Technical Report on the Development of the GSI-based WRF 4DVAR

Xin Zhang and Xiang-Yu Huang
MMM Division
National Center for Atmospheric Research

GSI Meeting
19 April 2013

Current Status

- Updates toward the GSI-based WRF 4D-Var have been done.
- All Changes have been synchronized and tested with the revision 1061 of Boulder repository (2nd April, 2013).

- The GSI-based WRF 4D-Var is able to run parallel.
- Multiple outer-loops setup works (with script controlled)
- JcDFI as weak constraint for WRF MASS CORE is added and works. (more investigation needed for regional WRF model.)
- All test cases pass the adjoint test.
- All test cases pass the gradient test.

- Most of the changes related to the GSI-based WRF 4D-Var were isolated using #ifdef.

Milestones:

- Re-development of the WRF tangent linear and adjoint model.
  - Zhang, X., X. Huang, and N. Pan, 2013: Development of the Upgraded Tangent Linear and Adjoint of the Weather Research and Forecasting (WRF) Model. J. Atmos. Oceanic Technol. doi:10.1175/JTECH-D-12-00213.1, in press.
- Experiences gained from upgrading the WRF 4D-Var from MPMD to SPMD.
  - Zhang, X., X. Huang, J. Liu and J. Poterjoy, 2013: Development of an efficient regional four-dimensional variational data assimilation system for WRF. Submitted to J. Atmos. Oceanic Technol.
- Existed 4D-Var developments in current GSI framework.
  - Thank Dr. R. Todling and others.

Outline

- Linking GSI with WRF TLM and ADM
- GSI-based WRF 4DVAR
- What to do …

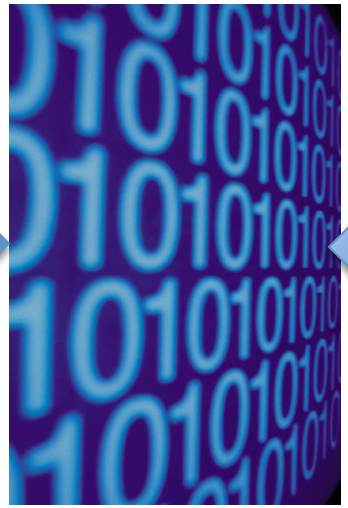
Outline

- Linking GSI with WRF TLM and ADM.
- GSI/WRF 4DVAR
- What to do ……

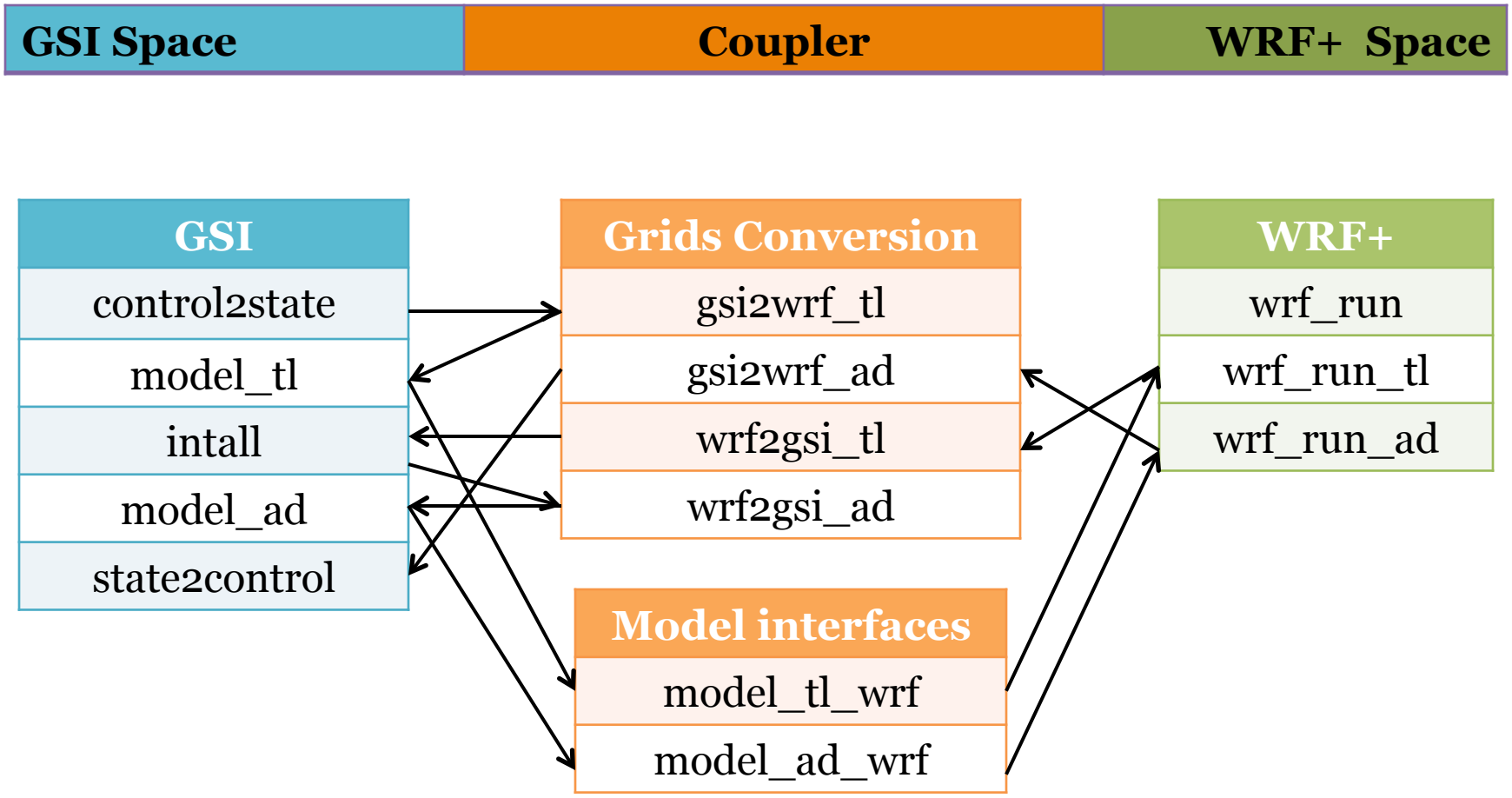
What do we need to link GSI with WRF+ ?

GSI Space		WRF+ Space

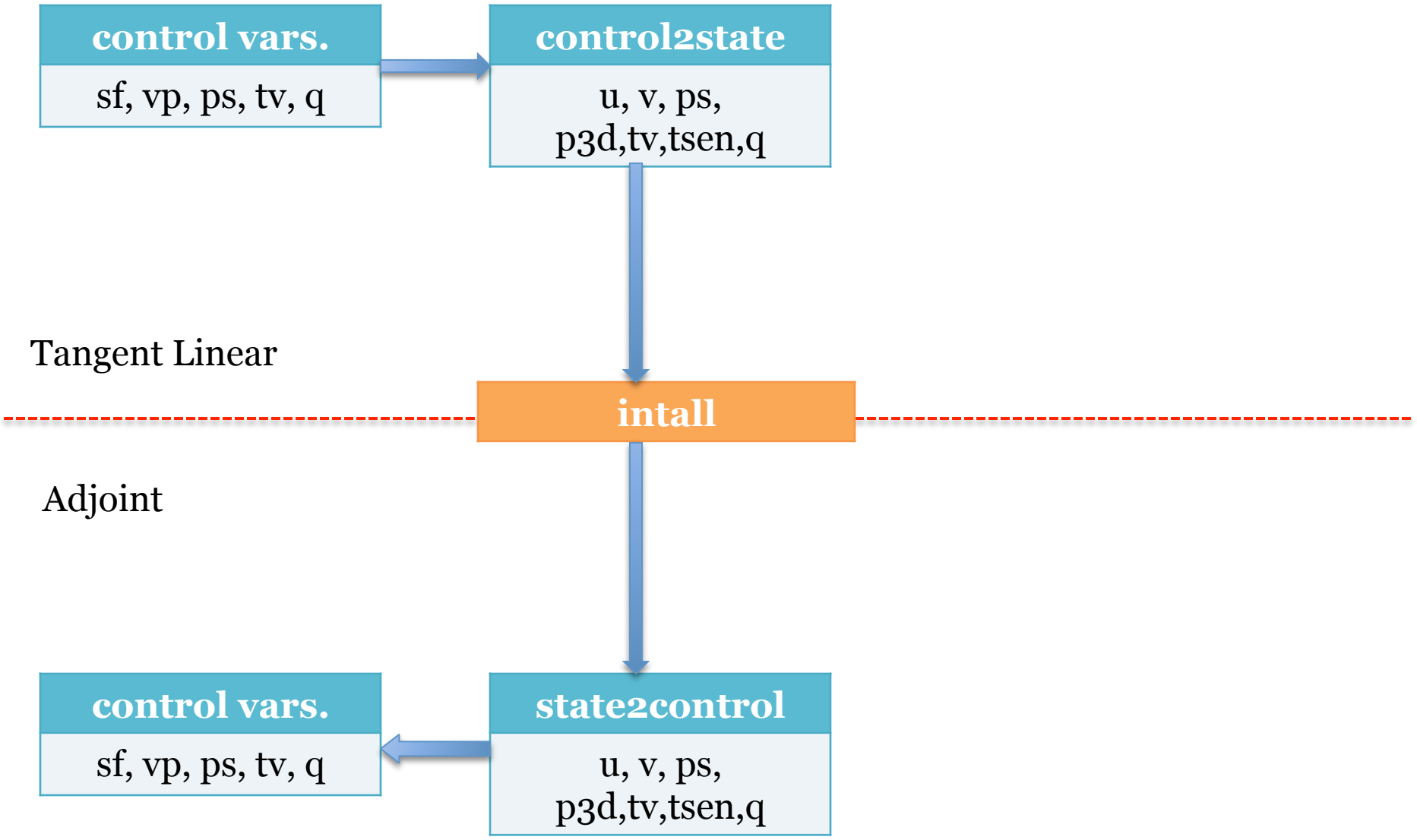
GSI		WRF+
U,V on A-Grid		U, V on C-grid
Virtual Temp.		Potential Temp.
Specific Humidity		Mixing Ratio
Surf. Press.		Dry Surf. Press.
model_tl		wrf_run_tl
model_ad		wrf_run_ad



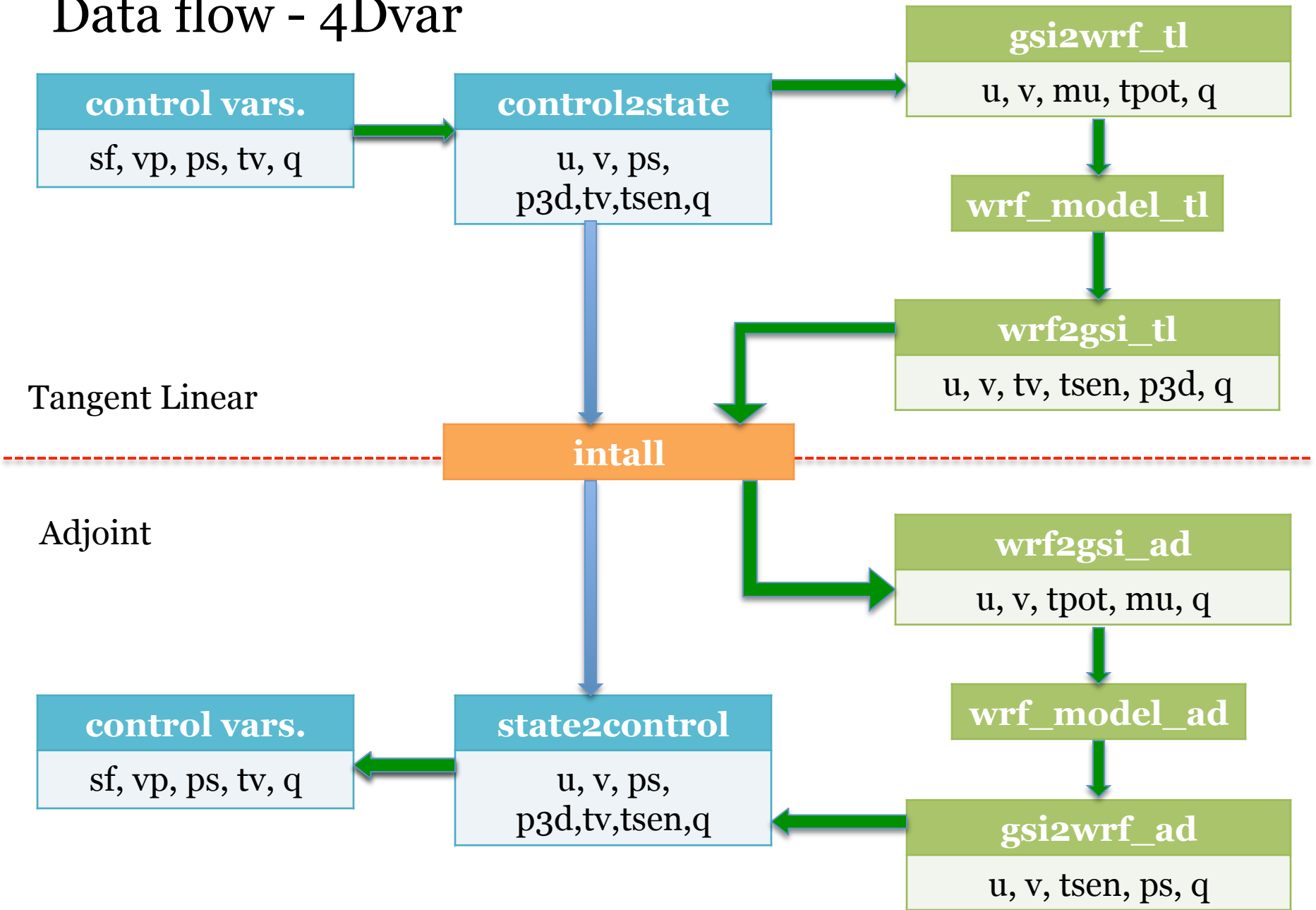
Linking GSI with WRF+



Data flow – 3DVar



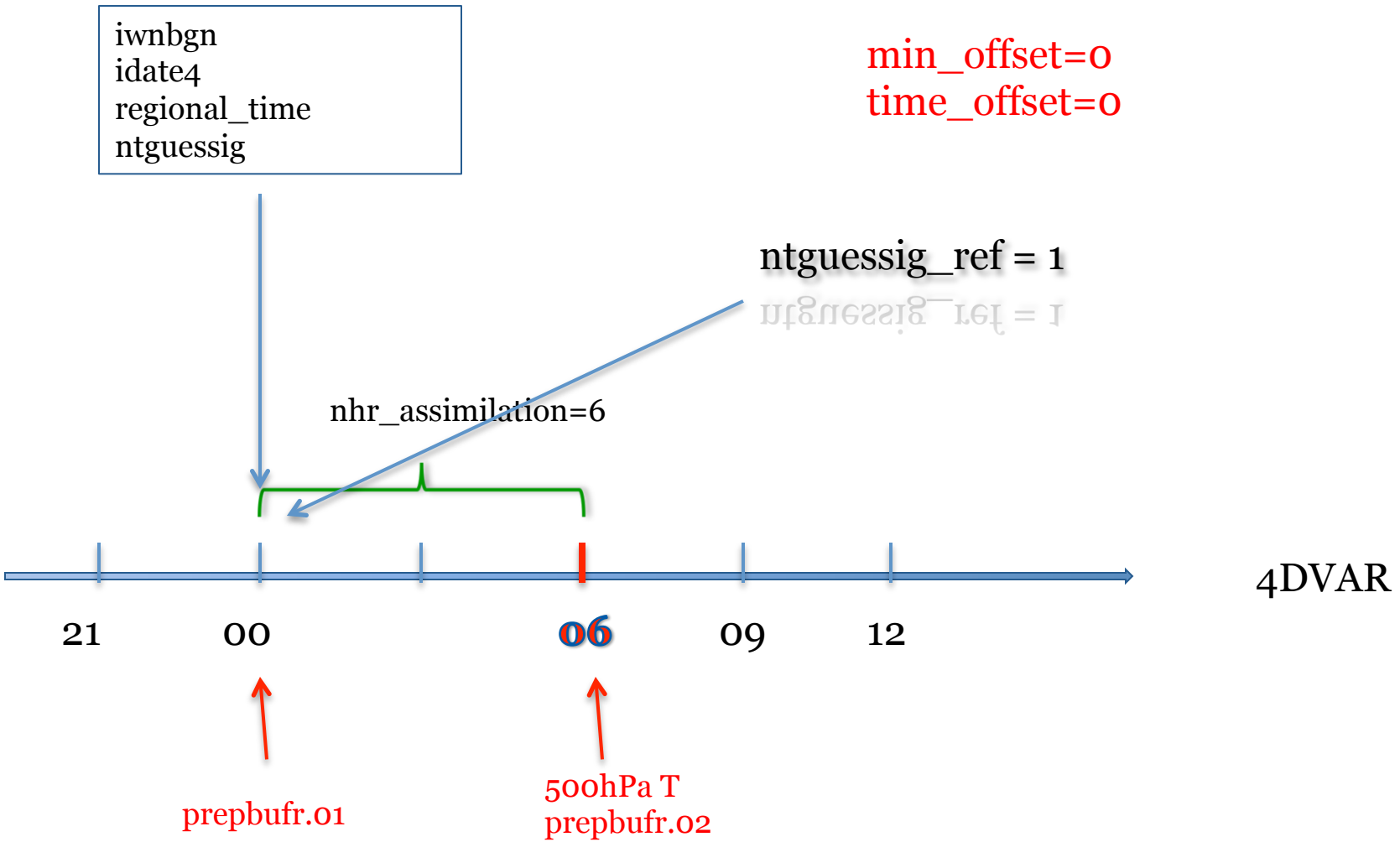
Data flow - 4Dvar



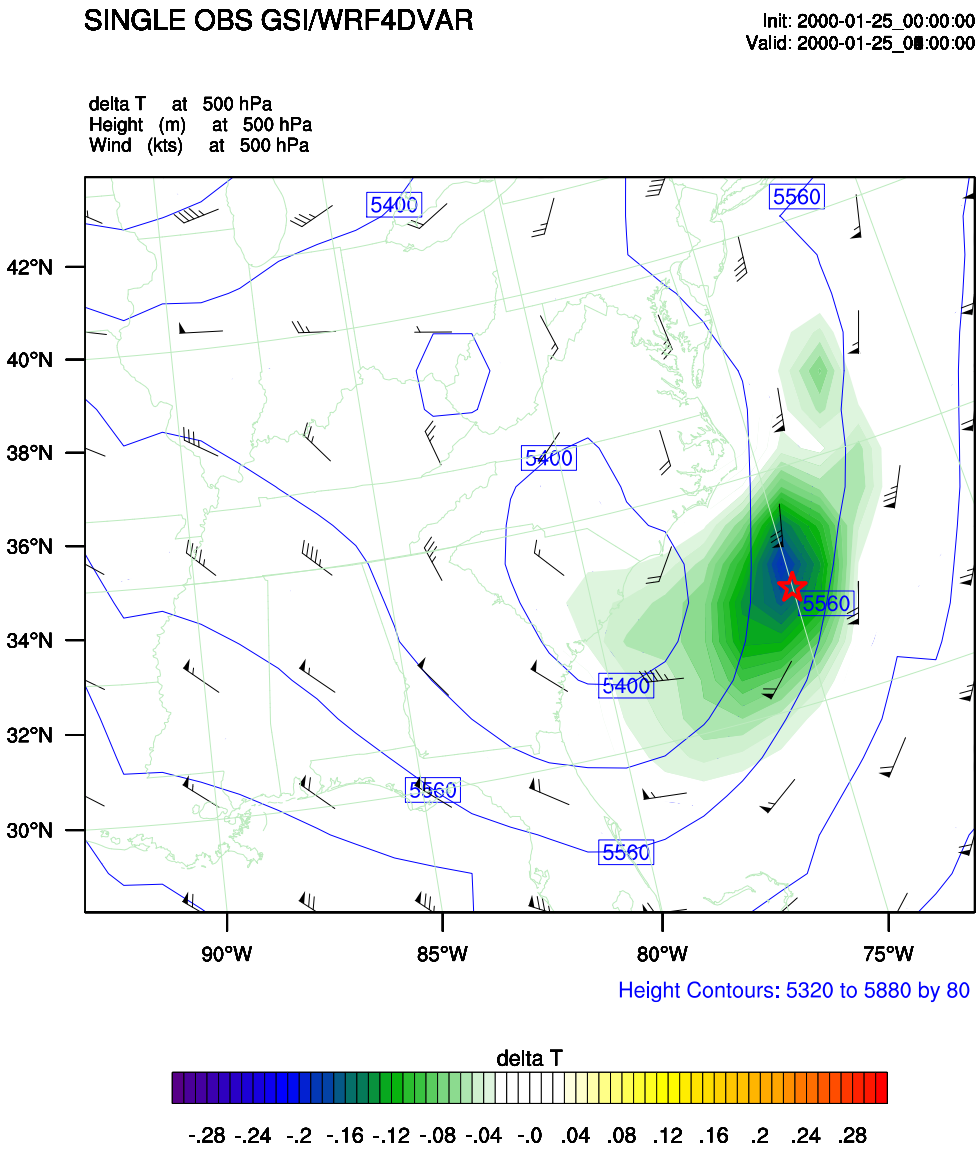
Outline

- Linking GSI with WRF TLM and ADM
- GSI-based WRF 4DVAR
- What to do ……

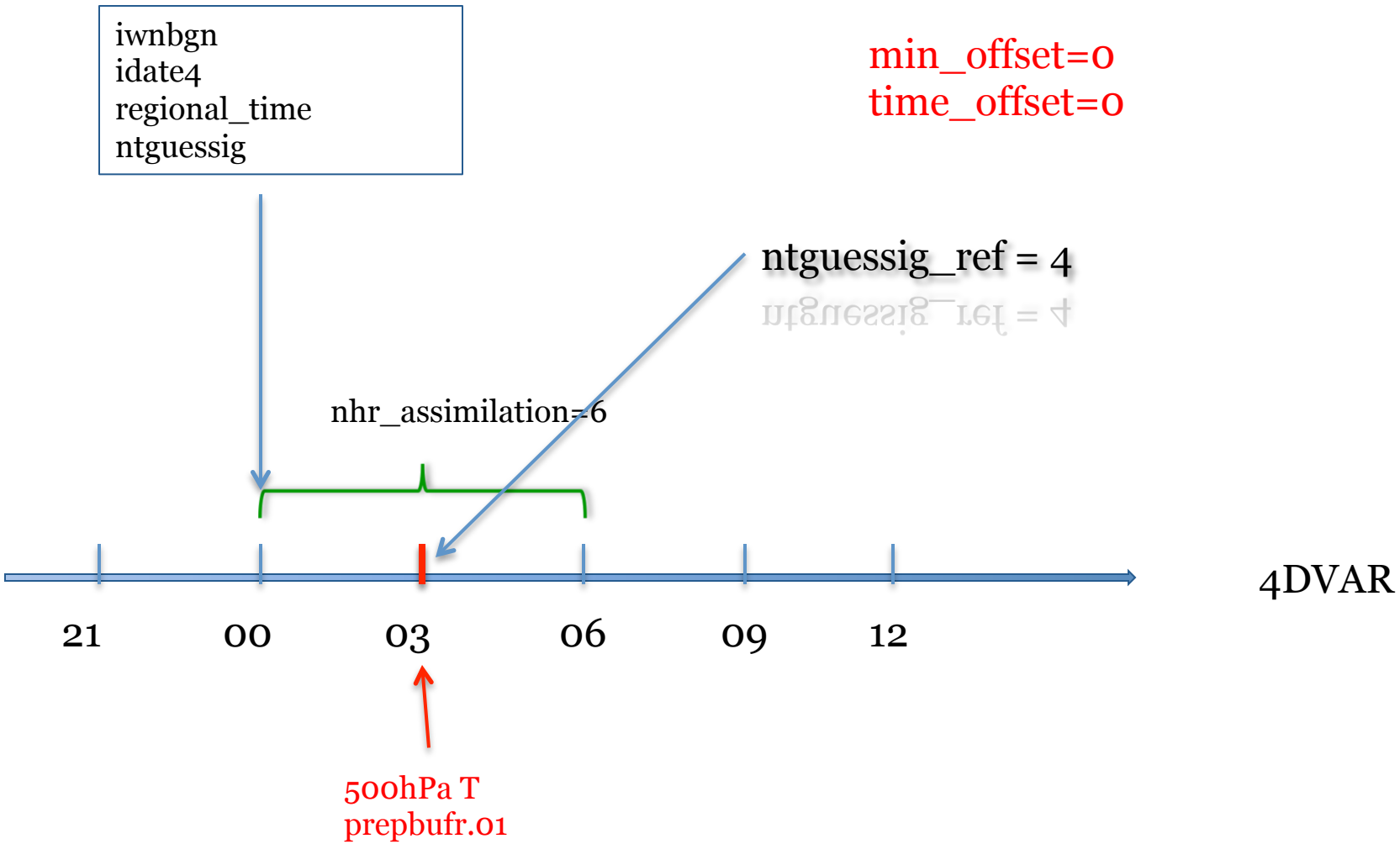
Single pseudo observation Experiments. Scenario I



Single Obs. (6h obs)

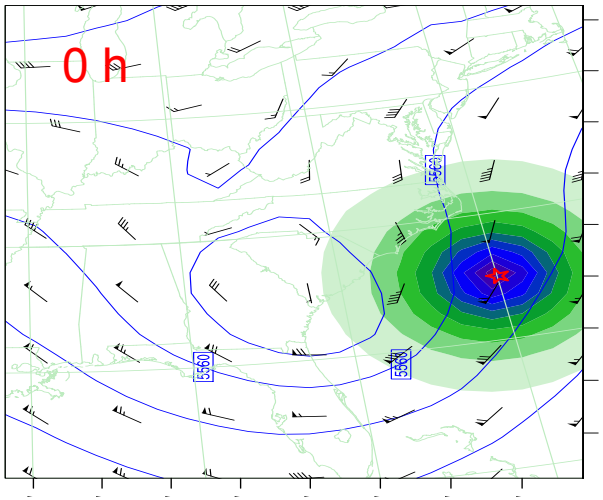


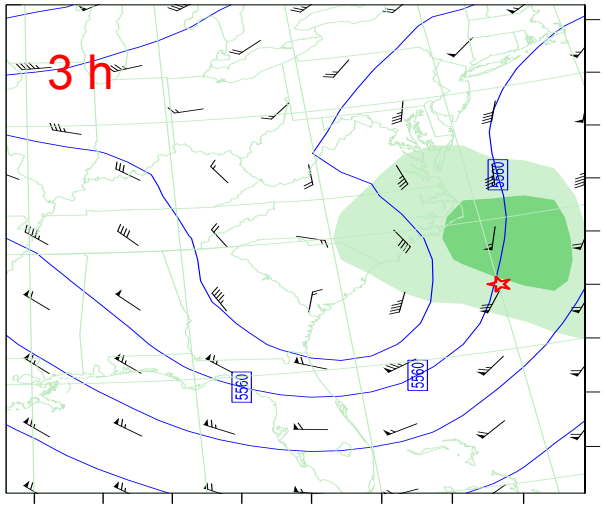
Single pseudo observation Experiments. Scenario II



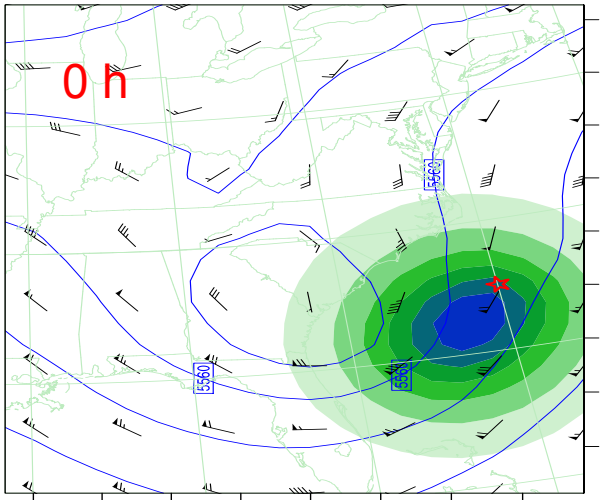
Single Obs. (3h obs.)

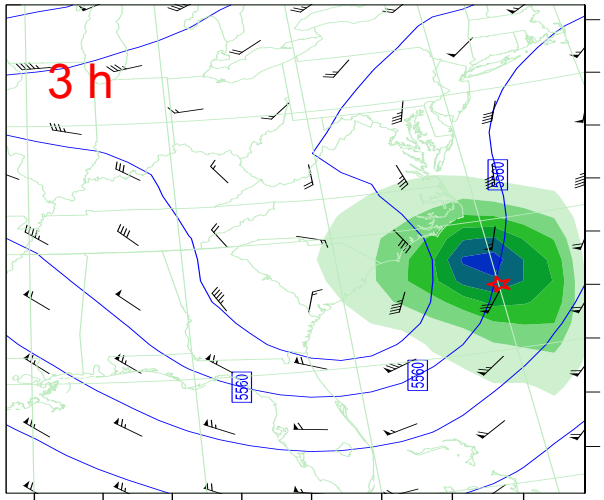
3DVAR





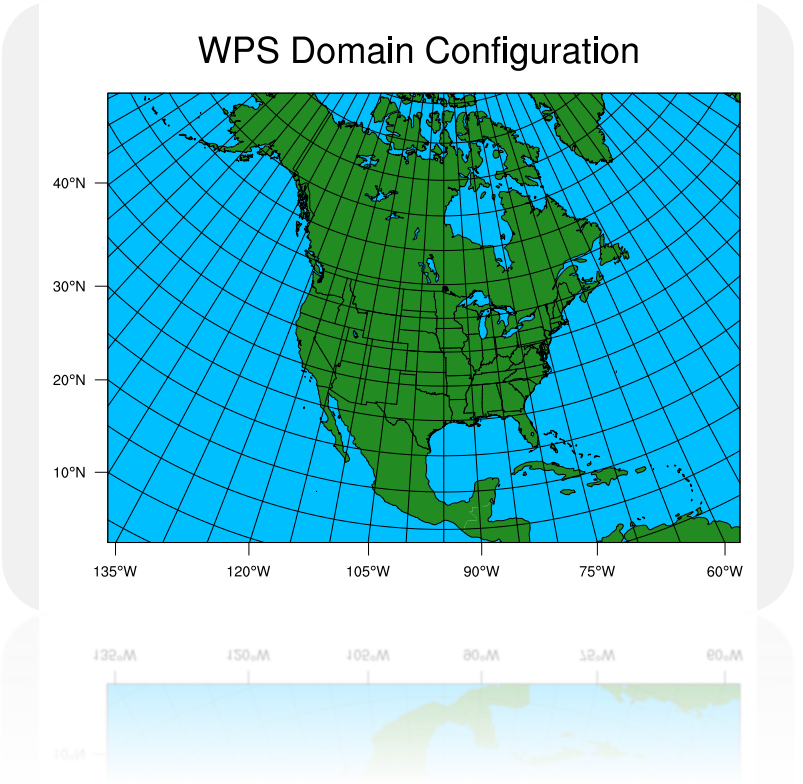
4DVAR





GSI Tutorial case – downgraded to lower resolution

- Analysis time:
  - 2011•03•22•12
- Domain size:
  - 175×83×L51
- Domain horizontal resolution:
  - 90km
- Assimilation window
  - 6 hours (2209 to 2215) for both 3dvar and 4dvar
- Minimization algorithm
  - lsqrtb = .true.
  - lcongrad = .true.
  - ltint = .true.
  - 1 outer loop, 50 inner loops



Observation Usage Statistics

	Surface Pressure	Temp.	Wind	Moisture	GPS	Radiance
3DVAR	13852	9233	25582	4210	8707	131221
4DVAR	13855	9233	25590	4210	8750	145356
Variation	+3	0	+8	0	+43	+14135

Question: We expect that the 4DVar mode should use more data (especially surface obs.) than 3DVar mode does. What's wrong ?

Adjoint Test

3DVAR -before

ADTEST starting
ADTEST use random_cv(xhat)
ADTEST 0.123456789012345678
ADTEST <F*F.Y,X>= 1.90534135460727 1496E+06
ADTEST <F.Y,F.Y>= 1.90534135460725 7294E+06
ADTEST 14 digits are identical
ADTEST rel. err.= 7.454133732279982765E-16
ADTEST mach.eps = 2.220446049250313081E-16

4DVAR-before

ADTEST starting
ADTEST use random_cv(xhat)
ADTEST 0.123456789012345678
ADTEST <F*F.Y,X>= 1.139774043173196539E+07
ADTEST <F.Y,F.Y>= 1.139774043173203059E+07
ADTEST 14 digits are identical
ADTEST rel. err.= 5.719781092889553150E-15
ADTEST mach.eps = 2.220446049250313081E-16

3DVAR-after

ADTEST starting
ADTEST use random_cv(xhat)
ADTEST 0.123456789012345678
ADTEST <F*F.Y,X>= 1.145017507932603126E+06
ADTEST <F.Y,F.Y>= 1.145017507932609413E+06
ADTEST 14 digits are identical
ADTEST rel. err.= 5.490245638256641478E-15
ADTEST mach.eps = 2.220446049250313081E-16

4DVAR-after

ADTEST starting
ADTEST use random_cv(xhat)
ADTEST 0.123456789012345678
ADTEST <F*F.Y,X>= 1.034088584772368520E+07
ADTEST <F.Y,F.Y>= 1.034088584772372805E+07
ADTEST 14 digits are identical
ADTEST rel. err.= 4.142859621812996916E-15
ADTEST mach.eps = 2.220446049250313081E-16

Observation Adjoint Test (By Rizvi Syed)

4DVAR

ADTEST_OBS <F*F.Y,X>= 6.758543423470276175E+05
ADTEST_OBS <F.Y,F.Y>= 6.758543423470275011E+05
ADTEST_OBS 15 digits are identical

Gradient Test

- Parallel problem in gradient test ? (32 PE used)
- One-processor run produces good gradient test results.
- No impact on final analysis.
- Conventional data only works fine.

3DVAR -before

grtest: gradient T1= 1.000799663437064
grtest: gradient T1= 1.000801257924277
grtest: gradient T1= 1.000795318307842
grtest: gradient T1= 1.000735631073795
grtest: gradient T1= 1.000138699719116
grtest: gradient T1= 0.9941693692653023
grtest: gradient T1= 0.9344760659344273
grtest: gradient T1= 0.3375430327161567

4DVAR-before

grtest: gradient T1= 0.9990476095484848
grtest: gradient T1= 0.9990423154564589
grtest: gradient T1= 0.9990367849678288
grtest: gradient T1= 0.9989816676095885
grtest: gradient T1= 0.9984306588591904
grtest: gradient T1= 0.9929206249020169
grtest: gradient T1= 0.9378202843136800
grtest: gradient T1= 0.3868168788307939

3DVAR-after

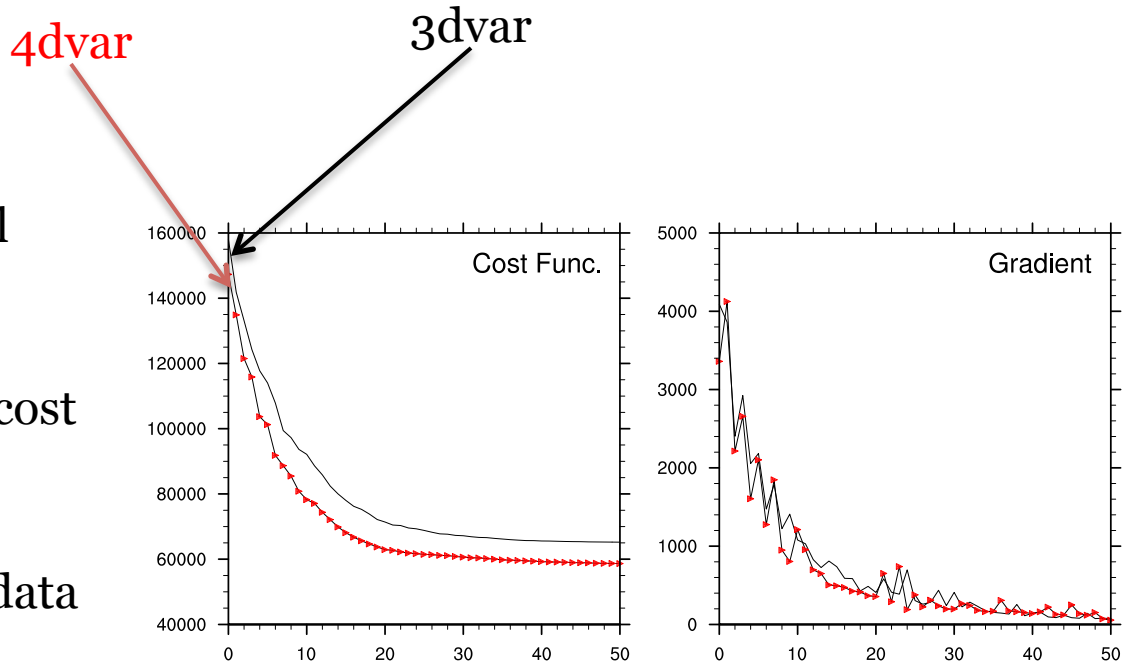
grtest: gradient T1= 1.000015804044360
grtest: gradient T1= 1.000010961587316
grtest: gradient T1= 1.000000368712534
grtest: gradient T1= 0.9999990975675600
grtest: gradient T1= 0.9999914246943746
grtest: gradient T1= 0.9999140072440680
grtest: gradient T1= 0.9991400604212262
grtest: gradient T1= 0.9914006048800456

4DVAR-after

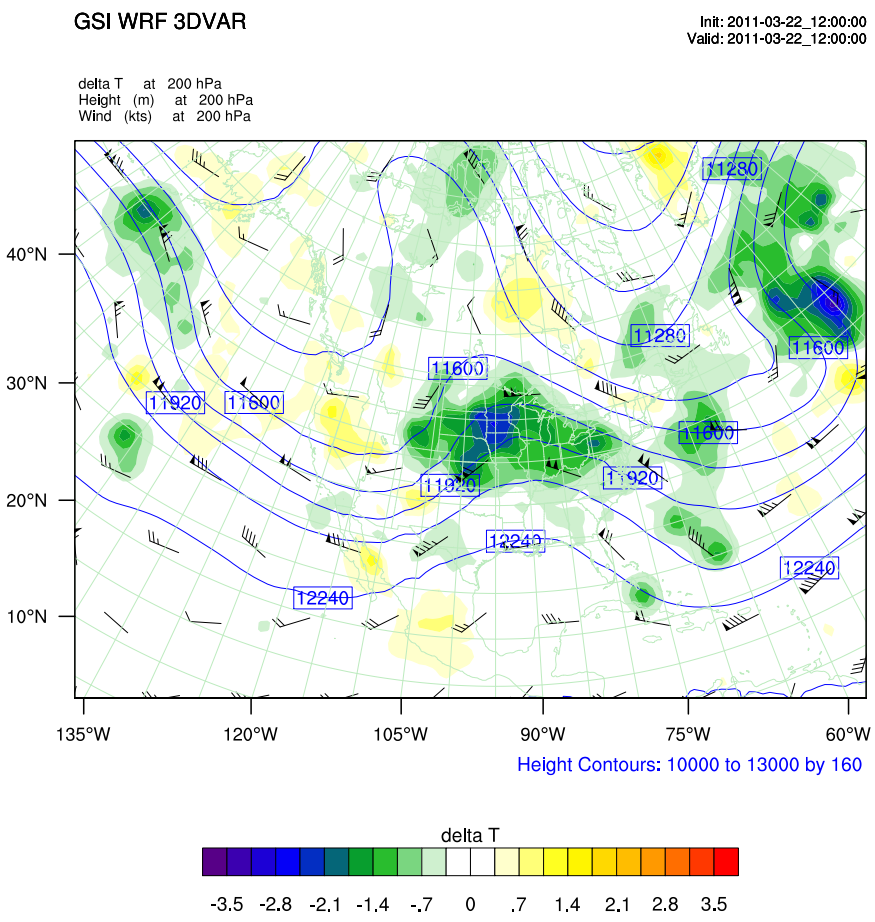
grtest: gradient T1= 0.9999606139497992
grtest: gradient T1= 1.000010809460202
grtest: gradient T1= 1.000001562918812
grtest: gradient T1= 0.9999995022610166
grtest: gradient T1= 0.9999934563438821
grtest: gradient T1= 0.9999345317021266
grtest: gradient T1= 0.9993453113466101
grtest: gradient T1= 0.9934531128869484

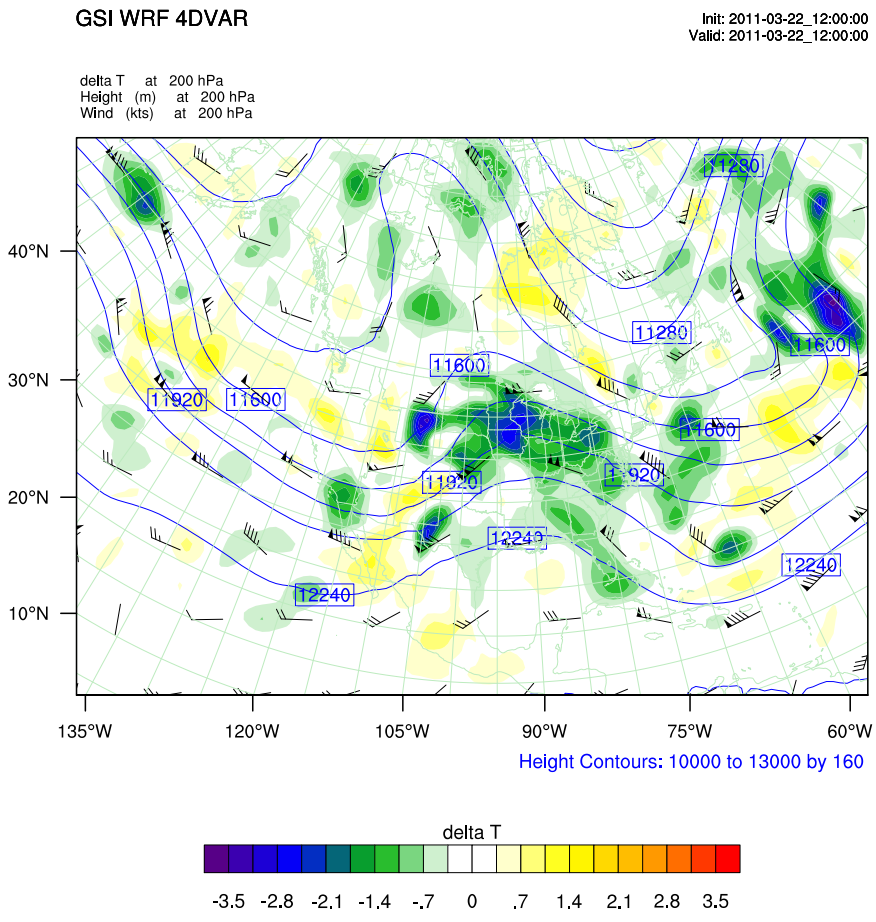
Minimization (pcgsoi)

- 4DVAR has smaller initial cost function.
- 4DVAR has smaller final cost function.
- Note: 4DVAR uses more data than 3DVar.

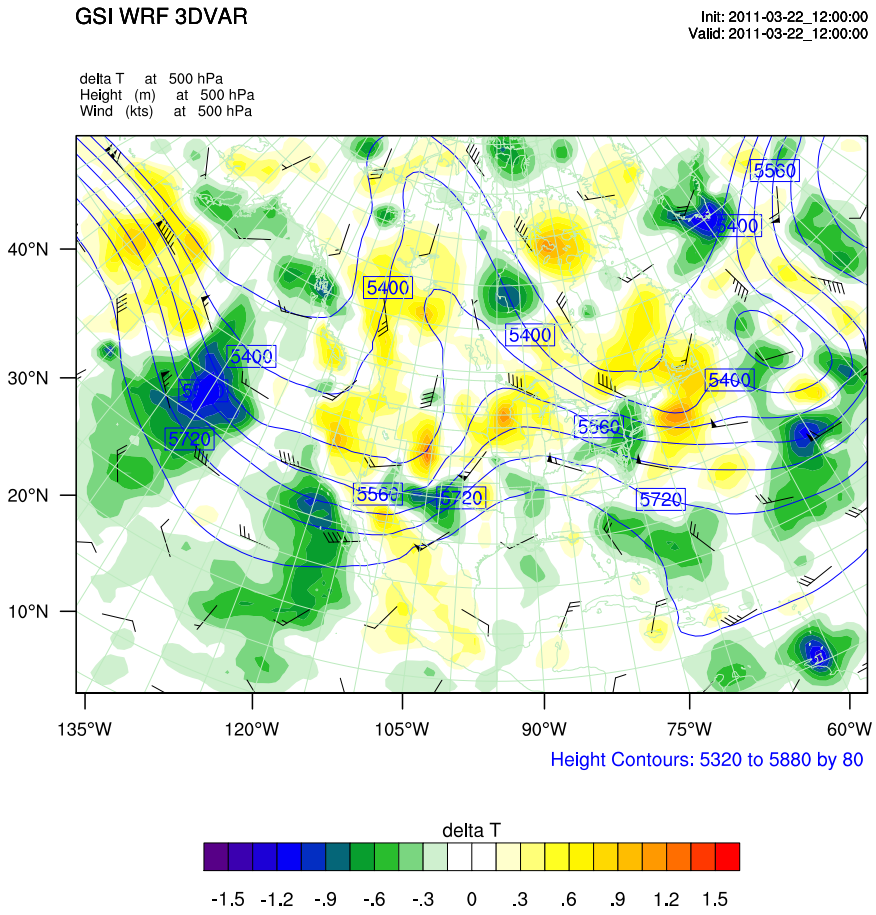


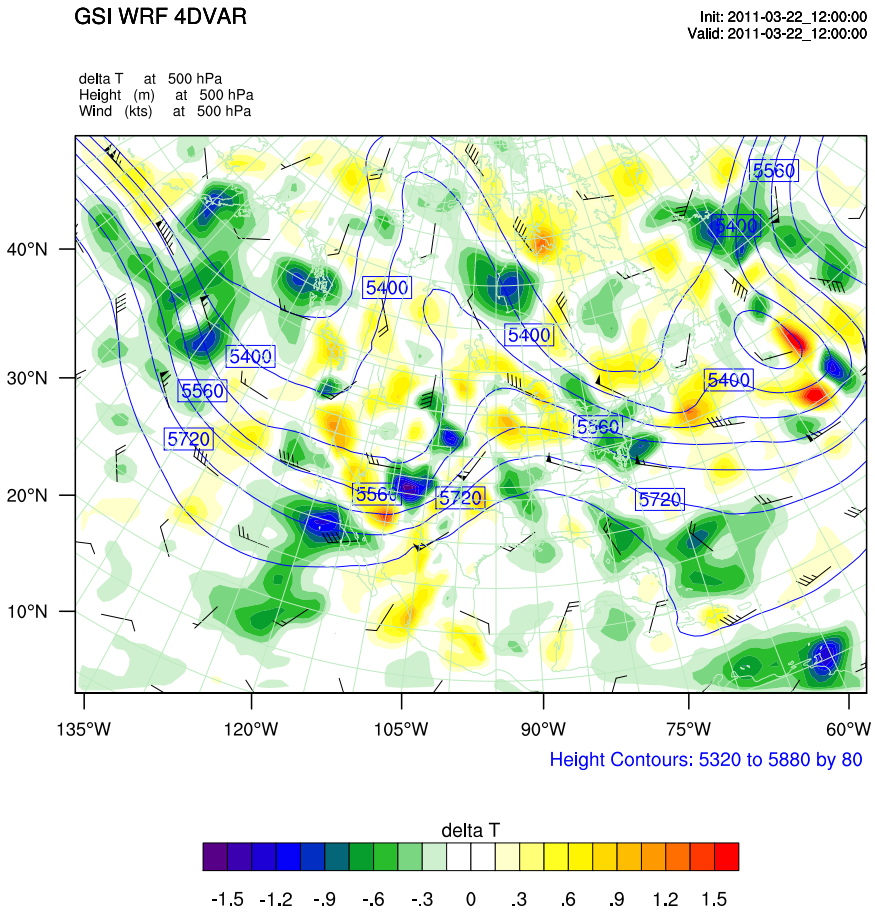
Increments (Temperature at 200hPa)



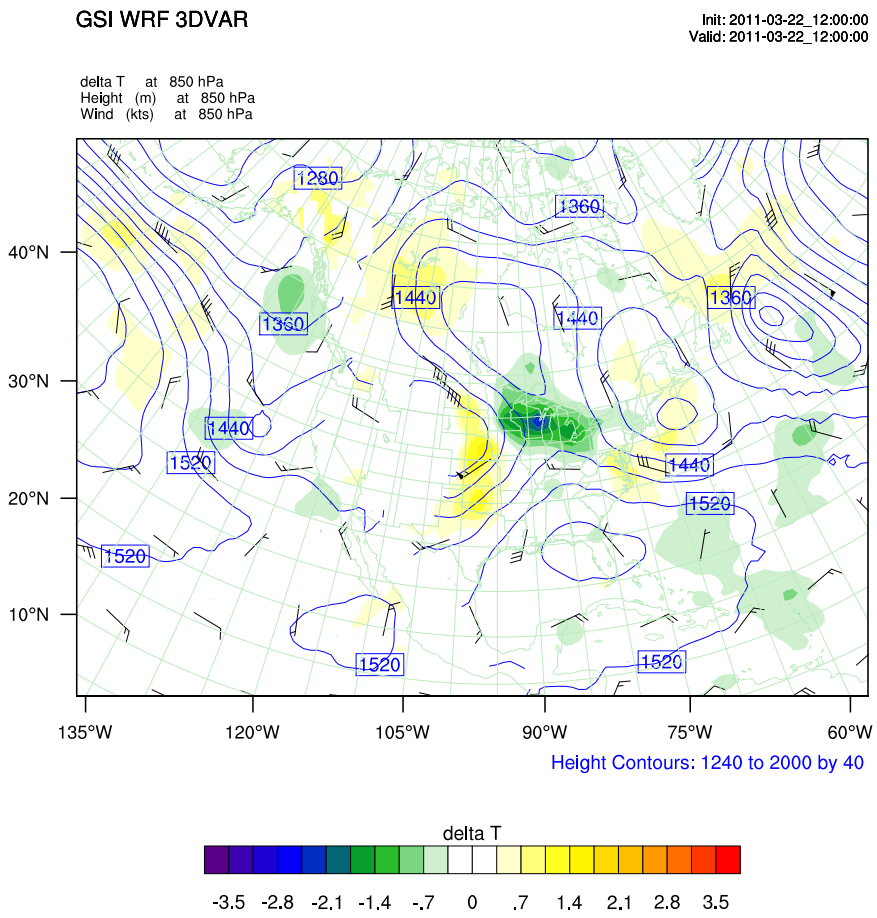


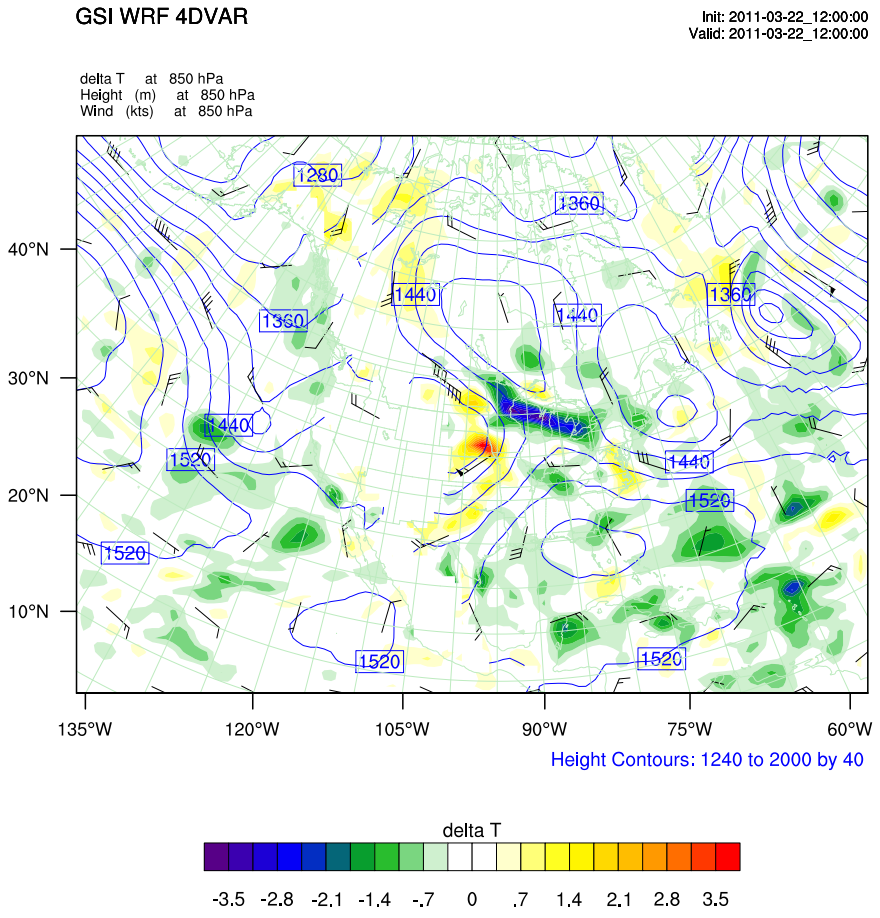
Increments (Temperature at 500hPa)



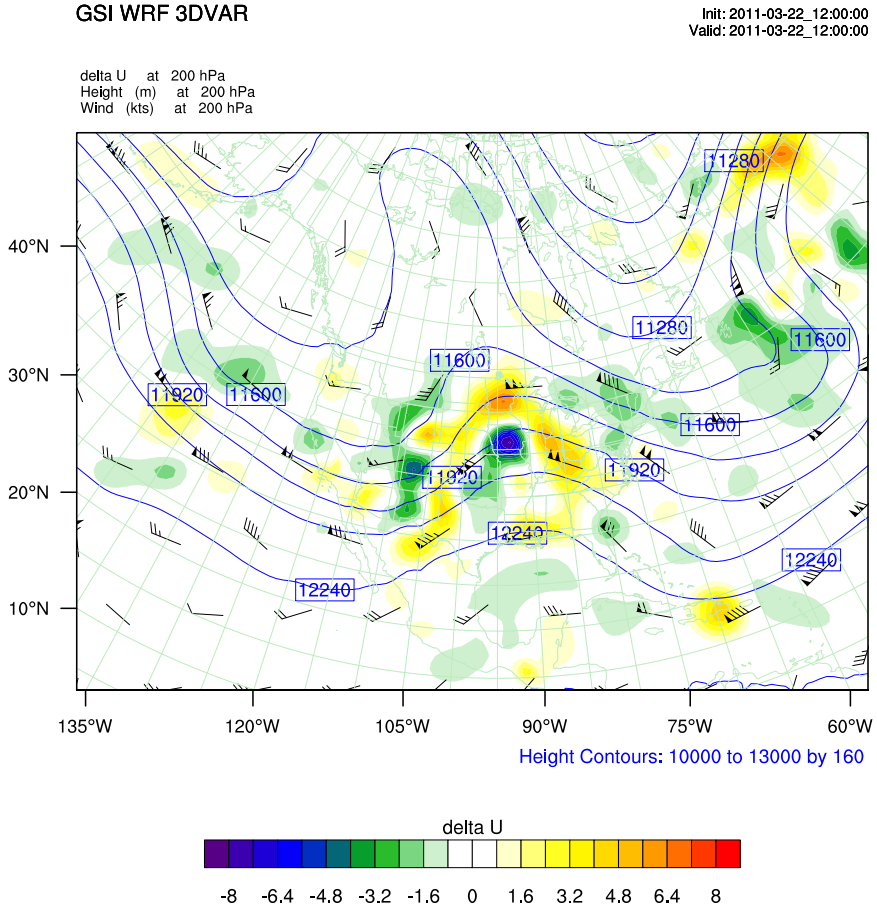


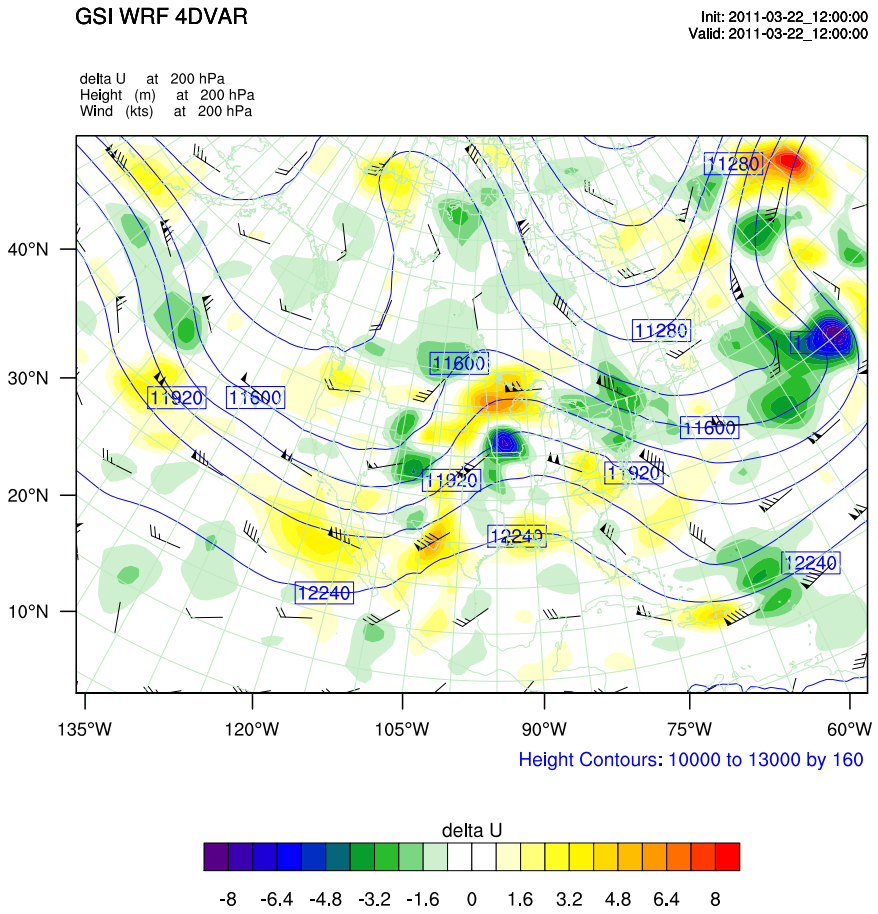
Increments (Temperature at 850hPa)



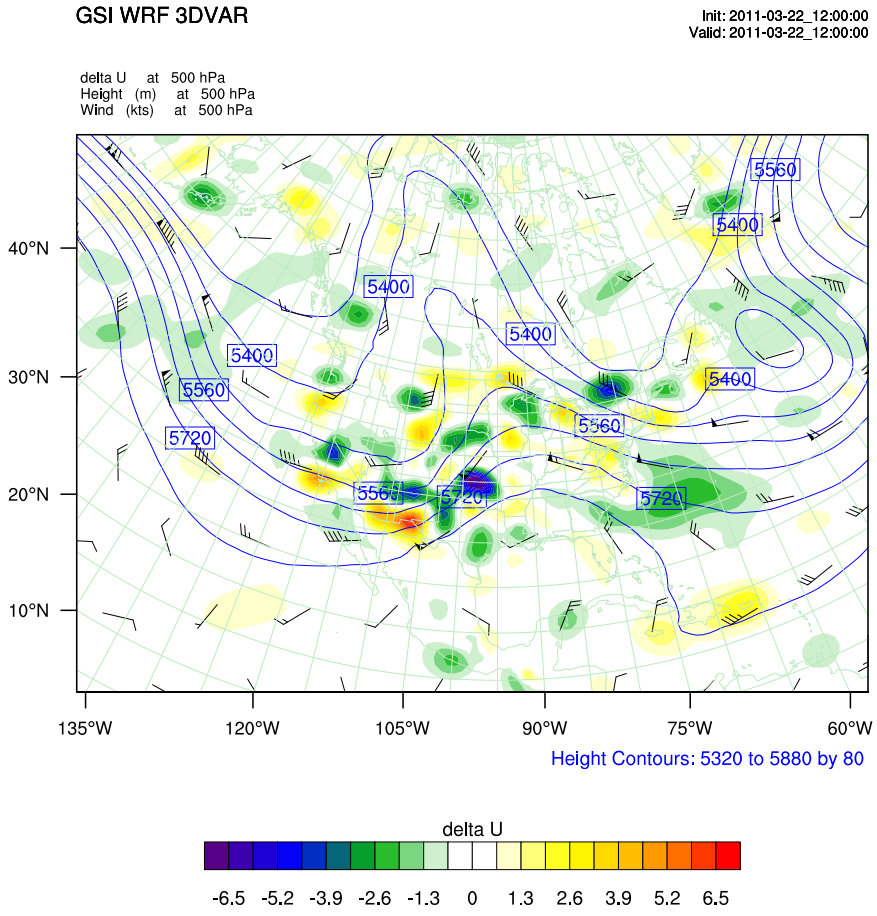


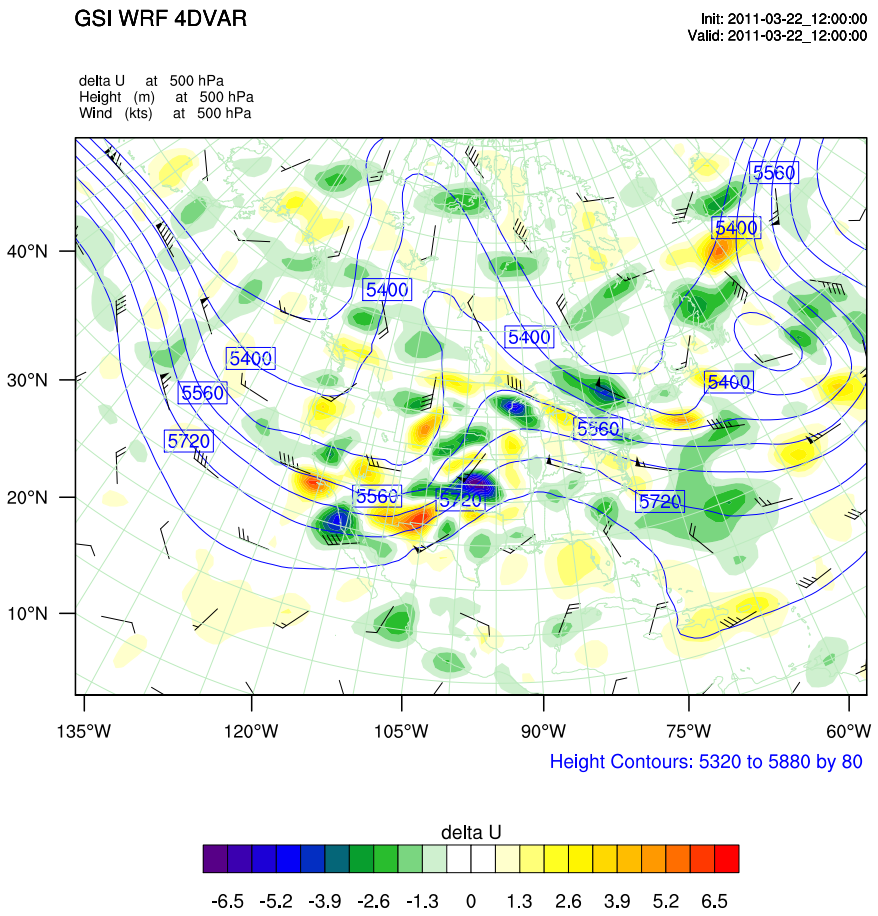
Increments (U component at 200hPa)



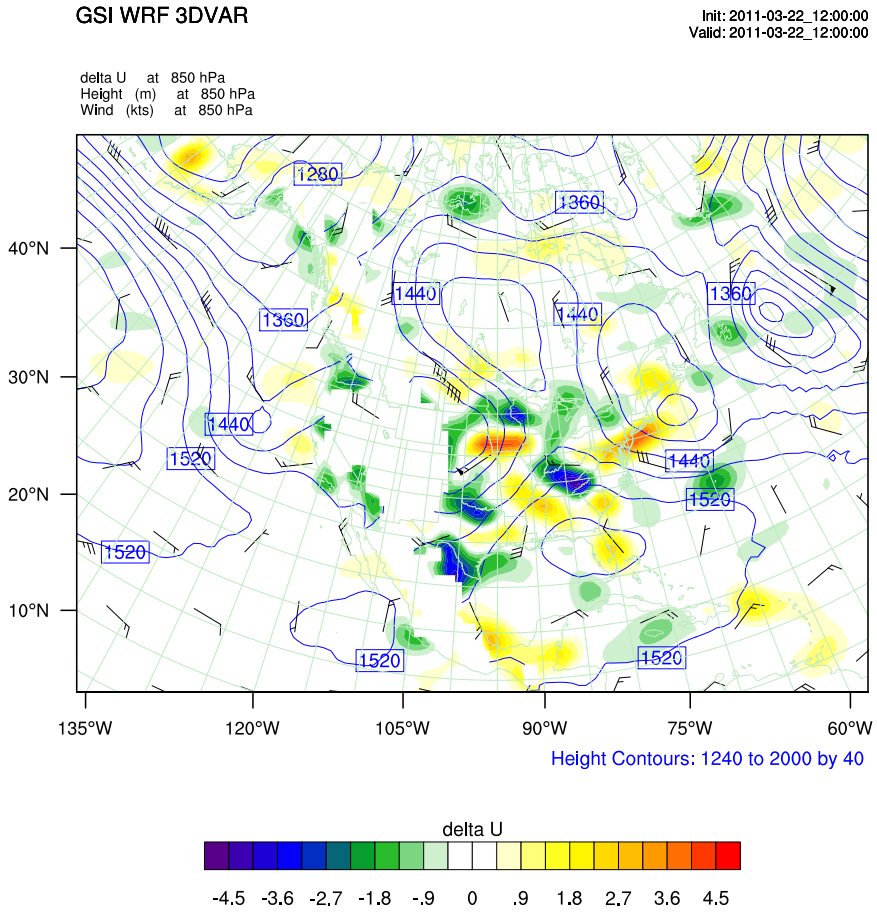


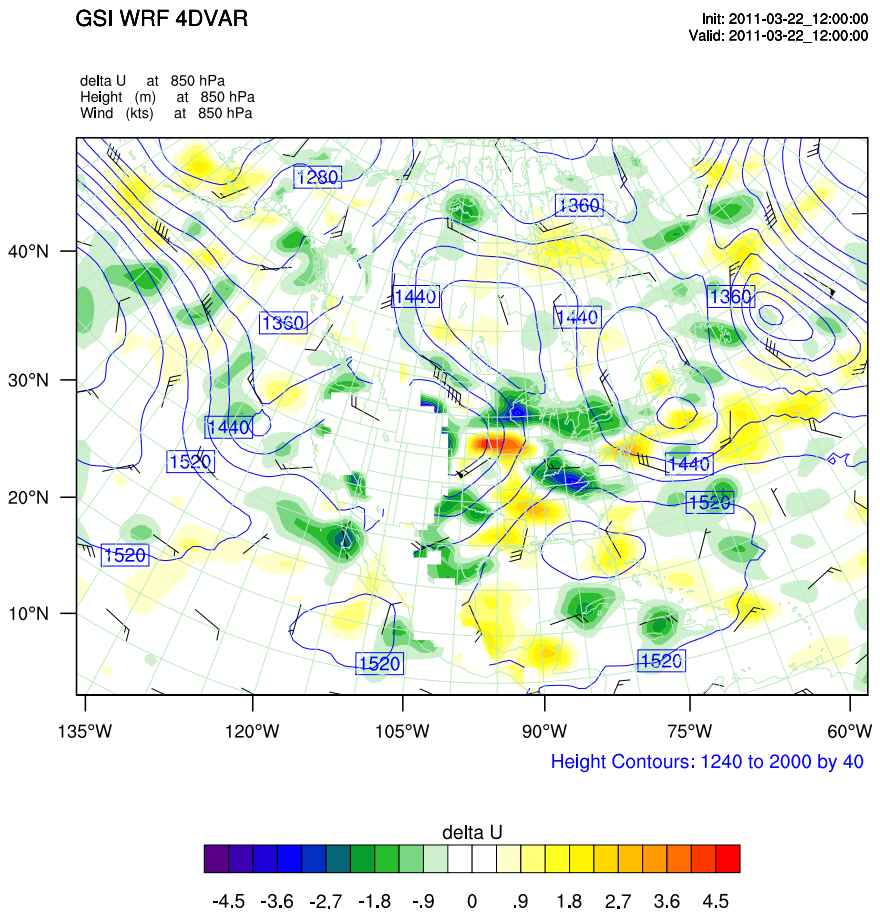
Increments (U component at 500hPa)



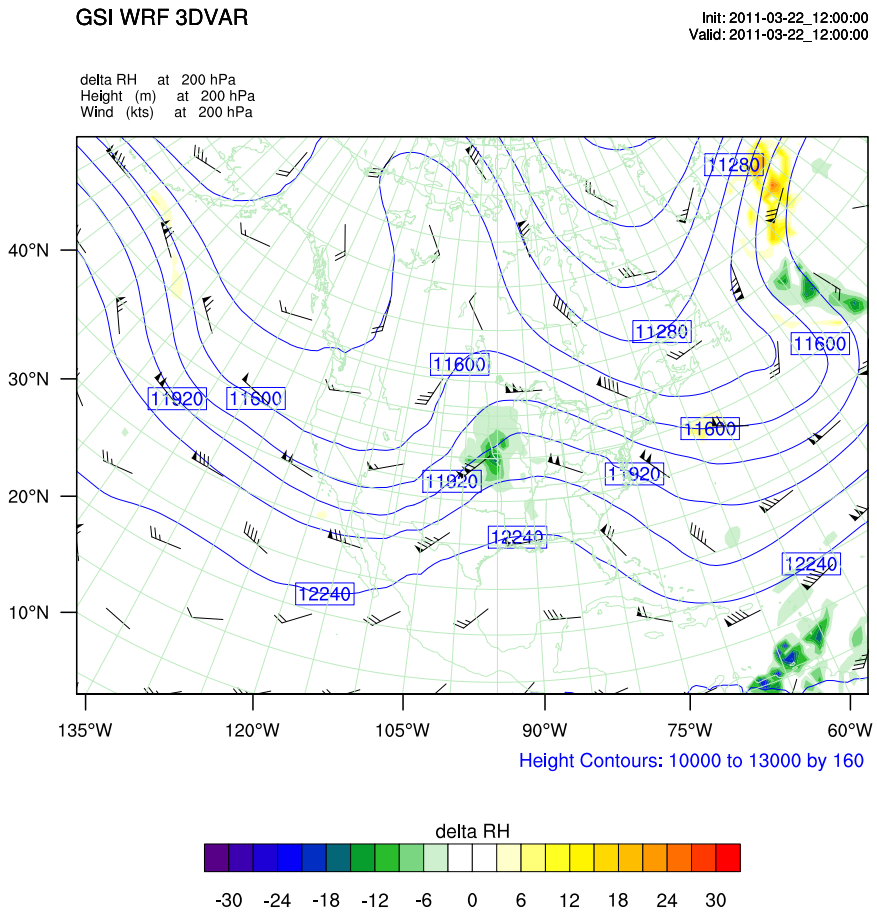


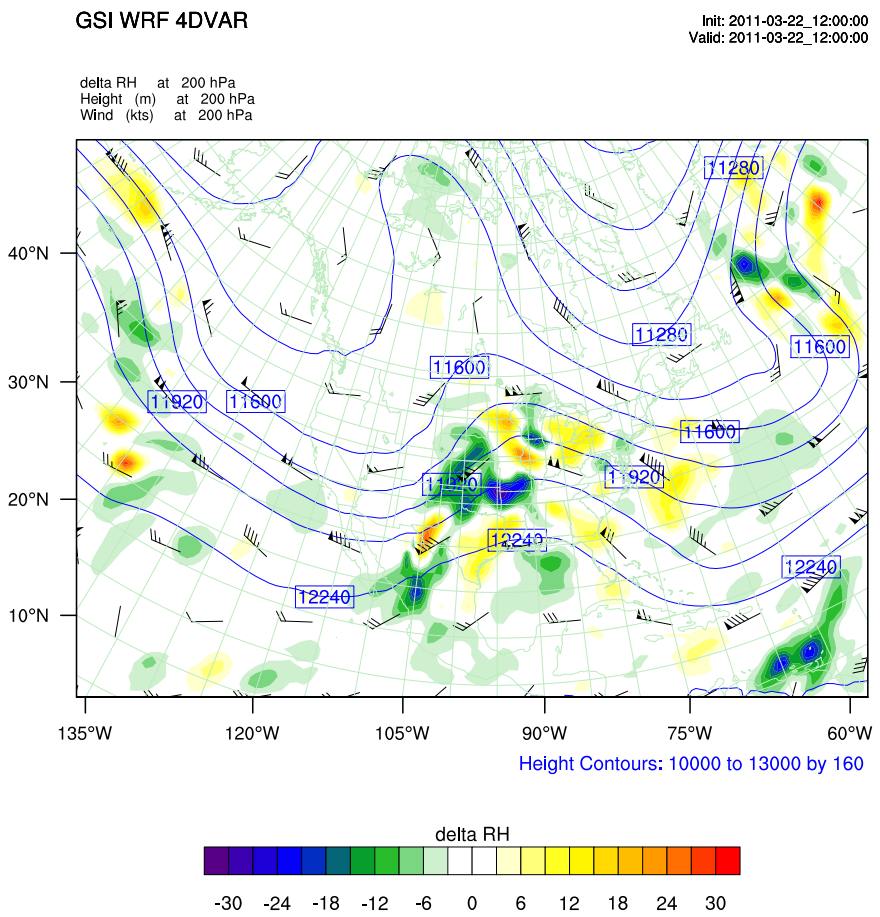
Increments (U component at 850hPa)



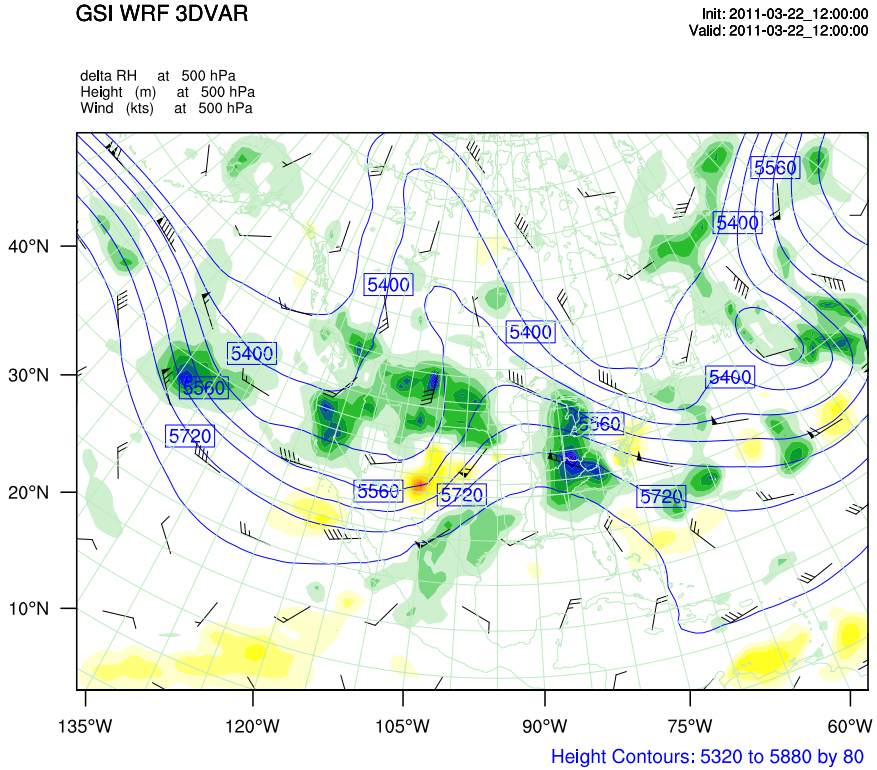


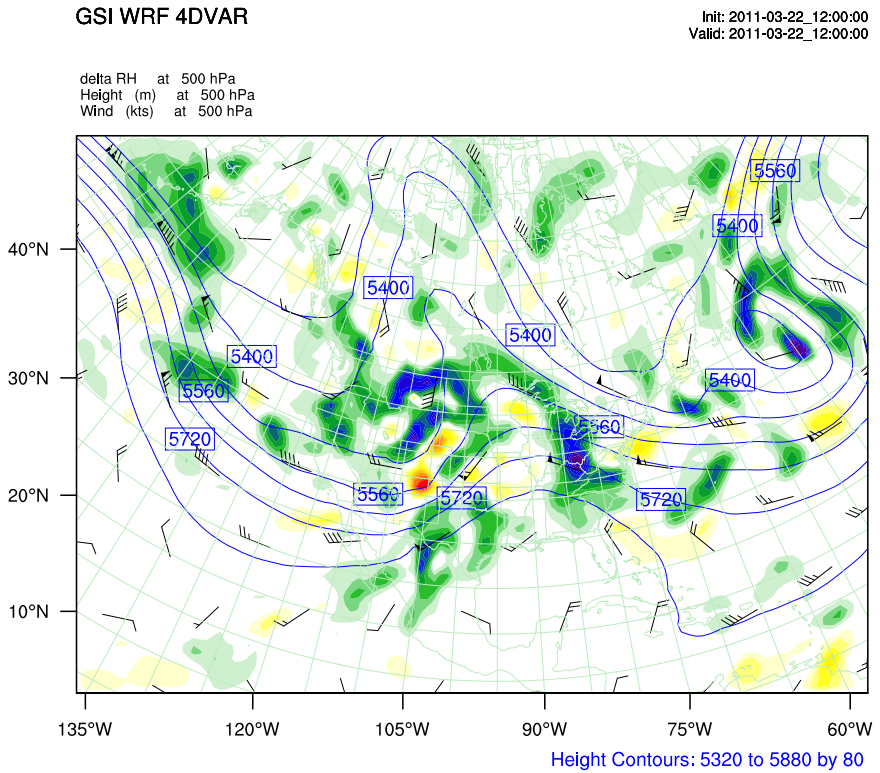
Increments (RH at 200hPa)



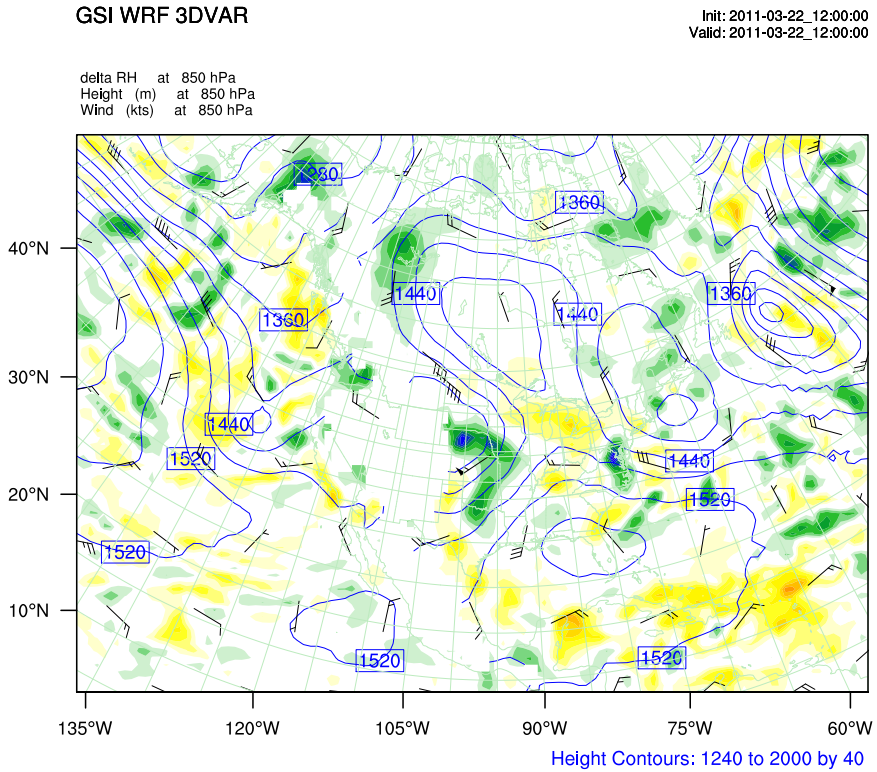


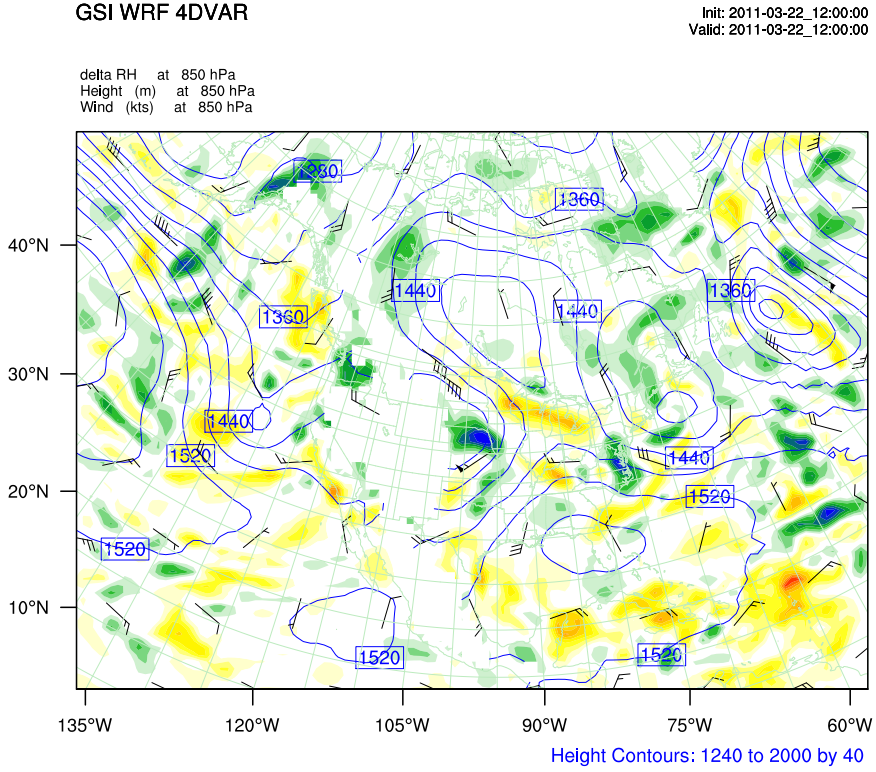
Increments (RH at 500hPa)





Increments (RH at 850hPa)

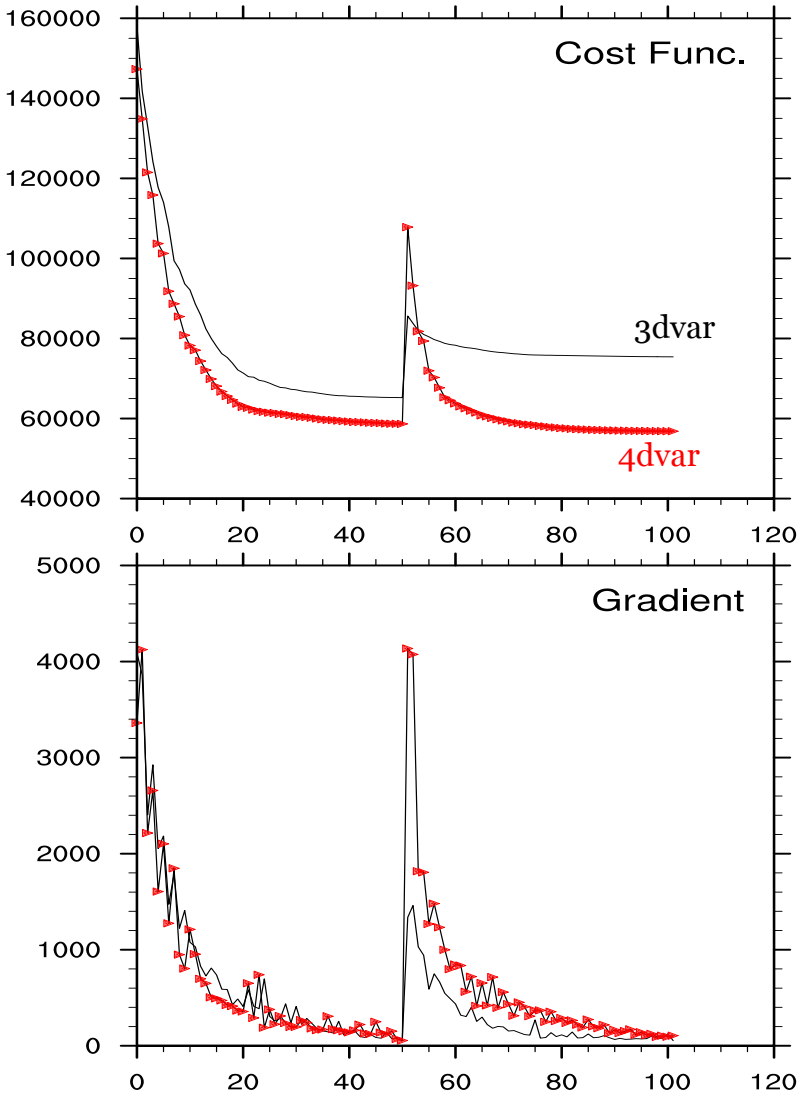




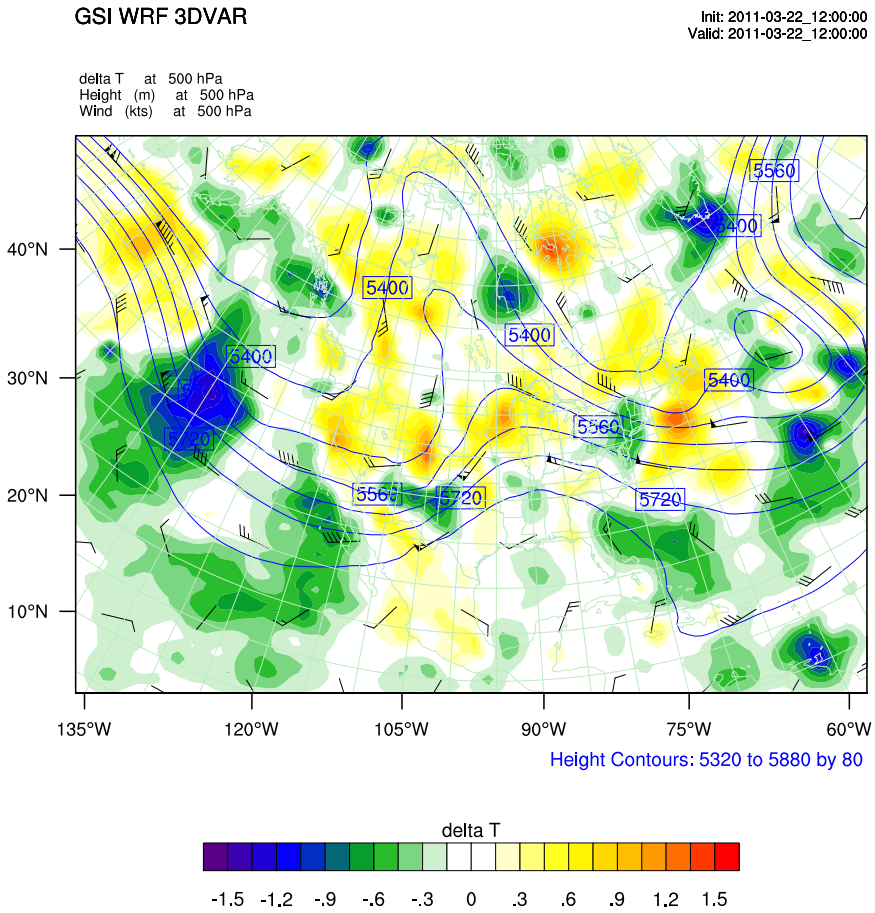
2 outer-loops Minimization (pcgsoi, inner loops: 50/50)

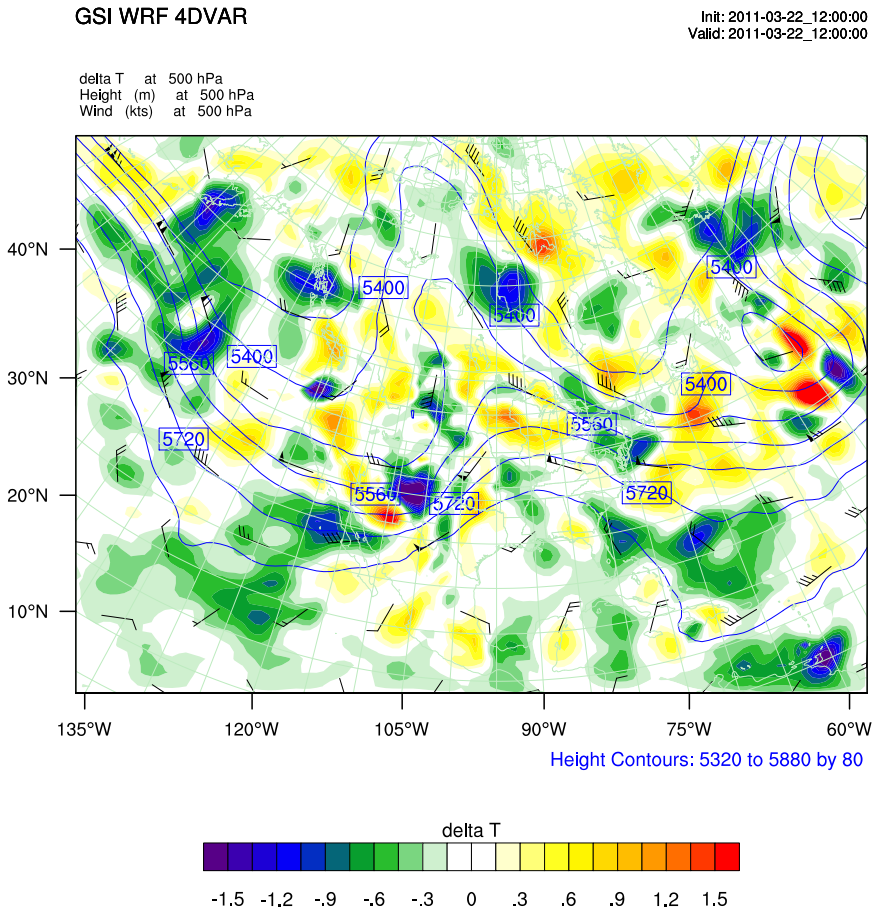
Obs. Number Statistics

Obs.	3DVAR		4DVAR	
	1st outer	2nd outer	1st outer	2nd outer
Surf. Press.	13852	13859	13855	13861
Temp.	9233	9233	9233	9233
Wind	25582	25636	25590	25636
Moist.	4210	4210	4210	4210
GPS	8707	8837	8750	8864
Radiance	131221	154758	145356	160179
Total	**192805**	**216533**	**206994**	**221983**

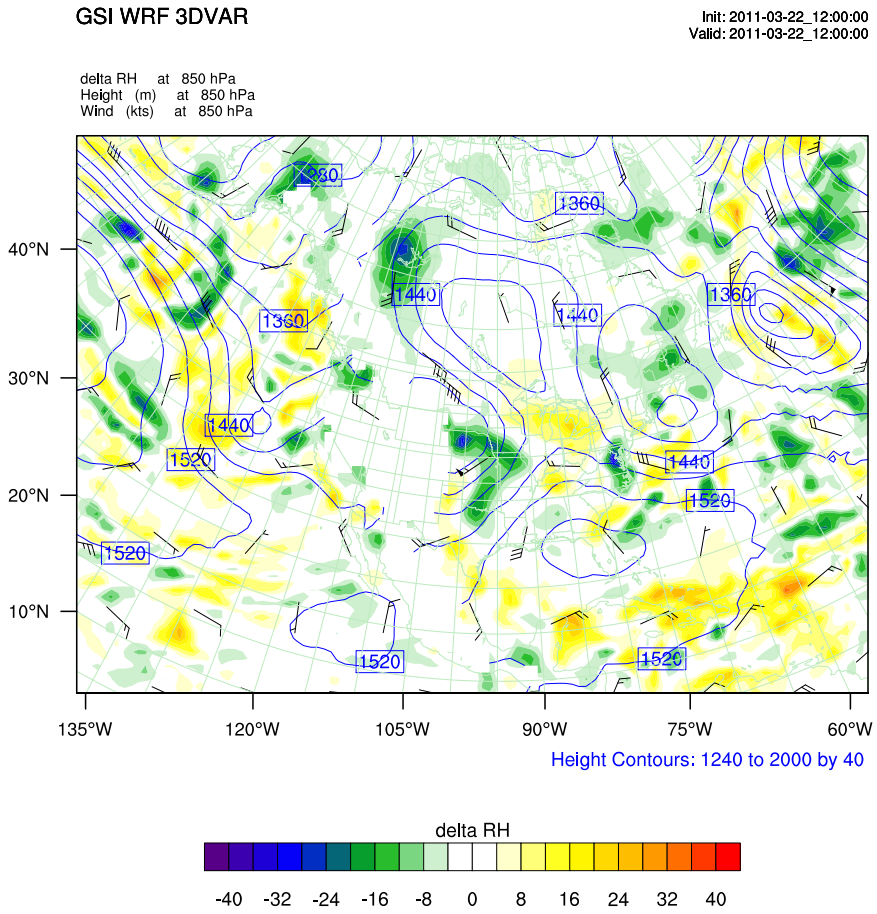


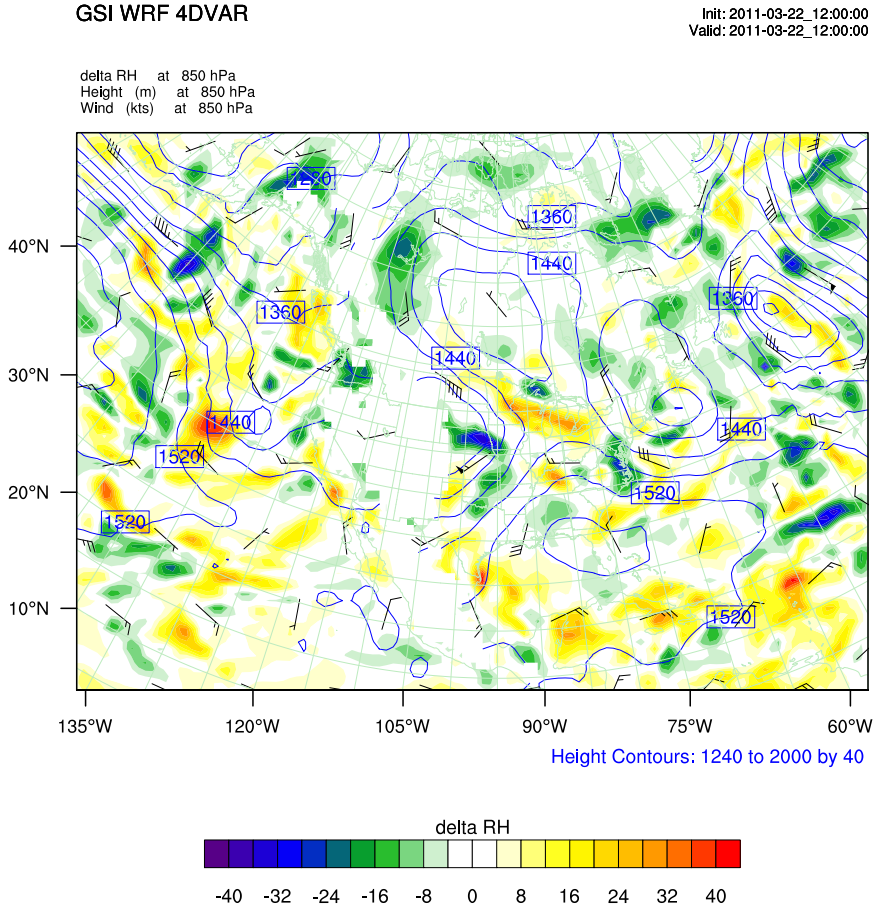
Increments after 2 outer loops (Temperature at 500hPa)





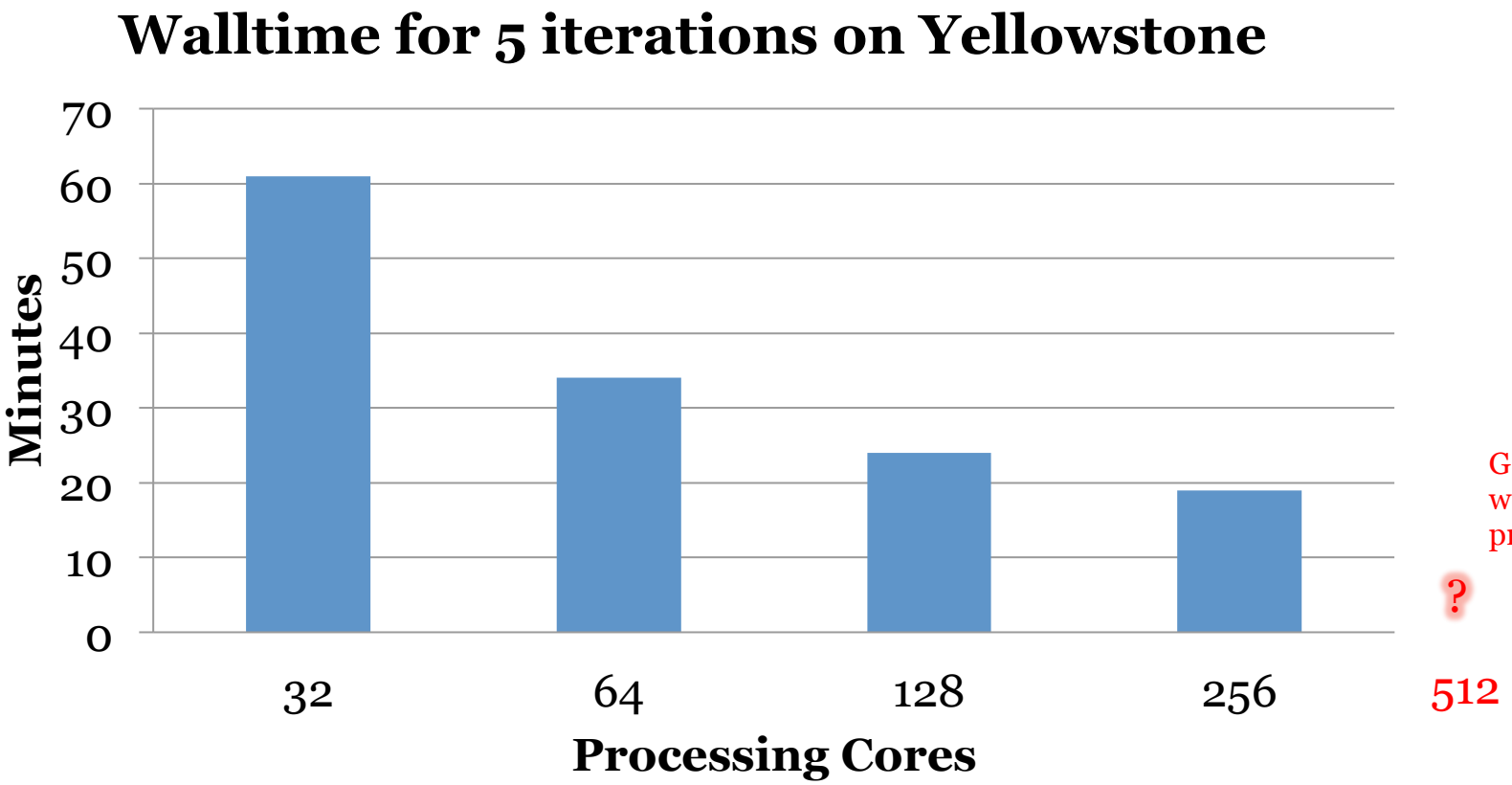
Increments after 2 outer loops (RH at 850hPa)





Parallel Performance

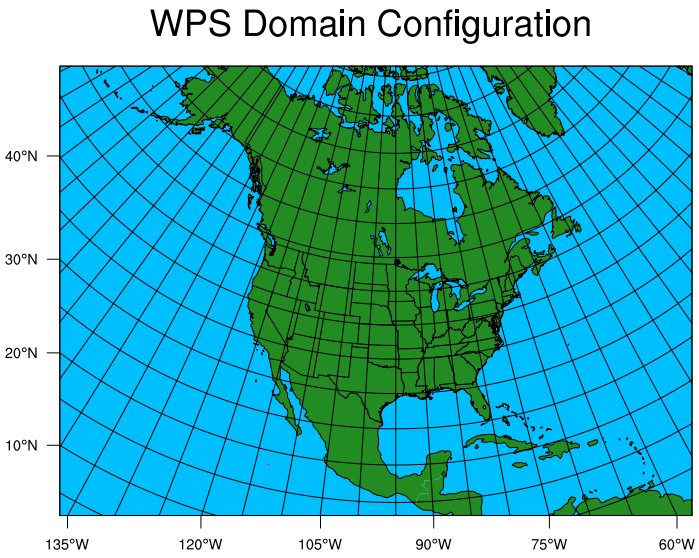
- Domain size:
  - 349×247×L51
- Domain horizontal resolution:
  - 30km
- Assimilation window
  - 6 hours (2209 to 2215)



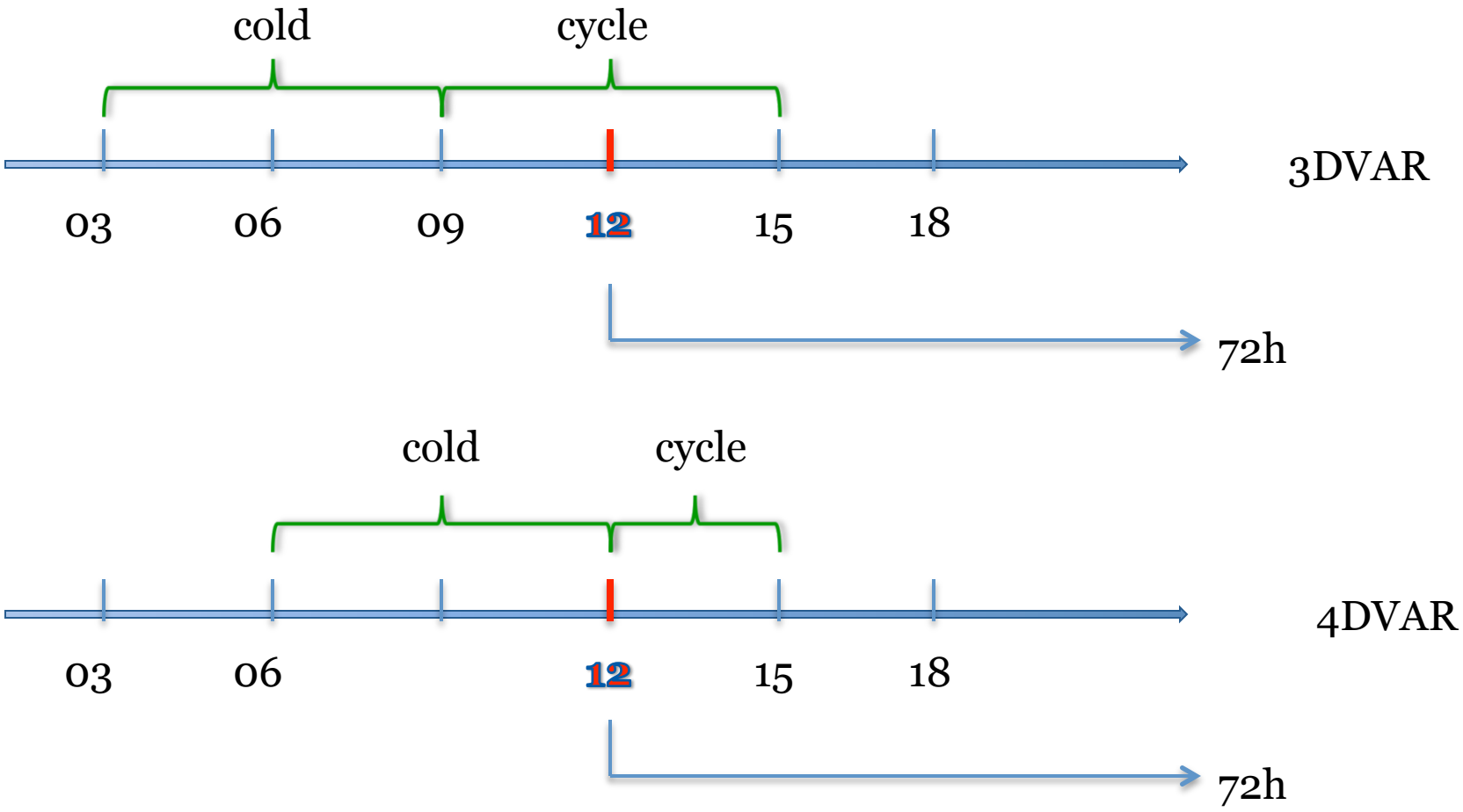
GSI does not allow we use more processing tasks.

One-month regression test

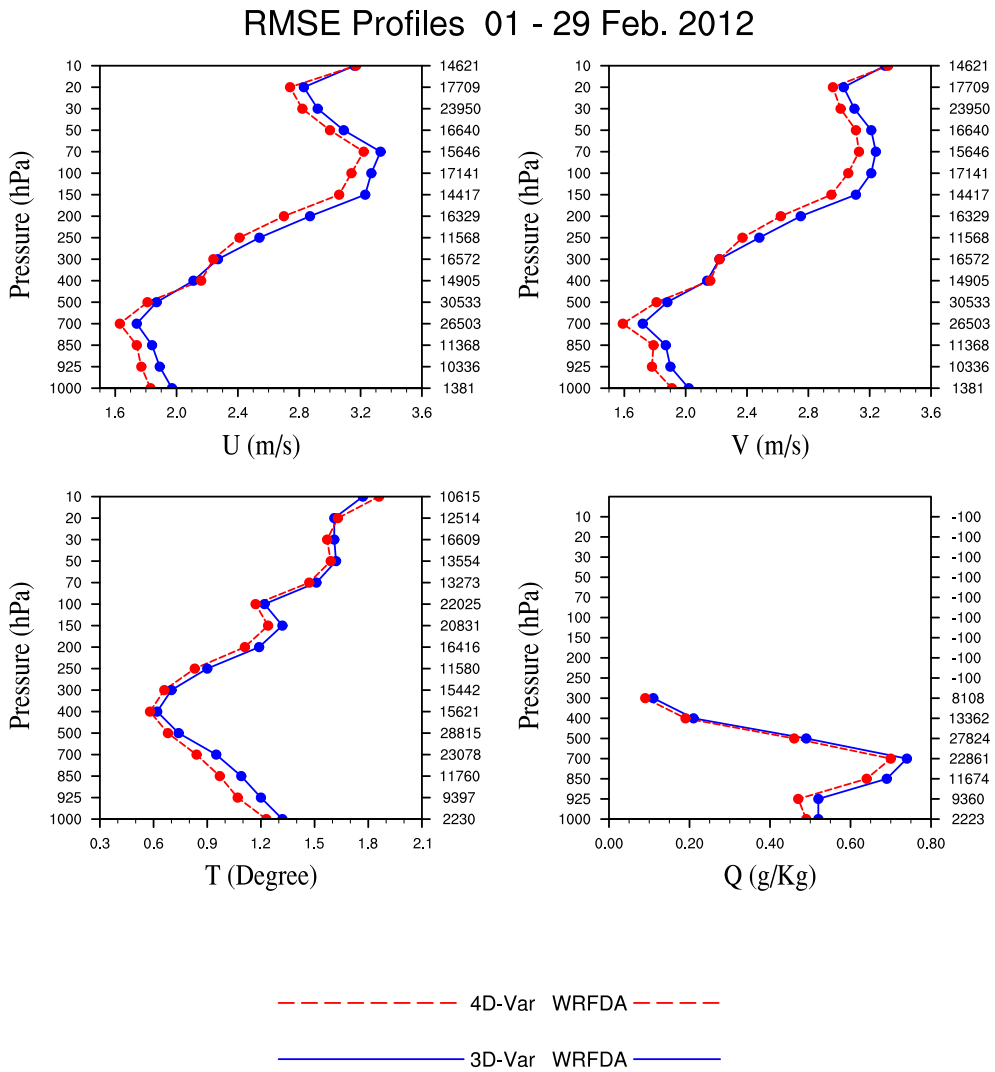
- Analysis time:
  - 2012•02 12Z
- Domain size:
  - 175×124×L51
- Domain horizontal resolution:
  - 60km
- Conventional data only



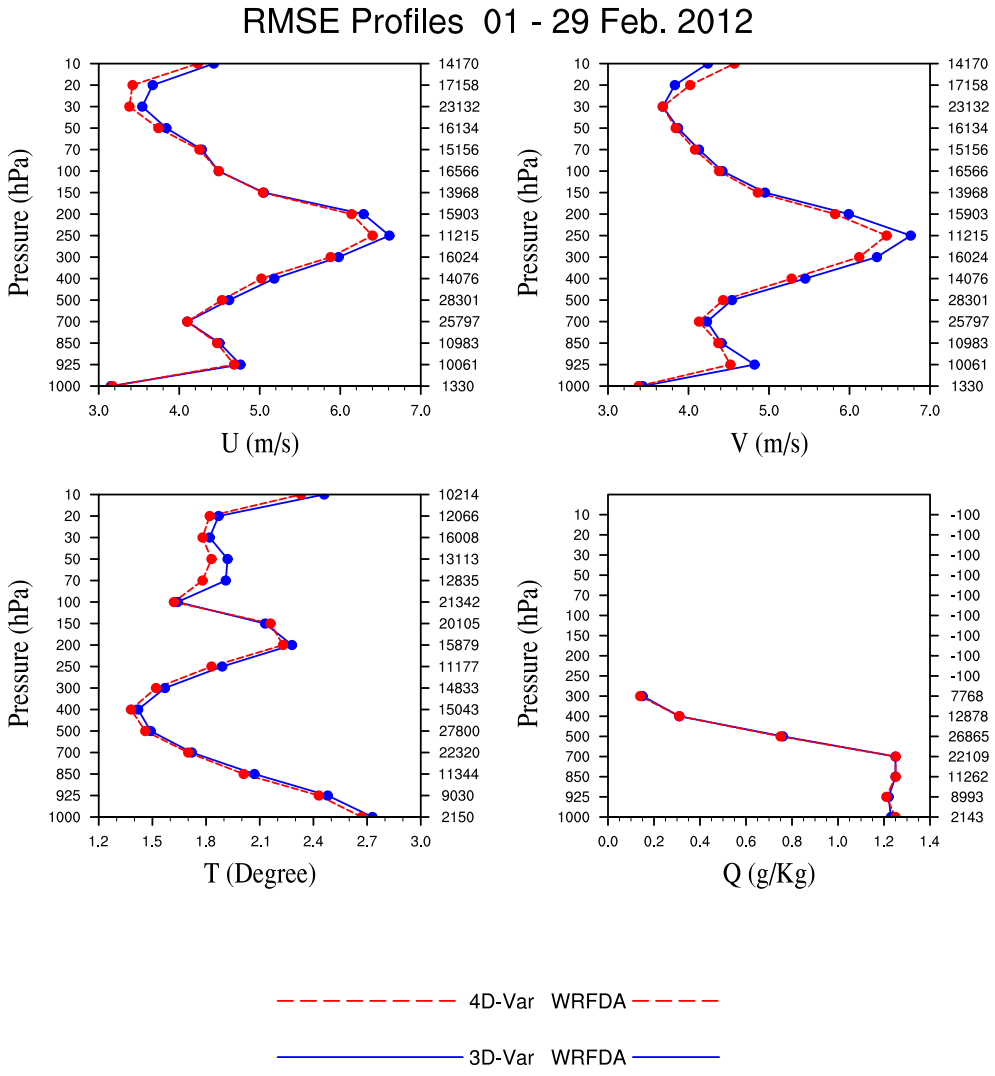
Experiments design:



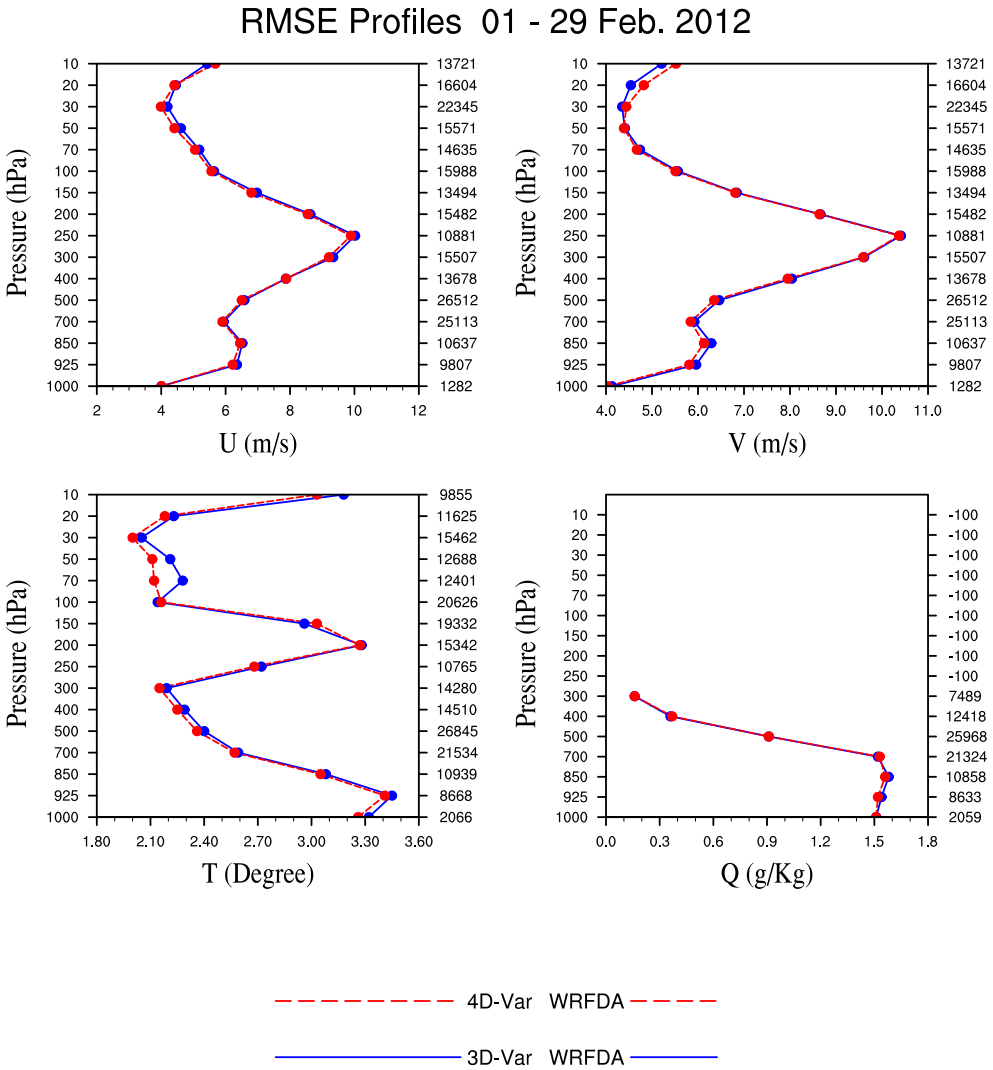
Aggregated RMSE Profiles—oH



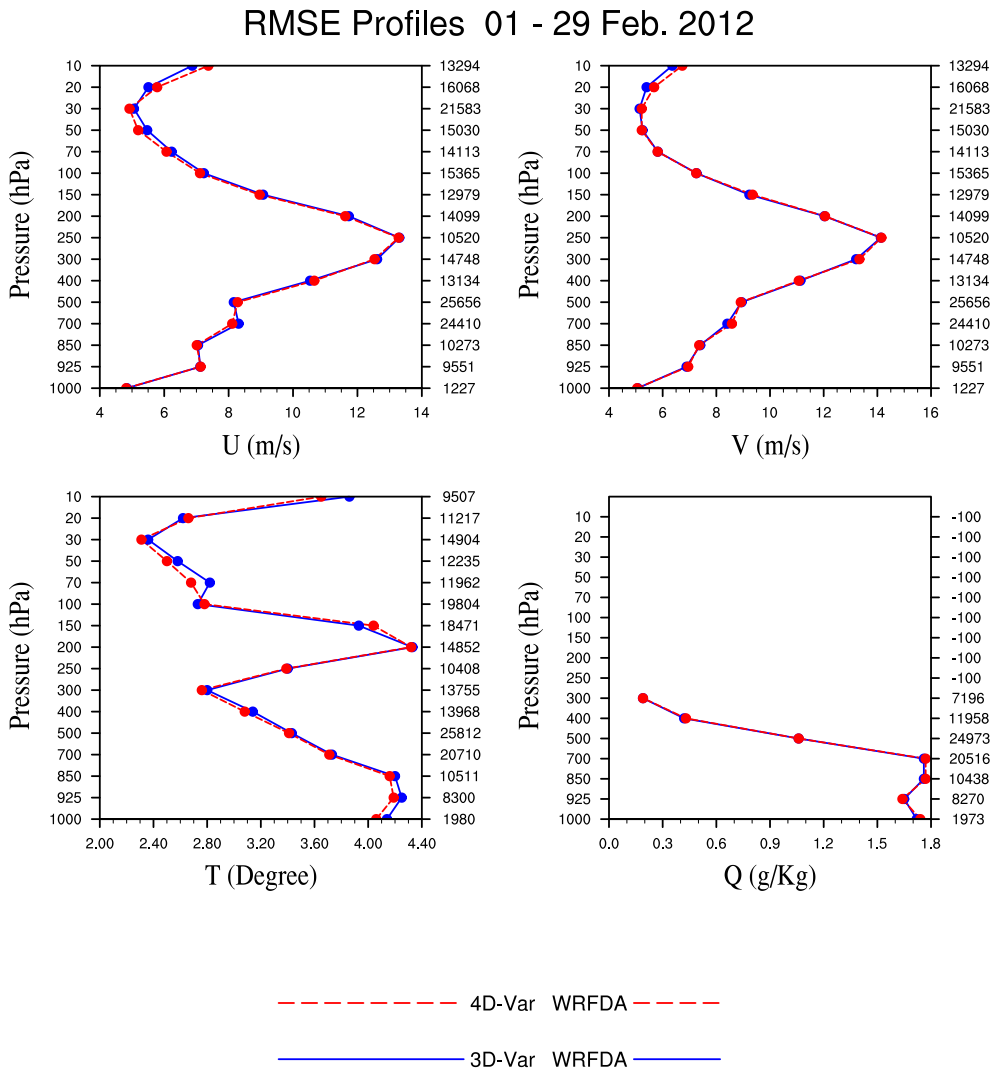
Aggregated RMSE Profiles—24H



Aggregated RMSE Profiles—48H



Aggregated RMSE Profiles—72H



Outline

- Linking GSI with WRF TLM and ADM
- GSI-based WRF 4DVAR
- What to do ……

Proposed changes

- M src/main/makefile_DTC
- A + src/main/read_wrf_mass_files.F90
- M src/main/wrf_binary_interface.F90
- M src/main/gridmod.F90
- D src/main/read_wrf_mass_files.f90
- M src/main/stpcalc.f90
- M src/main/model_tl.F90
- M src/main/model_ad.F90
- M src/main/stub_pertmod.F90
- M src/main/fill_mass_grid2.f90
- A + src/main/gesinfo.F90
- M src/main/adjtest.f90
- M src/main/read_prepbufr.f90
- M src/main/gsi_4dvar.f90
- A src/main/wrf_pertmod.F90
- D src/main/gesinfo.f90
- M src/main/wrwrfmassa.F90
- M src/main/wrf_netcdf_interface.F90
- M src/main/gsimod.F90
- M src/main/unfill_mass_grid2.f90
- M src/main/compute_derived.f90
- M src/main/read_wrf_mass_guess.F90
- M src/main/Makefile.dependency
- M src/main/enorm_state.f90
- M src/main/evaljcdfi.f90
- M src/main/obsmod.F90
- M arch/Config.pl
- A arch/postamble.4dvar

Major changes

- Add coupler interfaces of WRF+, use **#ifdef WRF_PERT** to isolate the changes for WRF 4DVar from NASA GMAO 4Dvar.

```
M       src/main/model_tl.F90
M       src/main/model_ad.F90
M       src/main/stub_pertmod.F90
A       src/main/wrf_pertmod.F90
```

- Add configure option for GSI/WRF 4DVar compilation. Compiling WRFPLUS firstly; pointing the envir. var. **WRF_DIR** to the PATH of WRFPLUS; set the envir. var. **VAR4D=1**

```
M       arch/Config.pl
A       arch/postamble.4dvar
```

Minor changes

- Add new namelist variable in setup : **nhr_inc_update**, which determines which guess file will be updated after the analysis. For example, **nhr_inc_update=1** updates the firstguess at the beginning of the time window (wrf_inou1), **nhr_inc_update=4** updates the firstguess at hour 3 (wrf_inou4) for **nhr_obsbin=1**

```
M      src/main/gsi_4dvar.f90
M      src/main/gsimod.F90
A +    src/main/read_wrf_mass_files.F90
D      src/main/read_wrf_mass_files.f90
M      src/main/wrf_netcdf_interface.F90
```

Minor changes --- continued

- Add new namelist variable : **nhr_inc_update**, which determines which guess file will be updated after analysis. For example, **nhr_inc_update=1** updates the firstguess at the beginning of the time window (wrf_inou1), **nhr_inc_update=4** updates the firstguess at hour 3 (wrf_inou4) for **nhr_obsbin=1.** Isolated with #ifdef

```
M       src/main/gsi_4dvar.f90
M       src/main/gsimod.F90
M       src/main/wrwrfmassa.F90
```

Minor changes --- continued

- Modified to accommodate adjoint test for GSI-based WRF 4DVar.

  Isolated with #ifdef

  A + src/main/adjtest.F90
  D src/main/adjtest.f90

- Add corresponding adjoint codes for fill and unfill.

  M src/main/fill_mass_grid2.f90
  M src/main/unfill_mass_grid2.f90

- Turn off trajectory init. when oberver is true. Isolated with #ifdef

  M src/main/compute_derived.f90

- Remove some WRF interfaces, which will be available in WRF+ lib..

  Isolated with #ifdef.

  M src/main/wrf_binary_interface.F90

Minor changes --- continued

- For GSI-based WRF 4DVar, we always use the guess file at the beginning of the assimilation window (wrf_inou1) to determine the regional_time. Isolated with #ifdef

```
M       src/main/gridmod.F90
```

- For GSI-based WRF 4DVar, **min_offset** is forced to be zero and the **nhr_assimilation** is used to calculate the end if the window. Isolated with #ifdef

```
M       src/main/gsimod.F90
A  +    src/main/gesinfo.F90
D       src/main/gesinfo.f90
```

Minor changes --- continued

- For GSI-based WRF 4DVar, we always use the guess file at the beginning of the assimilation window (wrf_inou1) as the first guess. Isolated with #ifdef

  M src/main/wrf_netcdf_interface.F90

- Allow GSI-based WRF 4DVar to read more than one observation files for each obs. type. Isolated with #ifdef

  M src/main/obsmod.F90

- Make dependency and compilation updates

  M src/main/Makefile.dependency
  M src/main/makefile_DTC

Bug fixes

- **ier** was used before being initialized

  M src/main/read_wrf_mass_guess.F90

```
Index: src/main/read_wrf_mass_guess.F90
===================================================================
--- src/main/read_wrf_mass_guess.F90                    (revision 1061)
+++ src/main/read_wrf_mass_guess.F90                    (working copy)
@@ -1205,6 +1205,7 @@
     if (nguess>0) then
!     Get pointer for each of the hydrometeors from guess at time index "it"
      it=ntguessig
+      ier = 0
       call GSI_BundleGetPointer ( GSI_MetGuess_Bundle(it), 'ql', ges_qc, istatus );ier=ier+istatus
       call GSI_BundleGetPointer ( GSI_MetGuess_Bundle(it), 'qi', ges_qi, istatus );ier=ier+istatus
       call GSI_BundleGetPointer ( GSI_MetGuess_Bundle(it), 'qr', ges_qr, istatus );ier=ier+istatus
```

- **disterrmax** was used before being initialized

  M src/main/read_prepbufr.f90

```
Index: src/main/read_prepbufr.f90
===================================================================
--- src/main/read_prepbufr.f90          (revision 1061)
+++ src/main/read_prepbufr.f90          (working copy)
@@ -624,6 +624,7 @@
     nmsg = 0
     icntpnt=0
     icntpnt2=0
+    disterrmax=-9999.0
     loop_msg: do while (ireadmg(lunin,subset,idate)== 0)
       if(.not.use_prepb_satwnd .and. trim(subset) =='SATWND') cycle loop_msg
       nmsg = nmsg+1
```

Bug fixes --- continued

- **psum** was calculated by processor zero only, but used by all processors immediately. Let it being calculated by all processors. Also, add the codes to calculate the energy norm for WRF_MASS_CORE in enorm_state.f90

M src/main/enorm_state.f90
M src/main/evaljcdfi.f90

```
Index: src/main/stpcalc.f90
===================================================================
--- src/main/stpcalc.f90                        (revision 1061)
+++ src/main/stpcalc.f90                        (working copy)
@@ -542,11 +542,11 @@

   stpinout=stp(istp_use)
 ! Estimate terms in penalty
+  do i=1,ipen
+    psum(i)=pbc(1,i)+(stp(iis-1)-stp(iis))*(2.0_r_quad*bsum(i)+ &
+                (stp(iis-1)-stp(iis))*csum(i))
+  end do
   if(mype == 0)then
-    do i=1,ipen
-      psum(i)=pbc(1,i)+(stp(iis-1)-stp(iis))*(2.0_r_quad*bsum(i)+ &
-                  (stp(iis-1)-stp(iis))*csum(i))
-    end do
     write(iout_iter,101) (psum(i),i=1,ipen)
   end if
   pjcostnew(1) = psum(1)                        ! Jb
```

Bug fixes --- continued

- **pjc** (cost function for Jcdfi) was multiplied by 0.5 more than one timer, the other is in grtest.f90. This error leads to gradient test fail with JcDFI.

M src/main/evaljcdfi.f90

```
ndex: src/main/evaljcdfi.f90
===================================================================
--- src/main/evaljcdfi.f90                    (revision 1061)
+++ src/main/evaljcdfi.f90                    (working copy)
@@ -67,7 +67,7 @@
 ! Jc = 1/2 * wgt * sfilter *sfilter
 ! afilter = wgt * sfilter
  call enorm_state(sfilter,pjc,afilter)
-pjc=half_quad*pjc
+!pjc=half_quad*pjc
  if (mype==0) write(6,*)'Jc DFI=',pjc

 ! Adjoint Jc multiplicative factor
```

Next

- JcDFI is only applied on the TLM trajectory of the observational sub-windows right now (only 7 snapshots for 6h window). It should be applied on the TLM trajectory of every time step during the assimilation window.

- Is it possible to relax the limit on the maximum allowable number of processing tasks ?

- Data usage in 3DVar and 4Dvar is suspicious ?

- Multiple incremental 4DVAR.

Thank You

The NESL Mission is:

To advance understanding of weather, climate, atmospheric composition and processes; To provide facility support to the wider community; and,

To apply the results to benefit society.

NCAR is sponsored by the National Science Foundation