2012 GSI Community Tutorial
August 21-23, 2012, Boulder, CO

GSI Fundamentals (2)
Run and Namelist

Ming Hu and Hui Shao

Developmental Testbed Center

Developmental Testbed Center

Outlines

- GSI fundamentals (1): Setup and Compilation
- GSI fundamentals (2): Run and Namelist
  - Input files needed by GSI
  - Run GSI with a sample run script
  - Configure GSI with GSI namelist
- GSI fundamentals (3): Diagnostics
- GSI fundamentals (4): Applications

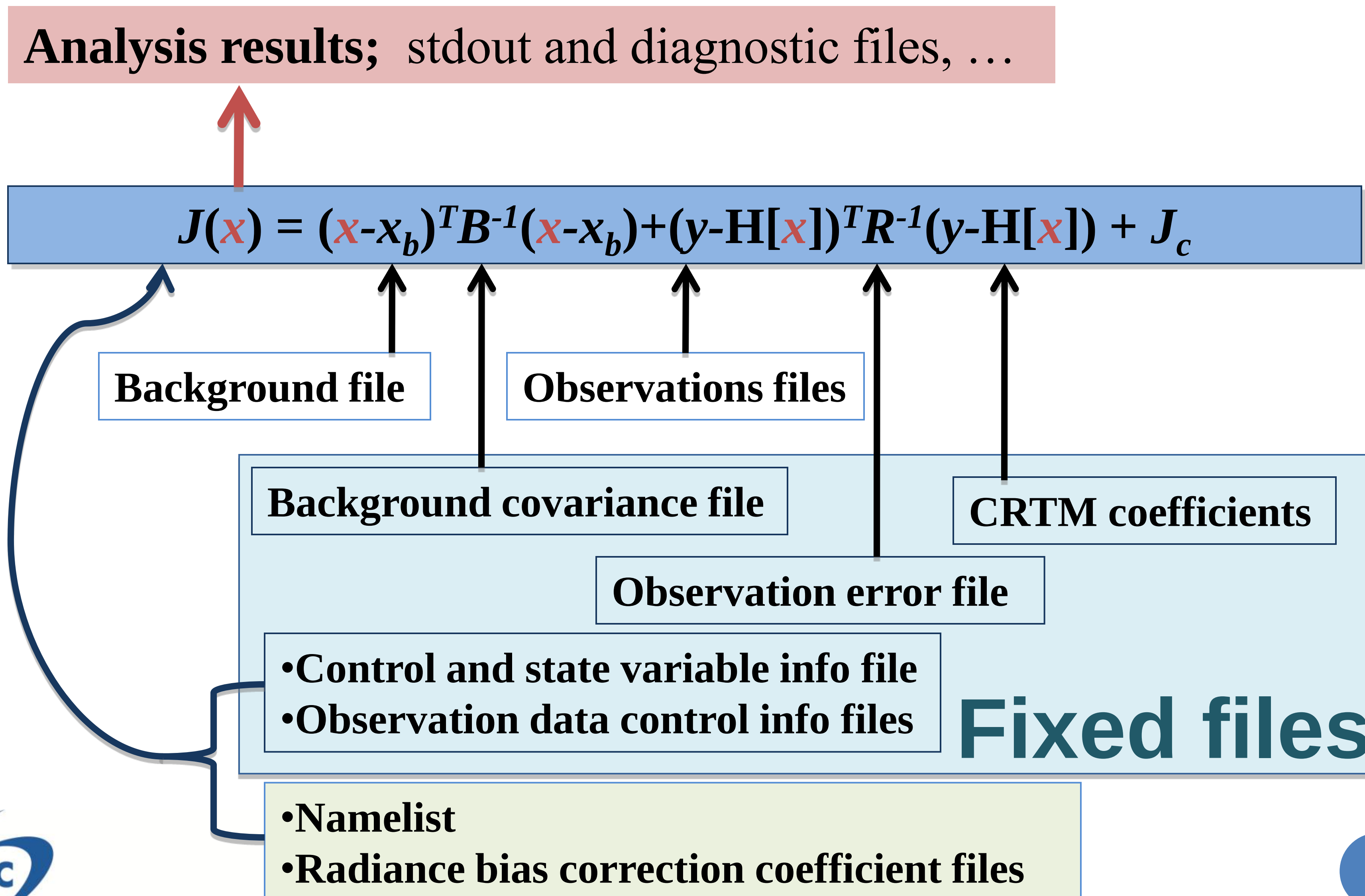
This talk is tailored based on the **Chapter 3** of the **GSI User's Guide** for Community Release V3.1

GSI input files

Before You Run…

GSI User's Guide (section 3.1)

GSI input and output files



Developmental Testbed Center

Background file

- Types of background fields
  a) WRF NMM input fields in binary format
  b) WRF NMM input fields in NetCDF format
  c) WRF ARW input fields in binary format
  d) WRF ARW input fields in NetCDF format
  e) GFS input fields in binary format
  f) NEMS-NMMB input fields
  g) RTMA input filed (2-dimensional binary format)
- DTC mainly support and test (a)-(d)
  - Cold start: WPS + REAL : wrfinput_d01
  - Cycle: forecast file from WRF model : wrfinput_d01_(time)

Sample Observation files

GSI Name	Content	Example file names
prepbufr	Conventional observations, including ps, t, q, pw, uv, spd, dw, sst, from observation platforms such as METAR, sounding, et al.	gdas1.t12z.prepbufr
satwnd	satellite winds	gdas1.t12z.satwnd.tm00.bufr_d
amsuabufr	AMSU-A 1b radiance (brightness temperatures) from satellites NOAA-15, 16, 17,18, 19 and METOP-A/B	gdas1.t12z.1bamua.tm00.bufr_d
amsubbufr	AMSU-B 1b radiance (brightness temperatures) from satellites NOAA15, 16,17	gdas1.t12z.1bamub.tm00.bufr_d
radarbufr	Radar radial velocity Level 2.5 data	ndas.t12z. radwnd. tm12.bufr_d
gpsrobufr	GPS radio occultation observation	gdas1.t12z.gpsro.tm00.bufr_d
ssmirrbufr	Precipitation rate observations fromSSM/I	gdas1.t12z.spssmi.tm00.bufr_d
tmirrbufr	Precipitation rate observations from TMI	gdas1.t12z.sptrmm.tm00.bufr_d
sbuvbufr	SBUV/2 ozone observations from satellite NOAA16, 17, 18, 19	gdas1.t12z.osbuv8.tm00.bufr_d
hirs2bufr	HIRS2 1b radiance from satellite NOAA14	gdas1.t12z.1bhrs2.tm00.bufr_d
hirs3bufr	HIRS3 1b radiance observations from satellite NOAA16, 17	gdas1.t12z.1bhrs3.tm00.bufr_d
hirs4bufr	HIRS4 1b radiance observation from satellite NOAA 18, 19 and METOP-A/B	gdas1.t12z.1bhrs4.tm00.bufr_d
msubufr	MSU observation from satellite NOAA 14	gdas1.t12z.1bmsu.tm00.bufr_d
airsbufr	AMSU-A and AIRS radiances from satellite AQUA	gdas1.t12z.airsev.tm00.bufr_d
mhsbufr	Microwave Humidity Sounder observation from NOAA 18, 19 and METOP-A/B	gdas1.t12z.1bmhs.tm00.bufr_d
ssmitbufr	SSMI observation from satellite f13, f14, f15	gdas1.t12z.ssmit.tm00.bufr_d
amsrebufr	AMSR-E radiance from satellite AQUA	gdas1.t12z.amsre.tm00.bufr_d
ssmisbufr	SSMIS radiances from satellite f16	gdas1.t12z.ssmis.tm00.bufr_d
gsnd1bufr	GOES sounder radiance (sndrd1, sndrd2, sndrd3 sndrd4) from GOES 11, 12, 13.	gdas1.t12z.goesfv.tm00.bufr_d
l2rwbufr	NEXRAD Level 2 radial velocity	ndas.t12z.nexrad. tm12.bufr_d
gsndrbufr	GOES sounder radiance from GOES11, 12	gdas1.t12z.goesnd.tm00.bufr_d
gimgrbufr	GOES imager radiance from GOES 11, 12	
omibufr	Ozone Monitoring Instrument (OMI) observation NASA Aura	gdas1.t12z.omi.tm00.bufr_d
iasibufr	Infrared Atmospheric Sounding Interfero-meter sounder observations from METOP-A/B	gdas1.t12z.mtiasi. tm00.bufr_d
gomebufr	The Global Ozone Monitoring Experiment (GOME) ozone observation from METOP-A/B	gdas1.t12z.gome.tm00.bufr_d
mlsbufr	Aura MLS stratospheric ozone data	gdas1.t12z.mlsbufr.tm00.bufr_d
tcvitl	Synthetic Tropic Cyclone-MSLP observation	gdas1.t12z.syndata.tcvitals.tm00
seviribufr	SEVIRI radiance from MET-08,09,10	gdas1.t12z. sevcsr.tm00.bufr_d
atmsbufr	ATMS radiance from Suomi NPP	gdas1.t12z. atms.tm00.bufr_d
crisbufr	CRIS radiance from Suomi NPP	gdas1.t12z. cris.tm00.bufr_d
modisbufr	MODIS aerosol total column AOD observations from AQUA and TERRA	

Only part of the observation files available in real application

Fixed files

File name used in GSI	Content	Example files in *fix/*
anavinfo	Information file to set control and analysis variables	anavinfo_arw_netcdf anavinfo_ndas_netcdf
berror_stats	background error covariance	nam_nmmstat_na.gcv nam_glb_berror.f77.gcv
errtable	Observation error table	nam_errtable.r3dv
Observation data control file (more detailed explanation in Section 4.3)		
convinfo	Conventional observation information file	global_convinfo.txt
satinfo	satellite channel information file	global_satinfo.txt
pcpinfo	precipitation rate observation information file	global_pcpinfo.txt
ozinfo	ozone observation information file	global_ozinfo.txt
Bias correction used by radiance analysis		
satbias_angle	satellite scan angle dependent bias correction file	global_satangbias.txt
satbias_in	satellite mass bias correction coefficient file	sample.satbias
Radiance coefficient used by CRTM (not under ./fix directory)		
EmisCoeff.bin	IR surface emissivity coefficients	EmisCoeff.bin
AerosolCoeff.bin	Aerosol coefficients	AerosolCoeff.bin
CloudCoeff.bin	Cloud scattering and emission coefficients	CloudCoeff.bin
${satsen}.SpcCoeff.bin	Sensor spectral response characteristics	${satsen}.SpcCoeff.bin
${satsen}.TauCoeff.bin	Transmittance coefficients	${satsen}.TauCoeff.bin

All fixed files are linked to run directory by run script based on user's option

Before run GSI, you should know

- Location and name of background file
- Location and names of observation files
- Location of the Fix directory
- Location of the CRTM coefficients directory
- Have compiled GSI successfully: gsi.exe
- Where do you want to run GSI (run directory)
- Analysis time and configurations
  - Which background error convariance file to u
  - Observation data types used
  - …
- Computer resources you can use and how to use

Too Much!
No problem, let
GSI run script help

DTC Developmental Testbed Center

GSI run script

We do provide a sample run script for you to start with

~/comGSI_v3.1/run/run_gsi.ksh

GSI User's Guide (section 3.2)

GSI run script steps

- Request computer resources to run GSI.
- Set environmental variables for the machine architecture.
- Set experimental variables.

setup

- Check the definitions of required variables.

Check setups

Prepare run directory

- Generate a run directory (working directory).
- Copy the GSI **executable** to the run directory.
- Copy **background** file and link ensemble members if running the hybrid to the run directory.
- Link **observations** to the run directory.
- Link **fixed** files (statistic, control, and coefficient files) to the run directory.
- Generate **namelist** for GSI in the run directory.
- Run the GSI executable.
- Post-process: save analysis results, generate diagnostic files, clean run directory.

GSI run script steps

- Request computer resources to run GSI.
- Set environmental variables for the machine architecture.

setup

```
#!/bin/ksh
#####################################################
# machine set up (users should change this part)
#####################################################
#
#
# GSIPROC = processor number used for GSI analysis
#------------------------------------------------
  GSIPROC=1
  ARCH='LINUX'
# Supported configurations:
            # IBM_LSF, IBM_LoadLevel
            # LINUX, LINUX_LSF, LINUX_PBS,
            # DARWIN_PGI
```

GSI run script steps

- Request computer resources to run GSI. **setup**
- Set environmental variables for the machine architecture.

```
#!/bin/ksh
####################################
# machine set up (users should change this
####################################
#
#
# GSIPROC = processor number used for GSI
#-----------------------------------------
  GSIPROC=1
  ARCH='LINUX'
# Supported configu
          # IBM_L
          #      ,
          # DARWI
```

Linux workstation:

Linux Cluster with PBS:

```
#$ -S /bin/ksh
#$ -N GSI_test
#$ -cwd
#$ -r y
#$ -pe comp 64
#$ -l h_rt=0:20:00
#$ -A ??????
```

IBM with LSF :

```
#BSUB -P ????????
#BSUB -a poe
#BSUB -x
#BSUB -n   12
#BSUB -R "span[ptile=2]"
#BSUB -J gsi
#BSUB -e gsi.err
#BSUB -W 00:02
#BSUB -q regular

set -x

# Set environment variables for IBM
export MP_SHARED_MEMORY=yes
export MEMORY_AFFINITY=MCM
```

GSI run script steps

- Request computer resources to run GSI.
- Set environmental variables for the machine architecture.

setup

```
#!/bin/ksh
#####################################
# machine set up (users should change this
#####################################
#
#
# GSIPROC = processor number used for GSI a
#-------------------------------------------
  GSIPROC=1
  ARCH='LINUX'
# Supported configurations:
            # IBM_LSF, IBM_LoadLevel
            # LINUX, LINUX_LSF, LINUX_PBS,
            # DARWIN_PGI
```

IBM with LSF :

```
GSIPROC=12
ARCH='IBM_LSF'
```

Linux Cluster with PBS:

```
GSIPROC=64
ARCH='LINUX_PBS'
```

Linux workstation:

```
GSIPROC=1
ARCH='LINUX'
```

ARCH is used to decide the **BYTE_ORDER** and **RUN_COMMAND** :

- IBM with LSF: `RUN_COMMAND="mpirun.lsf "`
- Linux Cluster with PBS: `RUN_COMMAND="mpirun -np ${GSIPROC}"`

GSI run script : setup case

- Set experimental variables.

setup

```
##################################################
# case set up (users should change this part)
##################################################
# ANAL_TIME= analysis time  (YYYYMMDDHH)
# WORK_ROOT= working directory, where GSI runs
# PREPBURF = path of PreBUFR conventional obs
# BK_FILE  = path and name of background file
# OBS_ROOT = path of observations files
# CRTM_ROOT= path of crtm coefficients files
# FIX_ROOT = path of fix files
# GSI_EXE  = path and name of the gsi executable
#------------------------------------------------
  ANAL_TIME=2008051112
  WORK_ROOT=./run_${ANAL_TIME}_arw_single
  BK_FILE=./bkARW/wrfout_d01_2008-05-11_12:00:00
  OBS_ROOT=./GSI_DATA/obs
  PREPBUFR=${OBS_ROOT}/newgblav.gdas1.t12z.prepbufr.nr
  CRTM_ROOT=./comGSI/crtm/CRTM_Coefficients
  FIX_ROOT=./comGSI_v3.1/fix
  GSI_EXE=./comGSI_v3.1/run/gsi.exe
```

Where are input data

Where is GSI system

GSI run script : setup case

- Set experimental variables. **setup**

```
#------------------------------------------------------------
# bk_core= which WRF core is used as background (NMM or ARW)
# bkcv_option= which background error covariance and parameter will
#              be used (GLOBAL or NAM)
# if_clean = clean :delete temporal files in working directory
(default)
#       no : leave running directory as is (this is for debug only)
  bk_core=ARW
  bkcv_option=NAM
  if_clean=clean
```

bk_core =ARW	:	ARW NetCDF
=NMM	:	NMM NetCDF

bkcv_option =NAM	:	background error file based on NAM (north H)
=GLOBAL	:	background error file based on GFS (global)

Developmental Testbed Center

GSI run script : setup hybrid case

- Set experimental variables.

setup

```
#----------------------------------------------------
# GSI hybrid options
# if_hybrid = .true. then turn on hybrid data analysis
# ntotmem = number of ensemble members used for hybrid
# mempath = path of ensemble members
#
# please note that we assume the ensemble member are located
#      under ${mempath} with name wrfout_d01_${iiimem}.
#      If this is not the case, please change the following lines:
#
#      if [ -r  ${mempath}/wrfout_d01_${iiimem} ]; then
#         ln -sf ${mempath}/wrfout_d01_${iiimem} ./wrf_en${iiimem}
#      else
#         echo "member ${mempath}/wrfout_d01_${iiimem} is not exit"
#      fi
#
  if_hybrid=.false.
  ntotmem=40
  mempath=/ensemble_forecast/2011060212/wrfprd
```

GSI run script: setup run directory

Prepare run directory

- Generate a run directory (working directory).
- Copy the GSI **executable** to the run directory.
- Copy **background** file and link ensemble members if running the hybrid to the run directory.
- Link **observations** to the run directory.
- Link **fixed** files (statistic, control, and coefficient files) to the run directory.
- Generate **namelist** for GSI in the run directory.

```
# Bring over background field
# (it's modified by GSI so we can't link to it)
cp ${BK_FILE} ./wrf_inout
```

```
# Link ensember members if use hybrid
if [ ${if_hybrid} = .true. ] ; then
        ……
 if [ -r  ${mempath}/wrfout_d01_${iiimem} ]; then
         ln -sf ${mempath}/wrfout_d01_${iiimem} ./wrf_en${iiimem}
      else
         echo "member ${mempath}/wrfout_d01_${iiimem} is not exit"
         exit 1
      fi
```

Developmental Testbed Center

GSI run script: setup run directory

Prepare run directory

- Generate a run directory (working directory).
- Copy the GSI **executable** to the run directory.
- Copy **background** file and link ensemble members if running the hybrid to the run directory.
- Link **observations** to the run directory.
- Link **fixed** files (statistic, control, and coefficient files) to the run directory.
- Generate **namelist** for GSI in the run directory.

```
# Link to the prepbufr data
ln -s ${PREPBUFR} ./prepbufr

# Link to the radiance data
# ln -s ${OBS_ROOT}/le_gdas1.t12z.1bamua.tm00.bufr_d amsuabufr
# ln -s ${OBS_ROOT}/le_gdas1.t12z.1bamub.tm00.bufr_d amsubbufr
# ln -s ${OBS_ROOT}/le_gdas1.t12z.1bhrs3.tm00.bufr_d hirs3bufr
# ln -s ${OBS_ROOT}/le_gdas1.t12z.1bhrs4.tm00.bufr_d hirs4bufr
# ln -s ${OBS_ROOT}/le_gdas1.t12z.1bmhs.tm00.bufr_d mhsbufr
# ln -s ${OBS_ROOT}/le_gdas1.t12z.gpsro.tm00.bufr_d gpsrobufr
```

Byte-order:
Little Endian versus Big Endian

GSI run script: create GSI namelist

- Generate namelist for GSI in the run directory.

```
# Build the GSI namelist on-the-fly
cat << EOF > gsiparm.anl
&SETUP
   miter=2,niter(1)=10,niter(2)=10,
   write_diag(1)=.true.,write_diag(2)=.false.,write_diag(3)=.true.,
   gencode=78,qoption=2,
   factqmin=0.0,factqmax=0.0,deltim=$DELTIM,
   ndat=77,iguess=-1,
   oneobtest=.false.,retrieval=.false.,
   nhr_assimilation=3,l_foto=.false.,
   use_pbl=.false.,
 /
 &GRIDOPTS
   JCAP=$JCAP,JCAP_B=$JCAP_B,NLAT=$NLAT,NLON=$LONA,nsig=$LEVS,hybrid=.true.,
   wrf_nmm_regional=${bk_core_nmm},wrf_mass_regional=${bk_core_arw},
   diagnostic_reg=.false.,
   filled_grid=.false.,half_grid=.true.,netcdf=.true.,
 /
 &BKGERR
   vs=${vs_op}
   hzscl=${hzscl_op}
   bw=0.,fstat=.true.
 /
```

Complete list is in Appendix B.
Most of options should stick with default values

GSI run script: run and submit

- Run the GSI executable

```
case $ARCH in
   'IBM_LSF'|'IBM_LoadLevel')
      ${RUN_COMMAND} ./gsi.exe < gsiparm.anl > stdout 2>&1  ;;

   * )
      ${RUN_COMMAND} ./gsi.exe > stdout 2>&1  ;;
esac
```

- To submit the run script

```
•IBM supercomputer LSF:          ./bsub < run_gsi.ksh
•IBM supercomputer LoadLevel: ./llsubmit run_gsi.ksh
•Linux cluster PBS:              ./qsub run_gsi.ksh
•Linux workstation:              ./run_gsi.ksh
```

No difference from other job submitting

GSI run results: file in run directory

- Examples of GSI run directory after clean

amsuabufr	**fit_p1.2011032212**	fort.221
amsubbufr	**fit_q1.2011032212**	gpsrobufr
anavinfo	**fit_rad1.2011032212**	gsi.exe
berror_stats	**fit_t1.2011032212**	gsiparm.anl
convinfo	**fit_w1.2011032212**	hirs3bufr
diag_amsua_metop-a_anl.2011032212	fort.201	hirs4bufr
diag_amsua_metop-a_ges.2011032212	fort.202	l2rwbufr
diag_amsua_n15_anl.2011032212	fort.203	list_run_directory
diag_amsua_n15_ges.2011032212	fort.204	mhsbufr
diag_amsua_n18_anl.2011032212	fort.205	ozinfo
diag_amsua_n18_ges.2011032212	fort.206	pcpbias_out
diag_amsub_n17_anl.2011032212	fort.207	pcpinfo
diag_amsub_n17_ges.2011032212	fort.208	prepbufr
diag_conv_anl.2011032212	fort.209	prepobs_prep.bufrtable
diag_conv_ges.2011032212	fort.210	satbias_angle
diag_hirs3_n17_anl.2011032212	fort.211	satbias_in
diag_hirs3_n17_ges.2011032212	fort.212	satbias_out
diag_hirs4_metop-a_anl.2011032212	fort.213	satinfo
diag_hirs4_metop-a_ges.2011032212	fort.214	stdout
diag_mhs_metop-a_anl.2011032212	fort.215	**stdout.anl.2011032212**
diag_mhs_metop-a_ges.2011032212	fort.217	**wrfanl.2011032212**
diag_mhs_n18_anl.2011032212	fort.218	wrf_inout
diag_mhs_n18_ges.2011032212	fort.219	
errtable	**fort.220**	

repeat the GSI run:
- Turn off clean option
- Get in run directory
- Replace wrf_inout
- Run gsi.exe

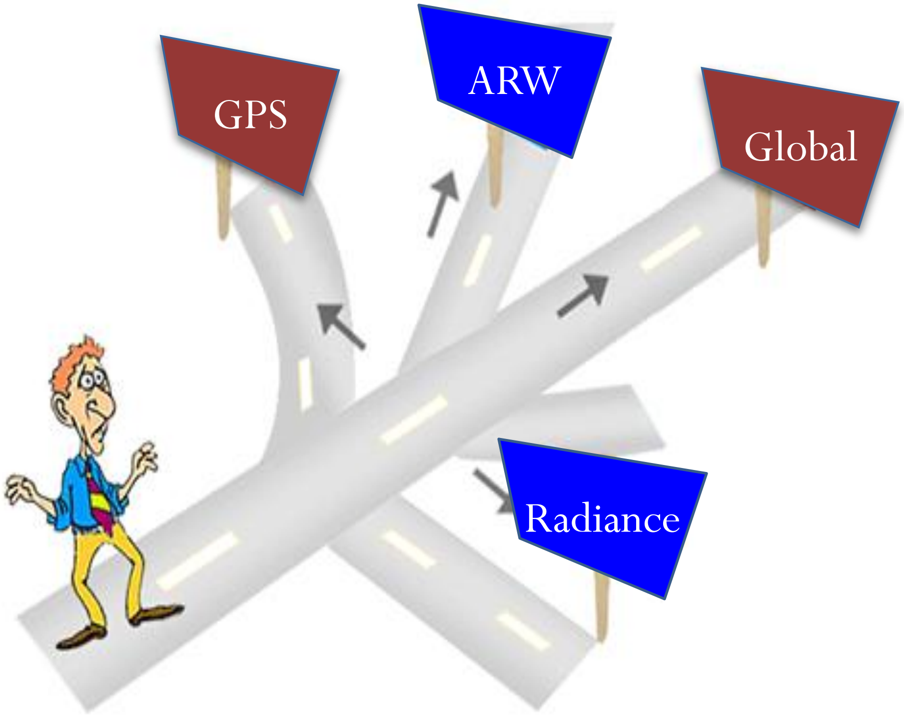
Please reference to "GSI Fundamentals (3): Diagnostics" and "GSI Fundamentals (4): Applications" for more details of the contents

Namelist Options

Also cover convinfo and satinfo to configure a observation data usage

GSI User's Guide

(section 3.3 and part of section 4.3)



GSI Namelist sections

1. SETUP: General control variables.
2. GRIDOPTS: Grid setup variable, including global regional specific namelist variables.
3. BKGERR: Background error related variables.
4. ANBKGERR: Anisotropic background error related variables.
5. JCOPTS: Constraint term in cost function (Jc)
6. STRONGOPTS: Strong dynamic constraint.
7. OBSQC: Observation quality control variables.
8. OBS_INPUT: Input data control variables.
9. SUPEROB_RADAR: Level 2 BUFR file to radar wind superobs.
10. LAG_DATA: Lagrangian data assimilation related variables.
11. HYBRID_ENSEMBLE: Parameters for use with hybrid ensemble option
12. RAPIDREFRESH_CLDSURF: Options for cloud analysis and surface enhancement for RR application
13. CHEM: Chemistry data assimilation
14. SINGLEOB_TEST: Pseudo single observation test setup.

Complete list is in Appendix B: Total 294 namelist variables

Namelist: outer loop and inner loop

- miter: number of outer loops of analysis.
- niter(1): max iteration number of inner iterations for the 1st outer loop.
- niter(2): max iteration number of inner iterations for the 2nd outer loop.
- Inner loop will stop when:
  - reaches this maximum number
  - reaches the convergence condition
  - fails to converge.

The niter sets 10 for fast test in the sample script, usually set 50 for regional applications

```
&SETUP
  miter=2,niter(1)=10,niter(2)=10,
  write_diag(1)=.true.,write_diag(2)=.false.,write_diag(3)=.true.,
  gencode=78,qoption=2,factqmin=0.0,factqmax=0.0,deltim=$DELTIM,
  ndat=77,iguess=-1,oneobtest=.false.,retrieval=.false.,
  nhr_assimilation=3,l_foto=.false.,use_pbl=.false.,
/
```

Namelist: output of diagnostic files

- If following variable is true:
  - write_diag(1): write out diagnostic data before 1st loop (O-B)
  - write_diag(2): write out diagnostic data in between the 1st and 2nd loop
  - write_diag(3): write out diagnostic data at the end of the 2nd outer loop
    - if the outer loop number is 2, it saves information of O-A
- Examples in run directory
  - `O-A: diag_conv_anl.2011032212 diag_amsua_n18_anl.2011032212`
  - `O-B: diag_conv_ges.2011032212 diag_amsua_n18_ges.2011032212`
  - Please see Chunhua's talk on the content of each file and how to read them in "GSI Fundamentals (3): Diagnostics"

```
&SETUP
  miter=2,niter(1)=10,niter(2)=10,
  write_diag(1)=.true.,write_diag(2)=.false.,write_diag(3)=.true.,
  gencode=78,qoption=2,factqmin=0.0,factqmax=0.0,deltim=$DELTIM,
  ndat=77,iguess=-1,oneobtest=.false.,retrieval=.false.,
  nhr_assimilation=3,l_foto=.false.,use_pbl=.false.,
/
```

Namelist: single observation test

- oneobtest=.true
- In &SINGLEOB_TEST, set up
  - Innovation
  - Observation error
  - Observation variable
  - Location
- Section 4.2 for more details

Please see Dayrl's talk: "*Background and Observation Error Estimation and Tuning*" in **2011** GSI summer tutorial for example of single observation application (slides available on-line)

```
&SETUP
  miter=2,niter(1)=10,niter(2)=10,
  write_diag(1)=.true.,write_diag(2)=.false.,write_diag(3)=.true.,
  gencode=78,qoption=2,factqmin=0.0,factqmax=0.0,deltim=$DELTIM,
  ndat=77,iguess=-1,oneobtest=.false.,retrieval=.false.,
  nhr_assimilation=3,l_foto=.false.,use_pbl=.false.,
/
```

```
&SINGLEOB_TEST
   maginnov=1.0,magoberr=0.8,oneob_type='t',
   oblat=38.,oblon=279.,obpres=500.,obdattim=${ANAL_TIME},
   obhourset=0.,
 /
```

Developmental Testbed Center

Namelist: the background file

- Regional:
  - if true, perform a regional GSI run
  - If false, perform a global GSI analysis.
- wrf_nmm_regional: if true, background comes from WRF NMM.
- wrf_mass_regional: if true, background comes from WRF ARW.
- netcdf: only works for performing a regional GSI analysis.
  - if true, WRF files are in NetCDF format
  - otherwise WRF files are in binary format

```
&GRIDOPTS
 regional=.true.,wrf_nmm_regional=.false,wrf_mass_regional=.true.,
 diagnostic_reg=.false.,
 filled_grid=.false.,half_grid=.true.,netcdf=.true.,
 /
```

Set data time window

- PrepBUFR/BUFR file cut-off time (more details in Ruifang's talk)
- time_window_max: namelist option to set maximum half time window (hours) for all data types.

```
&OBS_INPUT
  dmesh(1)=120.0,dmesh(2)=60.0,dmesh(3)=60.0,time_window_max=1.5,
  ......
```

- twindow:, set half time window for certain data type (hours).
  - In *convinfo* file, for conventional observations only
  - Conventional observations within the **smaller** window of *time_window_max* and *twindow* will be kept to further process

!otype	type	sub	iuse	twindow	numgrp	ngroup	nmiter	gross	ermax	ermin	var_b	var_pg	ithin	rmesh	pmesh
ps	120	0	1	3.0	0	0	0	4.0	3.0	1.0	4.0	0.000300	0	0.	0.
ps	132	0	-1	3.0	0	0	0	4.0	3.0	1.0	4.0	0.000300	0	0.	0.
t	120	0	1	3.0	0	0	0	8.0	5.6	1.3	8.0	0.000001	0	0.	0.
t	126	0	-1	3.0	0	0	0	8.0	5.6	1.3	8.0	0.001000	0	0.	0.
t	130	0	1	3.0	0	0	0	7.0	5.6	1.3	7.0	0.001000	0	0.	0.

Data usage (1): link observation files

- Link the observation files into run directory

```
# Link to the prepbufr data
ln -s ${PREPBUFR} ./prepbufr

# Link to the radiance data
# ln -s ${OBS_ROOT}/gdas1.t12z.1bamua.tm00.bufr_d amsuabufr
# ln -s ${OBS_ROOT}/gdas1.t12z.1bhrs4.tm00.bufr_d hirs4bufr
# ln -s ${OBS_ROOT}/gdas1.t12z.1bmhs.tm00.bufr_d  mhsbufr
```

GSI Name	Content	Example file names
prepbufr	Conventional observations, including ps, t, q, pw, uv, spd, dw, sst, from observation platforms such as METAR, sounding, et al.	gdas1.t12z.prepbufr
amsuabufr	AMSU-A 1b radiance (brightness temperatures) from satellites NOAA-15, 16, 17,18, 19 and METOP-A	gdas1.t12z.1bamua.tm00.bufr_d
amsubbufr	AMSU-B 1b radiance (brightness temperatures) from satellites NOAA15, 16,17	gdas1.t12z.1bamub.tm00.bufr_d
radarbufr	Radar radial velocity Level 2.5 data	ndas.t12z. radwnd. tm12.bufr_d
gpsrobufr	GPS radio occultation observation	gdas1.t12z.gpsro.tm00.bufr_d
ssmirrbufr	Precipitation rate observations from SSM/I	gdas1.t12z.spssmi.tm00.bufr_d
tmirrbufr	Precipitation rate observations from TMI	gdas1.t12z.sptrmm.tm00.bufr_d
sbuvbufr	SBUV/2 ozone observations from satellite NOAA16, 17, 18, 19	gdas1.t12z.osbuv8.tm00.bufr_d

Data usage (2): set namelist

- Set namelist section &**OBS_INPUT**

Thinning mesh size for each satellite group

upper limit on time window for all input data

```
&OBS_INPUT
 dmesh(1)=120.0,dmesh(2)=60.0,dmesh(3)=60.0,dmesh(4)=60.0,dmesh(5)=120,time_window_max=1.5,
 dfile(01)='prepbufr',   dtype(01)='ps',     dplat(01)='  ',   dsis(01)='ps',        dval(01)=1.0,  dthin(01)=0,
 dfile(02)='prepbufr'    dtype(02)='t',      dplat(02)='  ',   dsis(02)='t',         dval(02)=1.0,  dthin(02)=0,
 dfile(03)='prepbufr',   dtype(03)='q',      dplat(03)='  ',   dsis(03)='q',         dval(03)=1.0,  dthin(03)=0,
 dfile(04)='prepbufr',   dtype(04)='uv',     dplat(04)='  ',   dsis(04)='uv',        dval(04)=1.0,  dthin(04)=0,
……
 dfile(27)='msubufr',    dtype(27)='msu',    dplat(27)='n14',  dsis(27)='msu_n14',   dval(27)=2.0,  dthin(27)=2,
 dfile(28)='amsuabufr',  dtype(28)='amsua',  dplat(28)='n15',  dsis(28)='amsua_n15', dval(28)=10.0, dthin(28)=2,
 dfile(29)='amsuabufr',  dtype(29)='amsua',  dplat(29)='n16',  dsis(29)='amsua_n16', dval(29)=0.0,  dthin(29)=2,
```

Observation type

Satellite (platform) id (for satellite data)

Input observation file name
Can be changed if need.

Satellite thinning mesh group

Weighting factor for super-obs

Sensor/instrument/satellite flag from satinfo files

Also see Kathryn's talk on data thinning in "GSI Fundamentals (4): Applications"

Developmental Testbed Center

Data usage (3): set conventional obs

- *convinfo*
  - control the usage of conventional data (t, q, ps, wind, …) and GPS RO refractivity and bending angle based on data type.
  - User's Guide section 4.3 for more details

!otype	type	sub	iuse	twindow	numgrp	ngroup	nmiter	gross	ermax	ermin	var_b	var_pg	ithin	rmesh	pmesh
ps	120	0	1	3.0	0	0	0	4.0	3.0	1.0	4.0	0.000300	0	0.	0.
ps	180	0	1	3.0	0	0	0	4.0	3.0	1.0	4.0	0.000300	0	0.	0.
ps	181	0	1	3.0	0	0	0	3.6	3.0	1.0	3.6	0.000300	0	0.	0.
ps	183	0	-1	3.0	0	0	0	4.0	3.0	1.0	4.0	0.000300	0	0.	0.
t	120	0	1	3.0	0	0	0	8.0	5.6	1.3	8.0	0.000001	0	0.	0.
t	126	0	-1	3.0	0	0	0	8.0	5.6	1.3	8.0	0.001000	0	0.	0.
t	130	0	1	3.0	0	0	0	7.0	5.6	1.3	7.0	0.001000	0	0.	0.
t	131	0	1	3.0	0	0	0	7.0	5.6	1.3	7.0	0.001000	0	0.	0.
t	180	0	1	3.0	0	0	0	7.0	5.6	1.3	7.0	0.004000	0	0.	0.
t	181	0	-1	3.0	0	0	0	7.0	5.6	1.3	7.0	0.004000	0	0.	0.

iuse=1: use data
iuse=0: do not use data
iuse=-1: monitor data

Gross Check Parameters

Data thinning Parameters

The last column **npred** has been deleted to fit the table in a page

Data usage (4): set radiance obs

- *Satinfo* (User's Guide section 4.3 for more details)
- Control the usage of satellite radiance data (AMSU-A, AMSU-B, HIRS3, …) based on platform and channels.

!sensor/instr/sat	chan	iuse	error	error_cld	ermax	var_b	var_pg
amsua_n15	1	1	3.000	9.100	4.500	10.000	0.000
amsua_n15	2	1	2.000	13.500	4.500	10.000	0.000
amsua_n15	3	1	2.000	7.100	4.500	10.000	0.000
amsua_n15	4	1	0.600	1.300	2.500	10.000	0.000
amsua_n15	14	-1	2.000	1.400	4.500	10.000	0.000
amsua_n15	15	1	3.000	10.000	4.500	10.000	0.000
hirs3_n17	1	-1	2.000	0.000	4.500	10.000	0.000
hirs3_n17	2	-1	0.600	0.000	2.500	10.000	0.000

`ermax`: max error (for QC)

`error`: variance for each satellite channel

`iuse`

=-2: do not use

=-1: monitor if diagnostics produced

=0: monitor and use in QC only

=1: use data with complete quality control

=2 use data with no airmass bias correction

=3 use data with no angle dependent bias correction

=4 use data with no bias correction

Data usage : Summary

- Link the observation files

```
ln -s ${PREPBUFR} ./prepbufr
ln -s ${OBS_ROOT}/gdas1.t12z.1bamua.tm00.bufr_d amsuabufr
```

- Set namelist section &*OBS_INPUT*

```
&OBS_INPUT
 dmesh(1)=120.0,dmesh(2)=60.0,dmesh(3)=60.0,dmesh(4)=60.0,dmesh(5)=120,time_window_max=1.5,
 dfile(01)='prepbufr',  dtype(01)='ps',    dplat(01)=' ',    dsis(01)='ps',        dval(01)=1.0,  dthin(01)=0,
 dfile(02)='prepbufr'   dtype(02)='t',     dplat(02)=' ',    dsis(02)='t',         dval(02)=1.0,  dthin(02)=0,
 dfile(03)='prepbufr',  dtype(03)='q',     dplat(03)=' ',    dsis(03)='q',         dval(03)=1.0,  dthin(03)=0,
 dfile(28)='amsuabufr', dtype(28)='amsua', dplat(28)='n15',  dsis(28)='amsua_n15', dval(28)=10.0, dthin(28)=2,
 dfile(29)='amsuabufr', dtype(29)='amsua', dplat(29)='n16',  dsis(29)='amsua_n16', dval(29)=0.0,  dthin(29)=2,
```

- Set *info* file

!otype	type	sub	iuse	twindow	numgrp	ngroup	nmiter	gross	ermax	ermin	var_b	var_pg	ithin	rmesh	pmesh
ps	120	0	1	3.0	0	0	0	4.0	3.0	1.0	4.0	0.000300	0	0.	0.
ps	180	0	1	3.0	0	0	0	4.0	3.0	1.0	4.0	0.000300	0	0.	0.
ps	181	0	1	3.0	0	0	0	3.6	3.0	1.0	3.6	0.000300	0	0.	0.
ps	183	0	-1	3.0	0	0	0	4.0	3.0	1.0	4.0	0.000300	0	0.	0.
t	120	0	1	3.0	0	0	0	8.0	5.6	1.3	8.0	0.000001	0	0.	0.
t	126	0	-1	3.0											
t	130	0	1	3.0											
t	131	0	1	3.0											
t	180	0	1	3.0											
t	181	0	-1	3.0											

!sensor/instr/sat	chan	iuse	error	error_cld	ermax
amsua_n15	2	1	2.000	13.500	4.500
amsua_n15	3	1	2.000	7.100	4.500
amsua_n15	4	1	0.600	1.300	2.500
amsua_n15	14	-1	2.000	1.400	4.500
amsua_n15	15	1	3.000	10.000	4.500
hirs3_n17	1	-1	2.000	0.000	4.500

Questions? …

gsi_help@ucar.edu