter Lean, Ryad El Khatib, Tomas Wilhemsson, Yannick Trémolet

Merged branch names:

V0: das_CY42R1_NEW

nat_CY42R1_OOPS

dai_CY42_Jb_cleanup_2

dai_SB42R1_Jb_cleanup_3

V1: das_CY42R1_NEW_V1

stg_CY42R1_pre_oops

V2: das_CY42R1_NEW_V2

dipl_SB42R1_odb_view_objects

das_SB42R1_Ryad

nat_CY42R1_mkglobstab

stg_SB42R1_pre_oops_3

stg_CY42R1_fix_bgobs

dai_SB42R1_NEW_V1_Jb_cleanup_4

V3: das_CY42R1_NEW_V3

dipl_SB42R1_odb_mdb_colnames

V4: das_CY42R1_NEW_V4

Olivier Geometry SPAM script

V5: das_CY42R1_NEW_V5

stg_SB42R1_sha_hop_driver

day_SB42R1_V4_oops

dag_CY42R1_esuite (25/09)

dipl_CY42R1_fix_warm_report_status

nat_CY42R2_spectral_arp_fix

Notes:

CY42R2 contains updates from dag_CY42R1_esuite upto 25th September 2015

Yannick Trémolet - day_SB42R1_V4_oops

Upgrade of interfaces for OOPS

The main modification in this branch is the deletion of the oops project for the perforce repository. The code in that project is now kept in git on the stash repository (ssh://git@software.ecmwf.int:7999/oops/oops-ifs.git). The scripts to build the OOPS executables have been modified accordingly.

The branch also contains updates of the fortran side of the interfaces with OOPS and some refactoring of the handling of the background and first guess in the Jb computation.

Files created(IFS):
`oops/allobs_oper_mod.F90 odb_setup.F90`
`var/sujbwavelet_stdevs.F90 suscal_jb.F90`

Files created(SCRIPTS):
`sms/oopsbuild.sms`

Olivier Marsden - das_CY42R1_NEW_V4

Application of the SPAM script to geometry-related variables

For the OOPS project, any data used by the IFS whose value is not fixed at compile time should be passed to routines by argument. This goal requires the modification of almost every routine contained in the IFS source. Following on from the work done in CY42R1 for the derived types contained in the FIELDS type, the SPAM script has been applied again to deal with GEOMETRY-related derived types, specifically TDIM, TDIMV, TGEM and TMP. Instances of these data types may no longer be accessed in routines by USE statements, *i.e.* statements like `USE YOMDIM, ONLY YRDIM` are no longer to be used; instead, they are replaced by passing an argument of the appropriate type explicitly to the routine. For routines which require two or more of these arguments, a single argument of type GEOMETRY is substituted. The remaining derived types in the GEOMETRY type will be dealt with in the same way for CY42R3.

Peter Lean - dipl_SB42R1_odb_view_objects

Introduce ODB view objects into observation operator

The original IFS/ODB interface used global variables to store the observation data returned from an SQL query in ODB. As the first step of a wider refresh of the IFS/ODB interface, this branch introduces new ODB view derived types to store the observation data and metadata (e.g. column names) to the observation operator code. In addition, rather than being global variables, these view objects are now passed by argument to the subroutines where they are used.

Files created(IFS):
`op_obs/aeolus_getamd.F90 hop_decide_required_sqls.F90`

Files created(ODB):
`include/compat_fill_mdb_col_array_members.h compat_fill_mdb_members.h compat_fill_mdb_table_array_members.h compat_mdb_col_array_members.h compat_mdb_members.h compat_mdb_table_array_members.h odb_assoc_cols.h odb_it_members.h`

module/odb_interface.F90

Files modified(IFS):

module/varno_module.F90

mwave/mwave_emis.F90 mwave_get.F90 mwave_get_ad.F90 mwave_get_tl.F90
mwave_obsop.F90 mwave_obsop_ad.F90 mwave_obsop_test.F90 mwave_obsop_tl.F90
mwave_put.F90 mwave_put_tl.F90 mwave_screen.F90 mwave_wrapper.F90

op_obs/amv_get_preds.F90 amv_oberr.F90 amv_reassign.F90 bgobs.F90 ch4bcor.F90
dopplsim.F90 dopplsim_ad.F90 dopplsim_tl.F90 exheiz2p.F90 exheiz2p_lidar.F90
gpsro_2dad.F90 gpsro_2dop.F90 gpsro_2dtl.F90 gpsro_ad.F90 gpsro_oberror.F90
gpsro_op.F90 gpsro_tl.F90 hdepart.F90 hinth.F90 hjo.F90 hop.F90 hopad.F90
hoptl.F90 hqscatt.F90 hradp_ml.F90 hradp_ml_ad.F90 hradp_ml_tl.F90 hretr.F90
hretr_aeolus.F90 hsatang.F90 inv_refl1dstat.F90 mw_clearsky_screen.F90
mw_clearsky_screen_ecdecis.F90 mw_clearsky_screen_mfdecis.F90
obsop_composition.F90 obsop_rad.F90 obsop_varbc.F90 rad1cemis.F90 rad1cobe.F90
reflsim.F90 reflsim_2dop.F90 reo3bcor.F90 rtl_hop_1d.F90 rtl_hop_1d_ad.F90
rtl_hop_1d_tl.F90 rtl_hop_2d.F90 rtl_hop_2d_ad.F90 rtl_hop_2d_tl.F90
rtl_oberror.F90 rtl_screen.F90 sat_avg_stdev_filter.F90

pp_obs/ppobsacad.F90 ppobsactl.F90 ppobsap.F90

Files modified(ODB):

ddl/varno.h

module/odbio_msgpass.F90

Files deleted(IFS):

module/aeolus_getamd_mod.F90 aeolus_processing_mod.F90

Peter Lean - dipl_SB42R1_odb_mdb_colnames

Refer to ODB columns by column name in new IFS/ODB interface

In the original IFS/ODB interface, output from ODB was stored in 2d arrays, and the MDB indexes were used to refer to specific ODB columns. However, the MDB index names were often very different from the ODB column name, making the code unnecessarily hard to understand. e.g. MDBVAR describes the index of the obsvalue@body ODB column in the 2d array.

As part of the ongoing IFS/ODB interface refresh (currently confined to the observation operator code), the names used to refer to ODB columns have been updated so that the MDB index name is always identical to the ODB column name. For example, MDBVAR will change to OBSVALUE_AT_BODY. This should make it a little easier to understand what the code is doing.

Files modified(IFS):

control/cnt4.F90

module/varbc_setup.F90 varbc_to3.F90

mwave/mwave_get.F90 mwave_get_ad.F90 mwave_get_tl.F90 mwave_put.F90
mwave_put_tl.F90

op_obs/aeolus_getamd.F90 amv_get_preds.F90 amv_oberr.F90 amv_reassign.F90
ch4bcor.F90 dopplsim.F90 dopplsim_ad.F90 dopplsim_tl.F90 exheiz2p.F90
exheiz2p_lidar.F90 gpsro_2dad.F90 gpsro_2dop.F90 gpsro_2dtl.F90 gpsro_ad.F90
gpsro_oberror.F90 gpsro_op.F90 gpsro_tl.F90 hdepart.F90 hinth.F90 hjo.F90
hop.F90 hopad.F90 hoptl.F90 hqscatt.F90 hradp_ml.F90 hradp_ml_ad.F90
hradp_ml_tl.F90 hretr.F90 hretr_aeolus.F90 hsatang.F90 inv_refl1dstat.F90

mw_clearsky_screen.F90 mw_clearsky_screen_ecdecis.F90
mw_clearsky_screen_mfdecis.F90 obsop_composition.F90 obsop_rad.F90
obsop_varbc.F90 rad1cemis.F90 rad1cobe.F90 reflsim.F90 reflsim_2dop.F90
reo3bcor.F90 rtl_hop_1d.F90 rtl_hop_1d_ad.F90 rtl_hop_1d_tl.F90 rtl_hop_2d.F90
rtl_hop_2d_ad.F90 rtl_hop_2d_tl.F90 rtl_oberror.F90 rtl_screen.F90
sat_avg_stdev_filter.F90

phys_ec/cumastrn.F90

pp_obs/ppobsacad.F90 ppobsactl.F90 ppobsap.F90

var/congrad.F90

Files modified(ODB):

include/compat_fill_mdb_col_array_members.h compat_fill_mdb_members.h
compat_fill_mdb_table_array_members.h compat_mdb_col_array_members.h
compat_mdb_members.h compat_mdb_table_array_members.h

module/odb_interface.F90

Files modified(WAM):

Wam_oper/meansqs.F sbottom.F setwmask.F

Peter Lean - dipl_CY42R1_fix_wam_report_status

Fix inconsistency in wave observation pre-processing

Fixes a bug which could potentially casue corrupt CCMA databases.

A single observation report can contain observations for several variables. Separate report_status and datum_-status flags describe which observations are active,rejected,blacklisted etc. If all observations for one report are rejected, then the report_status flag should also be set to rejected.

However, in some situations, the wave observation pre-processing could leave the report_status as active even if all corresponding datum_status flags were rejected. This could potentially result in a corrupt CCMA database. This branch fixes the bug.

Files modified(WAM):

Wam_oper/rfl4wam.F90

Tomas Wilhelmsson - nat_CY42R1_OOPS

Further OOPS refactoring

Further OOPS refactoring to enable multiple resolution forecasts in one executable.

Files created(IFS):

module/yomoph0.F90

setup/suoph0.F90

Files modified(ALGOR):

module/control_vectors_data_mix.F90 control_vectors_oper_mod.F90 jb_control_vectors_-
oper_mod.F90 spectral_arp_mod.F90 spectral_fields_data.F90 spectral_fields_mod.F90
spectral_fields_oper_mod.F90 spectral_fields_para_mod.F90

Files modified(IFS):

c9xx/cseaice.F90 csstbld.F90

canari/caeincw.F90 caisse.F90 calice.F90 canife.F90 caohis.F90

climate/updcli.F90 updcli_mse.F90

control/cfcsens2obs.F90 cnt0.F90 cnt3_wait.F90 cnt3ad.F90 cnt4.F90 cprep1.F90 forecast_error.F90 get_clinc.F90 reresf.F90 scan2m.F90 scan2mtl.F90 stepo.F90 stepo_oops.F90

dia/chkevo.F90 inifaoutinfo.F90 posddh.F90 prepfdb.F90 suofname.F90 suppdate.F90 wrbudg.F90 wrmlppa.F90 wroutgpgb.F90 wroutspgb.F90

fullpos/predynfpos.F90

io_serv/io_serv_suiosctmpl.F90

module/factx_mod.F90 fields_mod.F90 gmv_subs_mod.F90 grib_utils_mod.F90 model_mod.F90 traj_main_mod.F90 yomgrib.F90 yomoph.F90

obs_preproc/mkglobstab.F90

ocean/wrcom.F90

oops/error_covariance_3d_mod.F90 fields_interp_mod.F90 fields_io_mod.F90 ifs_init.F90

parallel/read_spec_fromfa.F90

phys_ec/suphec.F90 wvcouple.F90

setup/su0yomb.F90 suarpio.F90 sueframe.F90 sufdb.F90 sugrib.F90 sugridg.F90 sugrido.F90 sugridug.F90 sugridug2.F90 sump.F90 sump0.F90 suoph.F90 suoptproma.F90 susc2c.F90 suspecb.F90 suspecg.F90 suspecg2.F90

sinvect/lcztoald.F90 lcztoifs.F90 sptrlcz.F90

utility/openfa.F90 openfainfo.F90 pkgrida.F90 pkspeca.F90 pksurfa.F90 prt_ctlvec_max.F90 prt_ctlvec_norms.F90 random_ctlvec.F90 read_surfgrid_traj_fromfa.F90 reset_accfie_vareps.F90 save_merr_tend.F90 savmoderr.F90 sbsfgs.F90 sualspa1.F90 sualspajb.F90 write_ctlvec_grib.F90 wrresf.F90

var/adtest.F90 cvargpad.F90 cvargptl.F90 litest.F90 read_surfgrid_traj.F90 sualctv.F90 suallr.F90 suallt.F90 suecges.F90 sujq.F90 sumoderr.F90 tlprop.F90 tltest.F90 writelct.F90 writesd.F90 xformev.F90

Files modified(OOPS):

ifs/CMakeLists.txt model/ErrorCovariance3D.interface.F90 model/FieldsIFS.interface.F90 model/GeometryIFS.interface.F90 model/ModelIFS.interface.F90 model/mpi_wrapper.F90

Files modified(SCRIPTS):

build/Makefile.root.ifsaux

def/inc_an.py inc_libs.py inc_stream.py

gen/ens_cal_rad ens_fetch_fields ens_stats_mem fetch_jb_fields_mem fetchmars ifstraj model p4_allcompilefiles run_parallel

nemo/ndiags.h nemo.h

sms/oopsifslib.sms p4setup.sms

Tomas Wilhelmsson - nat_CY42R1_mkglobstab

Split MKGLOBSTAB for OOPS

Split MKGLOBSTAB into observation space and an model space parts with the new derived type TLOCS passed between them.

Files created(IFS):
`module/yomlocs.F90`
`obs_preproc/mkglobstab_model.F90 mkglobstab_obs.F90`

Files modified(IFS):
`module/gom_mod.F90`
`oops/locations_mod.F90`

Tomas Wilhelmsson - nat_CY42R2_spectral_arp_fix

Fix a bug in calls to ALLOCATE_SPEC from ALLOCATE_ARP

Corrects a bug from nat_CY42R1_OOPS.

Files modified(IFS):
`module/spectral_arp_mod.F90`
`setup/suspsdt.F90`

Ryad El Khatib - das_SB42R1_Ryad

FULLPOS refactoring and introduction of Configuration 903

New configuration NCONF=903 which is a configuration for off-line post-processing : this configuration is still using the setup (su0yoma/b) but not the cascade of control subroutines CNT1-CNT2-CNT3-CNT4.

The aims of this new configurations are :

- laboratory of development for Fullpos-in-OOPS, by leaving the setup progressively and implementing soon a ”server” facility : this configuration should be able to handle several input files consecutively, with possibly different geometries, and make possibly different output geometries for each of them.
- replacement of the configuration 901 (for the transformation of ECMWF grib file into FA file)
- first generation post-processing server (loop on several input files, saving the time of mpi startup and model setup)
- externalization of Fullpos
- second generation post-processing server (a system similar to the mechanism of the IO server, but in the context of OOPS).

Usage : Normally, it should be enough to replace NCONF=001 by NCONF=903 in any job of post-processing.

Files created(IFS):
`control/cprep3.F90`
`setup/suct1.F90 sufpinif.F90`
`utility/filedate.F90 gribioflush.F90 logdis.F90 sigpost.F90`

Files modified(IFS):

control/cnt0.F90 cnt3_wait.F90 cnt4.F90

fullpos/subfpos.F90 sufprfpbuf_clim.F90 sufprfpbuf_geom.F90

interpol/suvsleta.F90

module/yomarg.F90

namelist/namafn.nam.h namct1.nam.h

setup/su0yoma.F90 su0yomb.F90 su1yom.F90 suct0.F90 sumpout.F90 suoph0.F90 sutim.F90

Alan Geer and Deborah Salmond - stg_CY42R1_pre_oops

OOPS observation operator developments

Major changes and code cleaning in the observation operator to support OOPS, with an additional aim to improve the code structure (to make it more modern, modular, encapsulated, easier to understand, and more maintainable). Included in this: - The way VarBC is accessed from the observation operator has been rationalised, with VarBC encapsulated in an object. - References to global data have been removed (for example, GOM arrays are not longer accessed directly and the new GOM_PLUS is the source of all model information passed to the observation operator) - Sets are now ordered by sequence number to support GOM_PLUS (the main non-reproducible change here). - PP routines (vertical interpolation for conventional data) are being replaced with a single top-level routine PPNEW. - A ”hop_driver” test harness has been introduced that can run the observation operator offline, starting from saved GOM_PLUS and ODB files.

Further cleaning, with attention to the TL and adjoint code, will follow in 42r3.

Files created(IFS):

module/varbc_class.F90

op_obs/obsop_composition.F90 obsop_varbc.F90

Files modified(IFS):

canari/caclsi.F90 cadavr.F90 can1.F90 canari.F90

control/adjotest.F90 cad1.F90 cdsta.F90 cfcsens2obs.F90 cgr1.F90 cnt0.F90 cnt1.F90 cnt2.F90 cnt3.F90 cnt3ad.F90 cnt3tl.F90 cnt4.F90 cnt4ad.F90 cnt4tl.F90 csekf1.F90 csekf2.F90 ctl1.F90 cva1.F90 cva2.F90 forecast_error.F90 sim4d.F90 stepo.F90 stepoad.F90 tesadj.F90 testli.F90 testlievol.F90

dfi/dfi.F90 dfi2.F90 dfi2mod.F90 dfi3.F90

fullpos/fpachmt.F90 predynfpos.F90

module/gom_plus.F90 sats_mix.F90 varbc_airep.F90 varbc_allsky.F90 varbc_eval.F90 varbc_gbrad.F90 varbc_pred.F90 varbc_rad.F90 varbc_rsonde.F90 varbc_setup.F90 varbc_sfcobs.F90 varbc_table.F90 varbc_tcwv.F90 varbc_to3.F90 yom_ygfl.F90 yomdimo.F90 yommwave.F90 yomobset.F90 yomsats.F90

mwave/mwave_emis.F90 mwave_get.F90 mwave_get_ad.F90 mwave_get_tl.F90 mwave_obsop.F90 mwave_obsop_ad.F90 mwave_obsop_test.F90 mwave_obsop_tl.F90 mwave_put.F90 mwave_screen.F90 mwave_wrapper.F90

namelist/namgfl.nam.h namsats.nam.h

obs_preproc/black.F90 defrun.F90 gefger.F90 suobsb.F90

oops/allobs_mod.F90 error_covariance_3d_mod.F90

op_obs/amv_get_preds.F90 amv_oberr.F90 bgobs.F90 co2slicing.F90 gems_profs.F90 hdepart.F90 hop.F90 hopad.F90 hoptl.F90 hradp_ml.F90 hretr.F90 obsop_rad.F90

obsv.F90 obsvad.F90 obsvtl.F90 radtr_ml.F90 radtr_ml_ad.F90 radtr_ml_tl.F90
reflsim.F90 reflsim_2dop.F90 rtl_hop_1d.F90 rtl_hop_2d.F90

phys_dmn/achmt.F90 aplpar.F90

pp_obs/ppcc.F90 ppclw.F90 ppobsa.F90 ppobsac.F90 ppobsap.F90 ppobsas.F90
ppobsn.F90 ppq.F90 ppt.F90 ppuv.F90

programs/merge_varbc.F90

setup/su0yomb.F90 su1yom.F90 sugfl1.F90

sinvect/balanced_reduction.F90 chnorm.F90 chsymeig.F90 cun1.F90 cun2.F90
cun3.F90 eof_matrix.F90 jacdav.F90 lcnorad.F90 lcnortl.F90 nalan1.F90 nalan2.F90
opk.F90 opm.F90 pcgbfgs.F90 suforce.F90 sulcz.F90 wrtsv.F90

var/adtest.F90 bgevecs.F90 bgvecs.F90 cain.F90 cainad.F90 cainin.F90 caininad.F90
chavar.F90 chavarad.F90 chavarin.F90 chavarinad.F90 congrad.F90 cosens.F90 cosjc.F90
cosjl.F90 cosjr.F90 costra.F90 cvar2.F90 cvar2ad.F90 cvar2in.F90 cvar2inad.F90 djbdy.F90
ecset.F90 evcost.F90 litest.F90 preppcm.F90 rtsetup.F90 sacmac1.F90 suecges.F90 sujbcov.F90
sujbstd.F90 sujbtest.F90 sujbwavelet.F90 sujbwavgen.F90 sujbwavgen_hybraw.F90 sujbwavstats.F9
suscal.F90 suvazx.F90 taskob.F90 taskobad.F90 taskobtl.F90 tlprop.F90 tltest.F90 upspec.F90
xformev.F90

Files modified(ODB):

ddl/ak_resat_averaging_kernel.sql conv_hdr.sql gbrad_body_rr.sql gbrad_rr.sql get_-
soe_resat.sql raingg_body_rr.sql raingg_rr.sql robhdr.sql robhdr_screen.sql robhdr_-
screen_conv.sql robody.sql robody_screen.sql robody_traj.sql sat_aeolusl2c.sql sat_-
atovs.sql sat_gpsro.sql sat_lrad.sql sat_satob.sql sat_smos.sql sat_ssmi.sql satbody_-
allsky.sql satbody_atovs.sql satbody_gpsro.sql satbody_radar.sql satbody_screen_-
atovs.sql sathdr_cloud_sink.sql sathdr_radar.sql sathdr_screen_atovs.sql sathdr_-
screen_gpsro.sql sathdr_screen_lrad.sql sathdr_screen_resat.sql sathdr_screen_satob.sql

Files modified(SATRAD):

interface/rttvi.h

programs/calc_radiance_fields.F90 gensatim.F90

rttov/ifs/phrtsetup.F90 ifs/rttvi.F90 main/rttov_calcemis_mw.F90 main/rttov_calcemis_-
mw_tl.F90

Files modified(SCRIPTS):

gen/ifsmin ifstraj model

Files deleted(IFS):

var/cvarbc.F90 cvarbcad.F90 cvarbcin.F90 cvarbcinad.F90 fjvarbc.F90 svvarbc.F90

Alan Geer - stg_SB42R1_pre_oops_3

Further OOPS observation operator developments

See stg_CY42R1_pre_oops

Files modified(IFS):

module/gom_plus.F90 testvar_mix.F90 varbc_class.F90 varbc_eval.F90
varbc_pred.F90

op_obs/bgobs.F90 hop.F90 hopad.F90 hoptl.F90 hretr_aeolus.F90 obsop_varbc.F90

pp_obs/ppobsas.F90 ppobsasad.F90 ppobsastl.F90
var/taskob.F90

Alan Geer - stg_CY42R1_fix_bgobs

Reinstate bgobs

The bgobs functionality (i.e. the randomisation method for computing HBHt, required by FSOI computations, for example) was broken by OOPS observation operator developments at 42R1. This functionality is reinstated.

Files modified(IFS):
module/gom_plus.F90
op_obs/bgobs.F90

Alan Geer - stg_SB42R1_sha_hop_driver

Checksums (hashes) for bit-reproducibility testing

Bit-reproducibility problems have been hard to track down in the IFS log files. We do look at norms (like RMS or mean of model fields, and the JO table) but these are insensitive to small changes. The first visible difference between experiments is typically seen downstream of the real problem, particularly in the minimisation. A much better way to detect whether memory contents have changed is to compute a checksum, a small number that has the property of being different even if only a single bit is different between two large data arrays. The well-known SHA-256 hash function is a good choice for this purpose and has been made available in the IFS. It is used to compute a checksum over the GOM arrays, allowing us to easily discover whether bit-reproducibility issues originate in the model or the observation operator.

Bundled with this branch was the hop_driver in support of OOPS developments, which is described earlier.

Files created(IFS):
programs/hop_driver.F90

Files created(IFSAUX):
module/sha256_wrapper.F90
utilities/sha256.c sha256_hash.c

Files modified(IFS):
module/gom_mod.F90 testvar_mix.F90