WRF Data Assimilation System

Xin Zhang Hui-Chuan Lin Xiang-Yu Huang

NCAR Earth System Laboratory

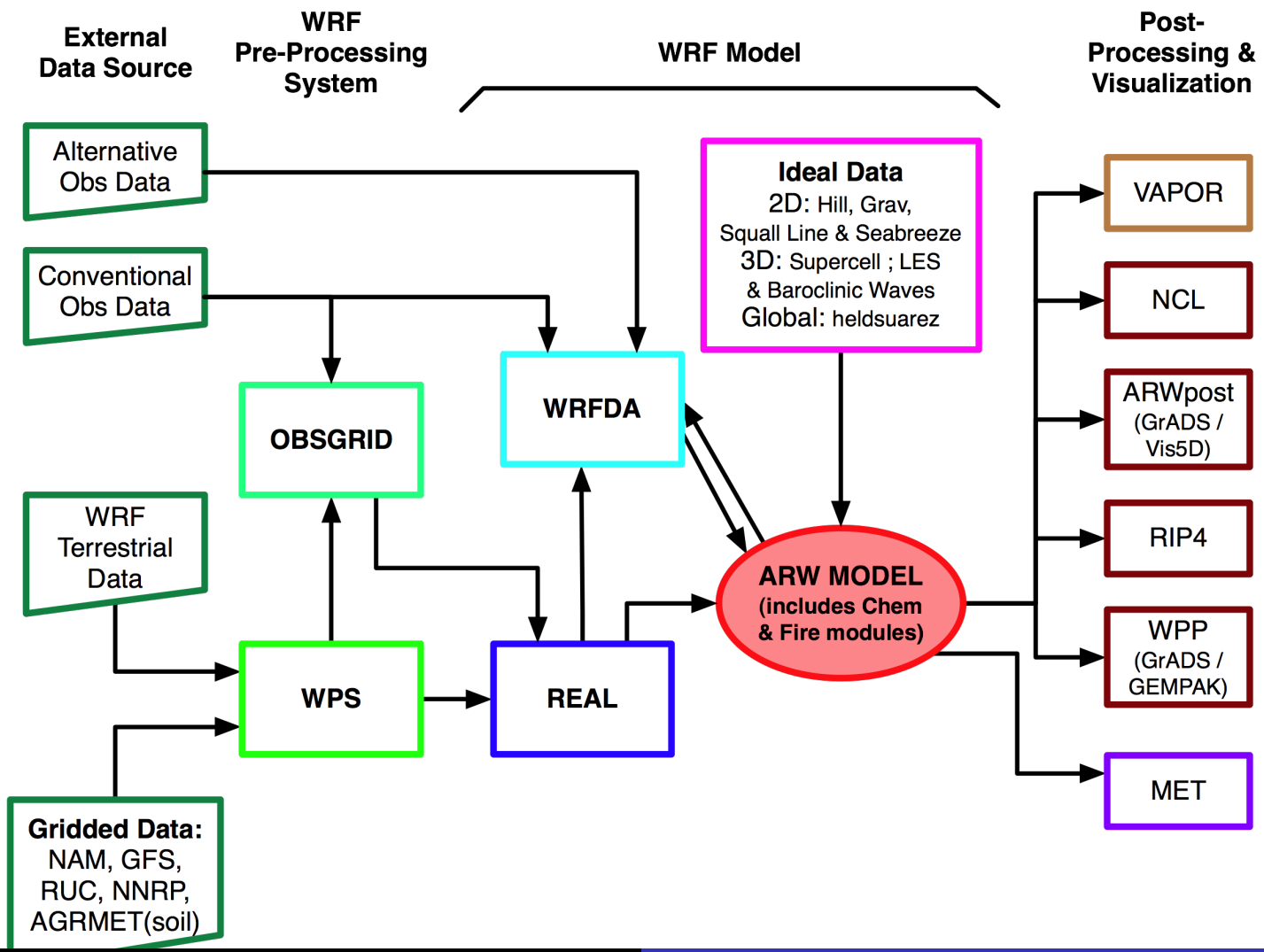
Presented in July 2011, WRFDA Tutorial

NCAR is sponsored by the National Science Foundation

What is WRFDA ?

- WRFDA : A Data Assimilation system for WRF (ARW) model
  - Variational and Ensemble methods
  - Used for both research and operational data analysis
- It is a supported community model, i.e. a free and shared resource with distributed development and centralized support

WRF Modeling System Flow Chart



WRFDA formulation

$$J = \frac{1}{2}(\mathbf{x} - \mathbf{x}_b)^T \mathbf{B}^{-1}(\mathbf{x} - \mathbf{x}_b) + \frac{1}{2}[\mathbf{y}_o - H(\mathbf{x})]^T \mathbf{R}^{-1}[\mathbf{y}_o - H(\mathbf{x})]$$

Define the first guess of the n^{th} outer loop: $\mathbf{x}_{n-1}$, note: $\mathbf{x}_0 = \mathbf{x}_b$

$$J = \frac{1}{2}(\mathbf{x}_n - \mathbf{x}_{n-1} + \mathbf{x}_{n-1} - \mathbf{x}_b)^T \mathbf{B}^{-1}(\mathbf{x}_n - \mathbf{x}_{n-1} + \mathbf{x}_{n-1} - \mathbf{x}_b)$$
$$+ \frac{1}{2}[\mathbf{y}_o - H(\mathbf{x}_n)]^T \mathbf{R}^{-1}[\mathbf{y}_o - H(\mathbf{x}_n)]$$

Define analysis increment of n^{th} outer loop: $\delta\mathbf{x}_n = \mathbf{x}_n - \mathbf{x}_{n-1}$

$$J = \frac{1}{2}(\delta\mathbf{x}_n + \mathbf{x}_{n-1} - \mathbf{x}_b)^T \mathbf{B}^{-1}(\delta\mathbf{x}_n + \mathbf{x}_{n-1} - \mathbf{x}_b)$$
$$+ \frac{1}{2}[\mathbf{y}_o - H(\mathbf{x}_{n-1} + \delta\mathbf{x}_n)]^T \mathbf{R}^{-1}[\mathbf{y}_o - H(\mathbf{x}_{n-1} + \delta\mathbf{x}_n)]$$

Define innovation : $\mathbf{d}_n = \mathbf{y}_o - H_{n-1}(\mathbf{x}_{n-1})$

$$J = \frac{1}{2}(\delta\mathbf{x}_n + \mathbf{x}_{n-1} - \mathbf{x}_b)^T\mathbf{B}^{-1}(\delta\mathbf{x}_n + \mathbf{x}_{n-1} - \mathbf{x}_b)$$
$$+ \frac{1}{2}[\mathbf{d}_n - \mathbf{H}_{n-1}(\delta\mathbf{x}_n)]^T\mathbf{R}^{-1}[\mathbf{d}_n - \mathbf{H}_{n-1}(\delta\mathbf{x}_n)]$$

$\mathbf{H}_{n-1} = \frac{\partial H_{n-1}}{\partial \mathbf{x}_{n-1}}$ is the tangent linear observational operator
To find the $\delta\mathbf{x}_n$ which lead J to minimium:

$$\nabla_{\delta\mathbf{x}_n} J = 0$$

$$\nabla_{\delta\mathbf{x}_n} J = \mathbf{B}^{-1}(\delta\mathbf{x}_n + \mathbf{x}_{n-1} - \mathbf{x}_b) - \mathbf{H}_{n-1}^T\mathbf{R}^{-1}(\mathbf{d}_n - \mathbf{H}_{n-1}(\delta\mathbf{x}_n))$$

At the minimium, $\nabla_{\delta \mathbf{x}_n} J$ vanishes:

$$(\mathbf{B}^{-1} + \mathbf{H}_{n-1}^T \mathbf{R}^{-1} \mathbf{H}_{n-1}) \delta \mathbf{x}_n = \mathbf{H}_{n-1}^T \mathbf{R}^{-1} \mathbf{d}_n - \mathbf{B}^{-1} (\mathbf{x}_{n-1} - \mathbf{x}_b)$$

Then

$$\delta \mathbf{x}_n = (\mathbf{B}^{-1} + \mathbf{H}_{n-1}^T \mathbf{R}^{-1} \mathbf{H}_{n-1})^{-1} [\mathbf{H}_{n-1}^T \mathbf{R}^{-1} \mathbf{d}_n - \mathbf{B}^{-1} (\mathbf{x}_{n-1} - \mathbf{x}_b)]$$

With the Woodbury Matrix Identity formulation:

$$(\mathbf{A} + \mathbf{UCV})^{-1} = \mathbf{A}^{-1} - \mathbf{A}^{-1} \mathbf{U} (\mathbf{C}^{-1} + \mathbf{VA}^{-1} \mathbf{U})^{-1} \mathbf{VA}^{-1}$$

$$(\mathbf{B}^{-1} + \mathbf{H}_{n-1}^T \mathbf{R}^{-1} \mathbf{H}_{n-1})^{-1} = \mathbf{B} - \mathbf{B} \mathbf{H}_{n-1}^T (\mathbf{R} + \mathbf{H}_{n-1} \mathbf{B} \mathbf{H}_{n-1}^T)^{-1} \mathbf{H}_{n-1} \mathbf{B}$$

$$\delta \mathbf{x}_n = [\mathbf{B} - \mathbf{B}\mathbf{H}_{n-1}^T(\mathbf{R} + \mathbf{H}_{n-1}\mathbf{B}\mathbf{H}_{n-1}^T)^{-1}\mathbf{H}_{n-1}\mathbf{B}] [\mathbf{H}_{n-1}^T\mathbf{R}^{-1}\mathbf{d}_n - \mathbf{B}^{-1}(\mathbf{x}_{n-1} - \mathbf{x}_b)]$$

$$\delta \mathbf{x}_n = \mathbf{B}\mathbf{H}_{n-1}^T(\mathbf{R} + \mathbf{H}_{n-1}\mathbf{B}\mathbf{H}_{n-1}^T)^{-1}[\mathbf{d}_n + \mathbf{H}_{n-1}(\mathbf{x}_{n-1} - \mathbf{x}_b)] - (\mathbf{x}_{n-1} - \mathbf{x}_b)$$

Please note that the new analysis is :

$$\mathbf{x}_a = \delta \mathbf{x} + \mathbf{x}_{n-1}$$

For first outer loop, $\mathbf{x}_0 = \mathbf{x}_b$ and $\mathbf{d} = \mathbf{y}_o - H(\mathbf{x}_b)$

$$J = \frac{1}{2}\delta\mathbf{x}^T\mathbf{B}^{-1}\delta\mathbf{x} + \frac{1}{2}(\mathbf{d} - \mathbf{H}\delta\mathbf{x})^T\mathbf{R}^{-1}(\mathbf{d} - \mathbf{H}\delta\mathbf{x})$$

$$\nabla_{\delta\mathbf{x}}J = \mathbf{B}^{-1}\delta\mathbf{x} - \mathbf{H}^T\mathbf{R}^{-1}(\mathbf{d} - \mathbf{H}\delta\mathbf{x})$$

$$\delta\mathbf{x} = \mathbf{B}\mathbf{H}^T(\mathbf{R} + \mathbf{H}\mathbf{B}\mathbf{H}^T)^{-1}\mathbf{d}$$

B in WRFDA

- Partly for preconditioning reasons: $\mathbf{B} = \mathbf{U}\mathbf{U}^T$ with $\mathbf{U} = \mathbf{U}_p$
- Horizontal transform ($\mathbf{U}_h$) is via
  - Regional - recursive filters
  - Glocal - power spectrum
- Vertical transform ($\mathbf{U}_v$) is via
  - cv3 - recursive filters
  - cv5&6 - via EOF
- Physical transformation ($\mathbf{U}_p$) depends upon the choice of the analysis control variable

Cost function in WRFDA

- Define: $\delta\mathbf{x} = \mathbf{U}\mathbf{v}$
- $\mathbf{v}$ is the analysis increment in control variable space
  - streamfunction
  - unbalanced velocity potential
  - unbalanced temperature
  - RH
  - unbalanced surface pressure
- The actual cost function in WRFDA :

$$J = \frac{1}{2}\mathbf{v}^T\mathbf{v} + \frac{1}{2}(\mathbf{d} - \mathbf{H}\mathbf{U}\mathbf{v})^T\mathbf{R}^{-1}(\mathbf{d} - \mathbf{H}\mathbf{U}\mathbf{v})$$

$$\mathbf{x}_a = \mathbf{U}\mathbf{v} + \mathbf{x}_{n-1}$$

Prepare the BE

$$J = \frac{1}{2}\mathbf{v}^T\mathbf{v} + \frac{1}{2}(\mathbf{d} - \mathbf{HUv})^T\mathbf{R}^{-1}(\mathbf{d} - \mathbf{HUv})$$

- Nominally, $\mathbf{B} = \mathbf{U}\mathbf{U}^T$ is the background error covariance
- For initial testing, default background error statistics may be used
  - be.dat file (CV option 5) from test case tar file can only be used with the case of online tutorial
  - be.dat.cv3 (CV option 3) from source code tar file can be used for general test domains, tuning is still needed for satisfied performance
- Ultimately, $\mathbf{B}$ should be specific to the particular model domain (and season)

Prepare the Background

$$J = \frac{1}{2}\mathbf{v}^T\mathbf{v} + \frac{1}{2}(\mathbf{d} - \mathbf{HUv})^T\mathbf{R}^{-1}(\mathbf{d} - \mathbf{HUv})$$

$$\mathbf{Uv} = \mathbf{x} - \mathbf{x}_b$$

- In cold-start mode: accomplished by running the WPS and real programs
  - The background is the wrfinput_d01 file
- In cycling mode: the output of the WRF model
  - WRF can output wrfinput-formatted files used for cycling

Prepare the Observations & Errors

$$J = \frac{1}{2}\mathbf{v}^T\mathbf{v} + \frac{1}{2}(\mathbf{d} - \mathbf{HUv})^T\mathbf{R}^{-1}(\mathbf{d} - \mathbf{HUv})$$

$$\mathbf{d} = \mathbf{y} - H(\mathbf{x}_b)$$

- Conventional observation input for WRFDA is supplied through
  - Little_R format, observation preprocessor, OBSPROC
  - Prepbufr format data directly
- Observation error covariance also provided by OBSPROC (R is a diagonal matrix)
- Separate input file (ASCII) for radar, both reflectivity and radial velocity
- Separate input file for satellite radiances, BUFR format

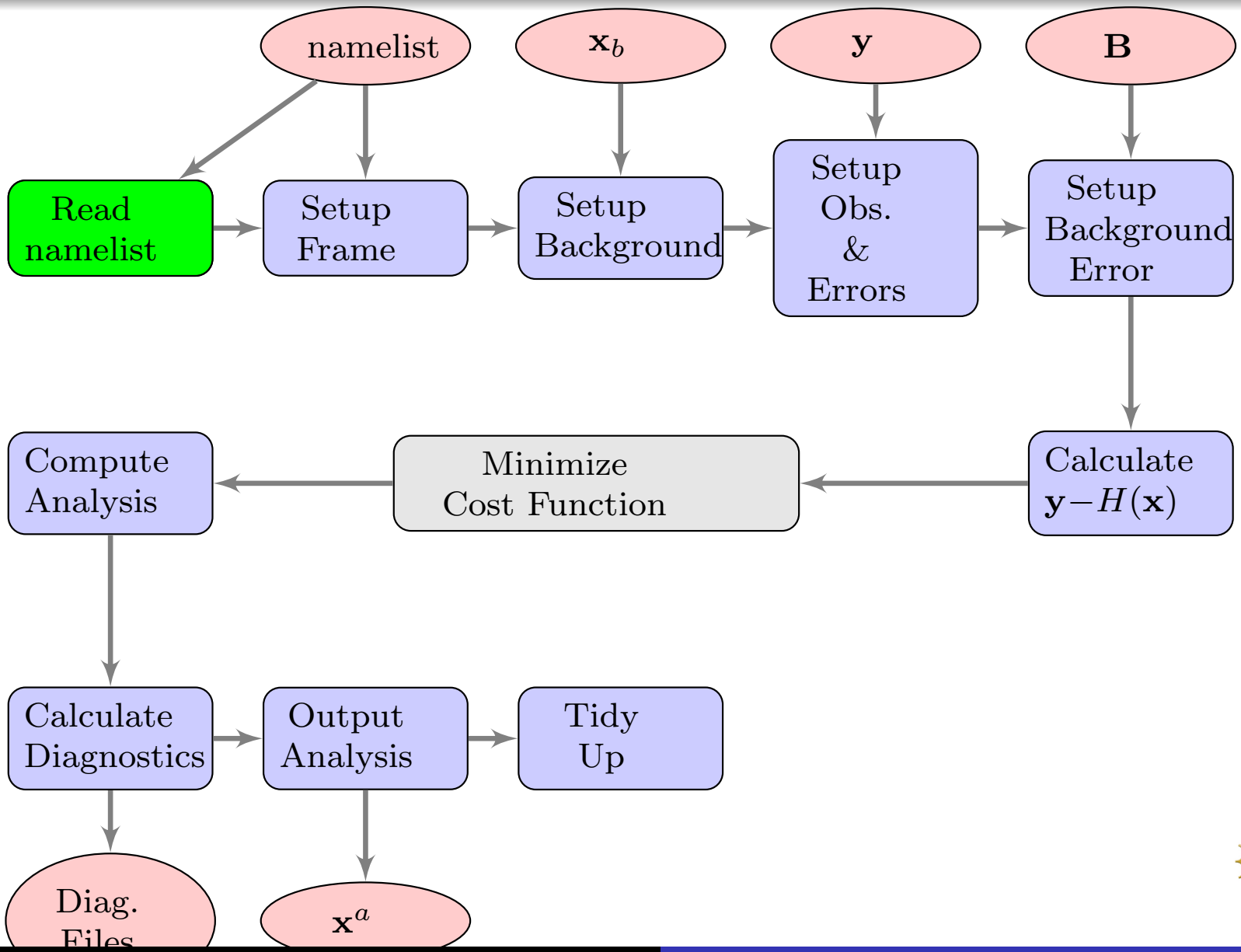
Run WRFDA

$$J = \frac{1}{2}\mathbf{v}^T\mathbf{v} + \frac{1}{2}(\mathbf{d} - \mathbf{HUv})^T\mathbf{R}^{-1}(\mathbf{d} - \mathbf{HUv})$$

- H is the observational operator, which calculate the counterpart of observations in model space
- Conjuagate gradient method
- Try to find a $\mathbf{v}$, which minimizes J and then $\mathbf{x}_a = \mathbf{Uv} + \mathbf{x}_{n-1}$

Update Boundary Condition

- After creating an analysis, $\mathbf{x}_a$, we have changed the initial conditions for the model
- However, tendencies in wrfbdy_d01 file are valid for background, $\mathbf{x}_b$
- The update_bc program adjusts these tendencies based on the difference $\mathbf{x}_a - \mathbf{x}_b$
- Of course, if $\mathbf{x}_a$ was produced for reasons other than running WRF, there is probably not a need to update boundary conditions

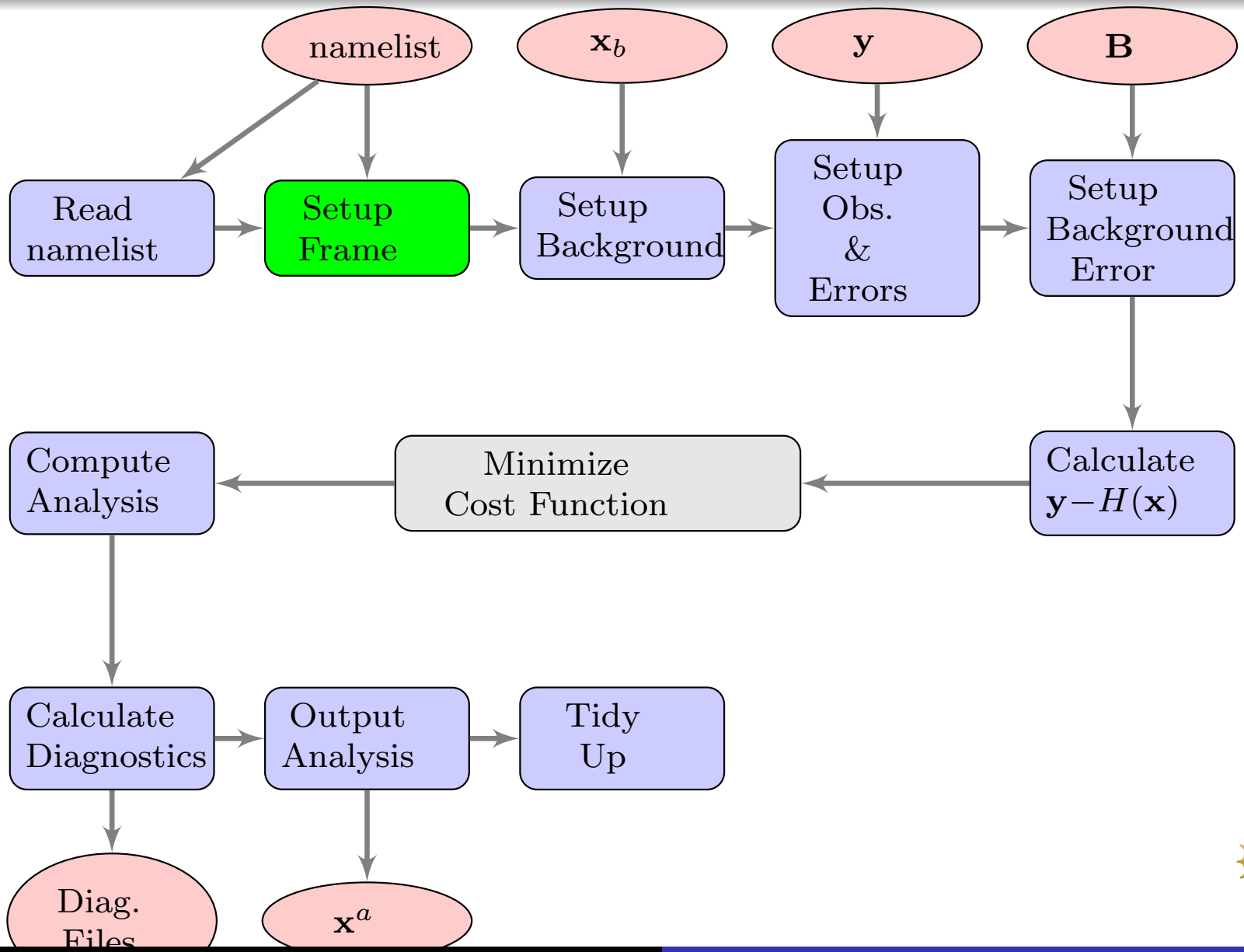

namelist
$\mathbf{x}_b$
$\mathbf{y}$
$\mathbf{B}$
Read namelist
Setup Frame
Setup Background
Setup Obs. & Errors
Setup Background Error
Compute Analysis
Minimize Cost Function
Calculate $\mathbf{y}-H(\mathbf{x})$
Calculate Diagnostics
Output Analysis
Tidy Up
Diag. Files
$\mathbf{x}^a$

Read namelist

- Reads WRFDA data assimilation options from *namelist.input* file
- Performs consistency checks between namelist options.

```
Calling order:
da_wrfvar_main ==> call da_wrfvar_init1, da_wrfvar_init2 ==> call initial_config


Calling subroutines:
da_wrfvar_main.f90 ==> da_wrfvar_init1.inc, da_wrfvar_init2.inc ==> module_configure.F
```

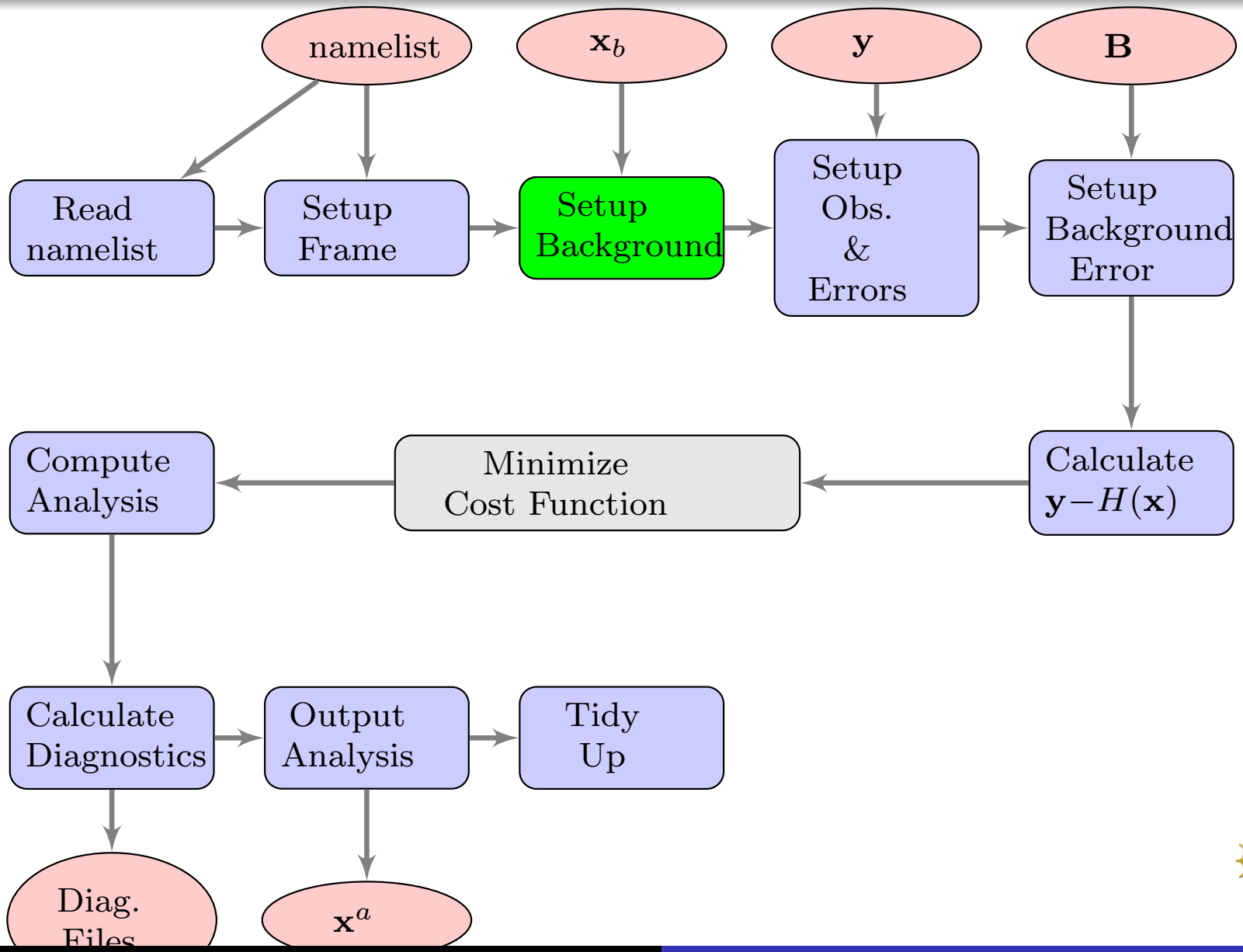

namelist
$\mathbf{x}_b$
$\mathbf{y}$
$\mathbf{B}$
Read namelist
Setup Frame
Setup Background
Setup Obs. & Errors
Setup Background Error
Compute Analysis
Minimize Cost Function
Calculate $\mathbf{y}-H(\mathbf{x})$
Calculate Diagnostics
Output Analysis
Tidy Up
Diag. Files
$\mathbf{x}^a$

Setup Frame

- Use WRF framework distributed memory capability to initialize tile, memory, patch dimensions, etc.

```
Calling order:
da_wrfvar_main ==> call da_wrfvar_init2 ==> call alloc_and_configure_domain
da_wrfvar_main ==> call da_wrfvar_run.inc ==> call da_wrfvar_interface ==> call da_solve
 ==> call da_solve_init


Calling subroutines:
da_wrfvar_main.f90 ==> da_wrfvar_init2.inc ==> module_domain.F
da_wrfvar_main.f90 ==> da_wrfvar_run.inc ==> da_wrfvar_interface.inc ==> da_solve.inc
==> da_solve_init.inc
```

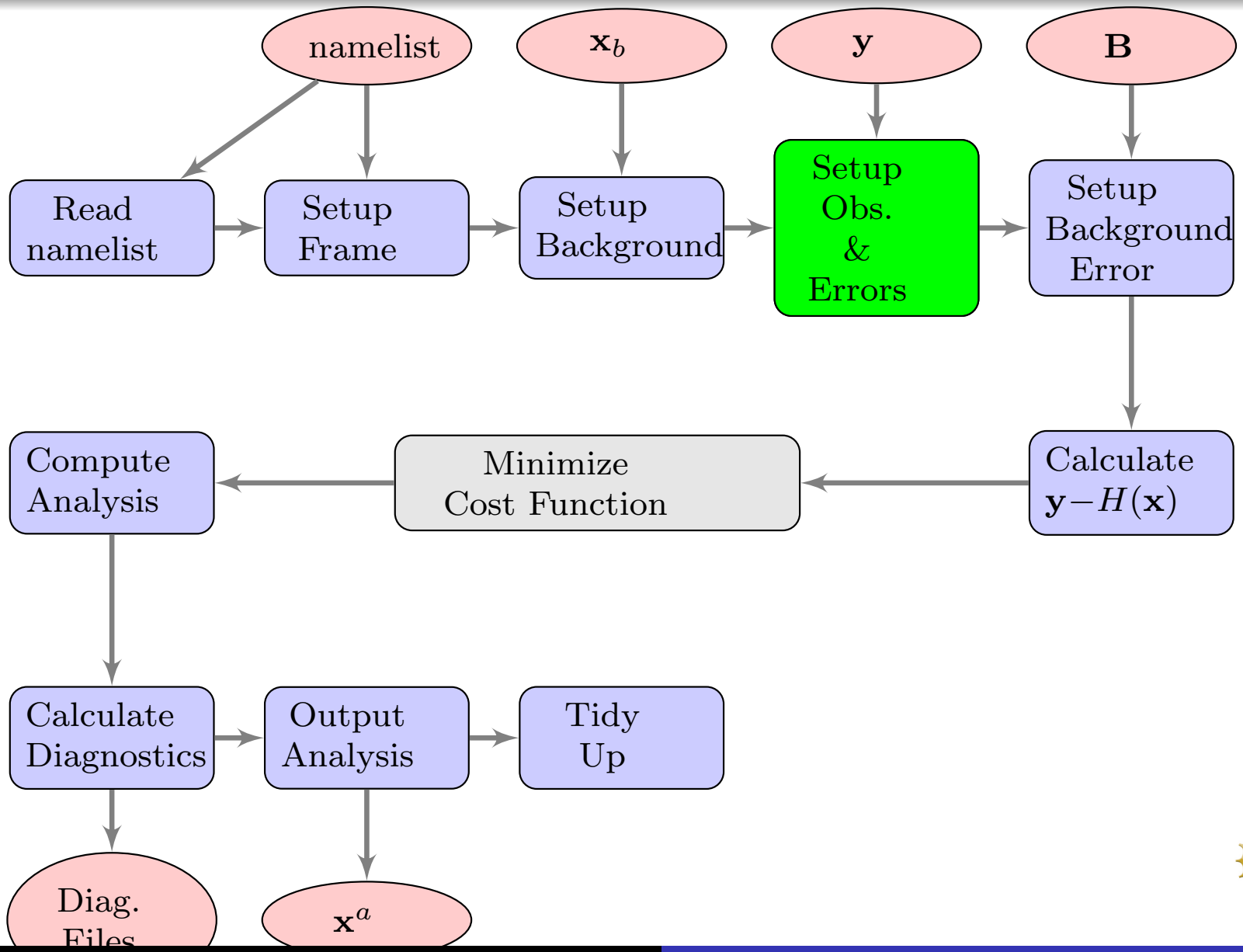

namelist
$\mathbf{x}_b$
$\mathbf{y}$
$\mathbf{B}$
Read namelist
Setup Frame
Setup Background
Setup Obs. & Errors
Setup Background Error
Compute Analysis
Minimize Cost Function
Calculate $\mathbf{y}-H(\mathbf{x})$
Calculate Diagnostics
Output Analysis
Tidy Up
Diag. Files
$\mathbf{x}^a$

Setup Background

- Reads the first-guess file
- Extracts fields used by WRFDA
- Creates background FORTRAN 90 derived data type xb etc.

```
Calling order:
da_wrfvar_main ==> call da_wrfvar_init2 ==> call da_med_initialdata_input
da_wrfvar_main ==> call da_wrfvar_run ==> call da_wrfvar_interface ==> call da_solve
==>call da_setup_firstguess


Calling subroutines:
da_wrfvar_main.f90 ==> da_wrfvar_init2.inc ==> da_med_initialdata_input.inc
da_wrfvar_main.f90 ==> da_wrfvar_run.inc ==> da_wrfvar_interface.inc ==> da_solve.inc
==>da_setup_firstguess.inc
```

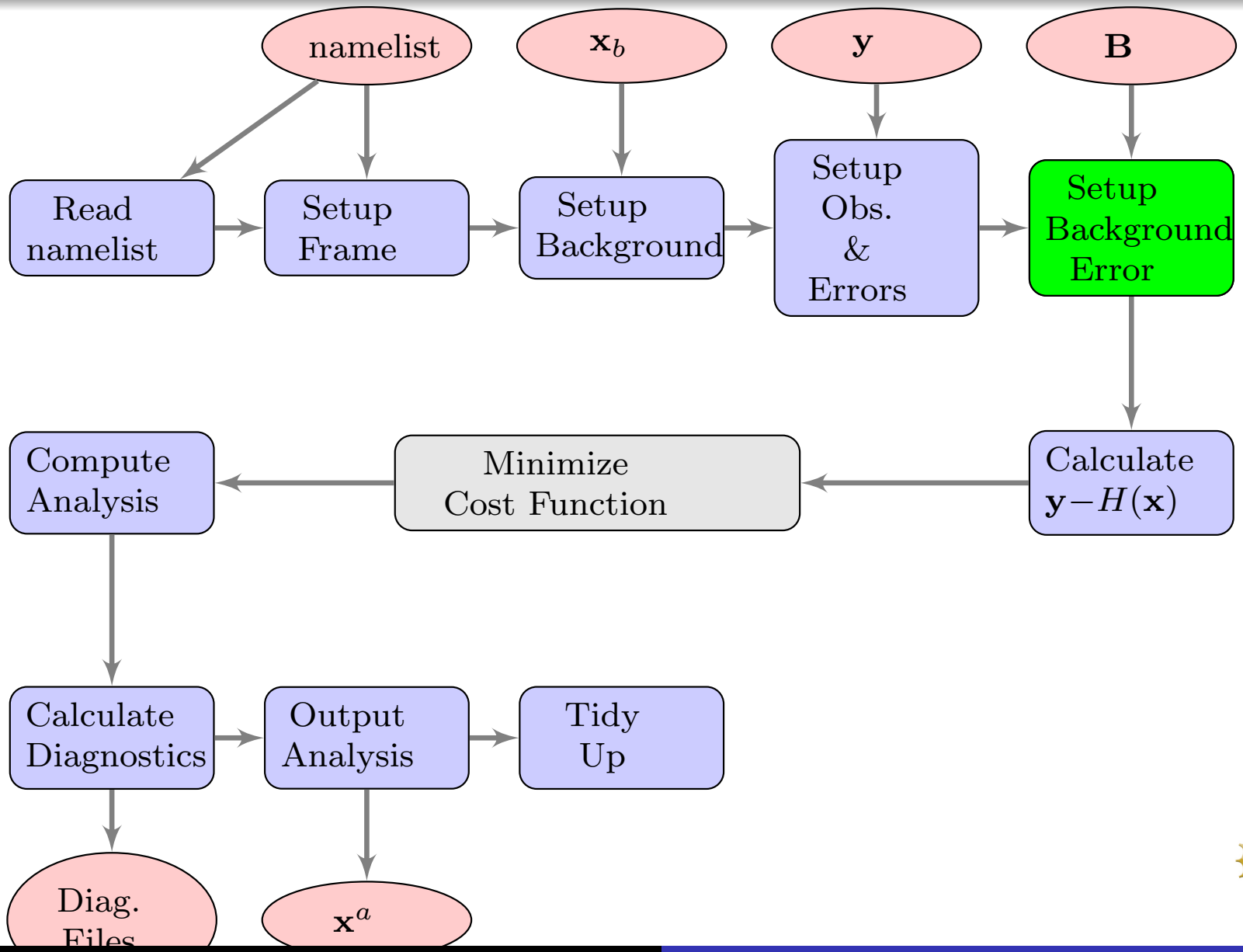

namelist
$\mathbf{x}_b$
$\mathbf{y}$
$\mathbf{B}$
Read namelist
Setup Frame
Setup Background
Setup Obs. & Errors
Setup Background Error
Compute Analysis
Minimize Cost Function
Calculate $\mathbf{y}-H(\mathbf{x})$
Calculate Diagnostics
Output Analysis
Tidy Up
Diag. Files
$\mathbf{x}^a$

Setup Observations & Errors

- Reads in observations
- Assign observational error
- Creates observation FORTRAN 90 derived data type *ob*
- Domain and time check

```
Calling order:
da_wrfvar_main ==> call da_wrfvar_run ==> call da_wrfvar_interface ==> call da_solve
==>call da_setup_obs_structures
```

```
Calling subroutines:
da_wrfvar_main.f90 ==> da_wrfvar_run.inc ==> da_wrfvar_interface.inc ==> da_solve.inc
==>da_setup_obs_structures.inc
```

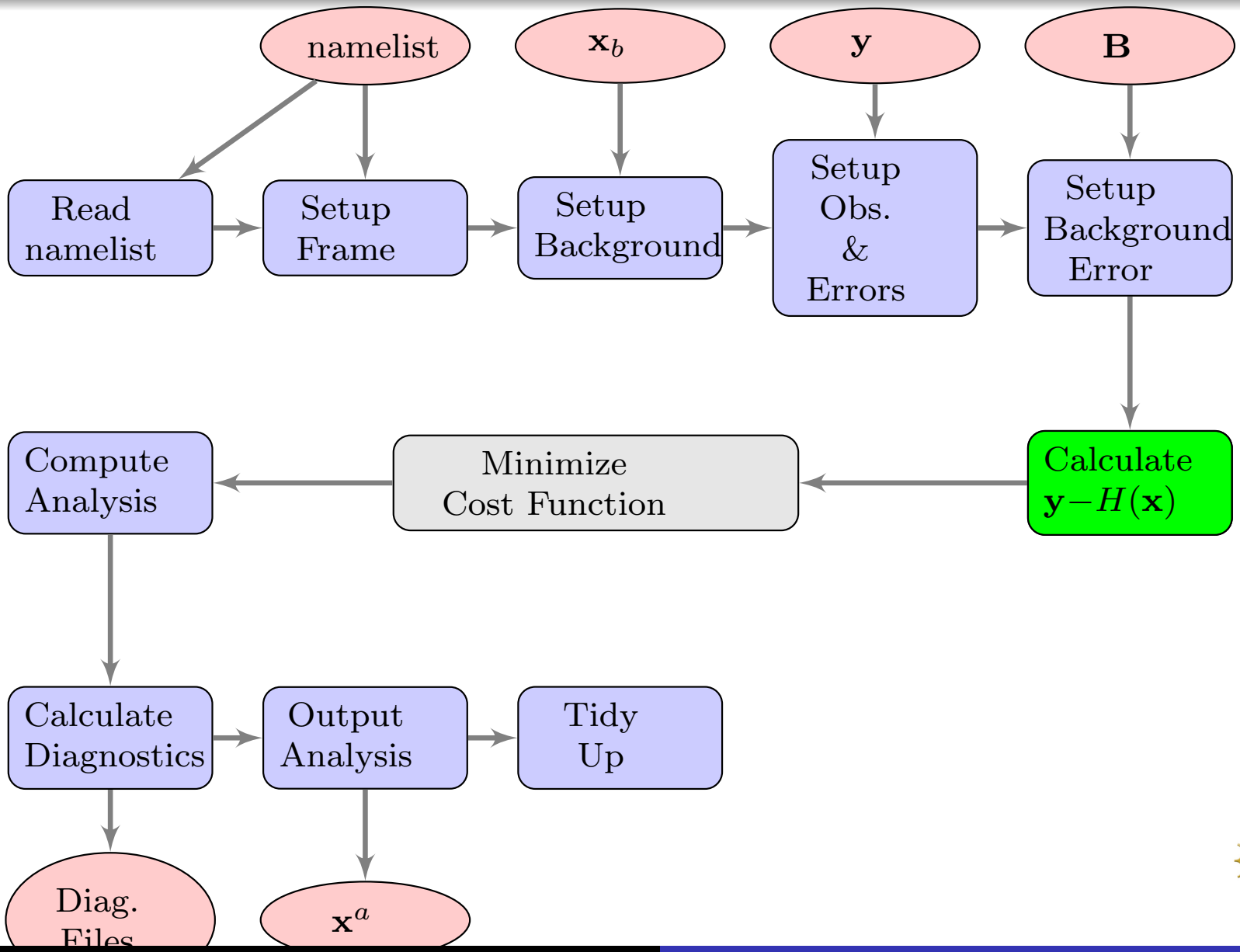

namelist
$\mathbf{x}_b$
$\mathbf{y}$
$\mathbf{B}$
Read namelist
Setup Frame
Setup Background
Setup Obs. & Errors
Setup Background Error
Compute Analysis
Minimize Cost Function
Calculate $\mathbf{y}-H(\mathbf{x})$
Calculate Diagnostics
Output Analysis
Tidy Up
Diag. Files
$\mathbf{x}^a$

Setup Background Error

- Reads in background error statistics
- Extracts necessary quantities: eigenvectors, eigenvalues, lengthscales, regression coefficients, etc.
- Creates background error FORTRAN 90 derived data type *be*
- Reference :**Online BE Documents**

```
Calling order:
da_wrfvar_main ==> call da_wrfvar_run ==> call da_wrfvar_interface ==> call da_solve
==>call da_setup_background_errors
```

```
Calling subroutines:
da_wrfvar_main.f90 ==> da_wrfvar_run.inc ==> da_wrfvar_interface.inc ==> da_solve.inc
==>da_setup_background_errors.inc
```

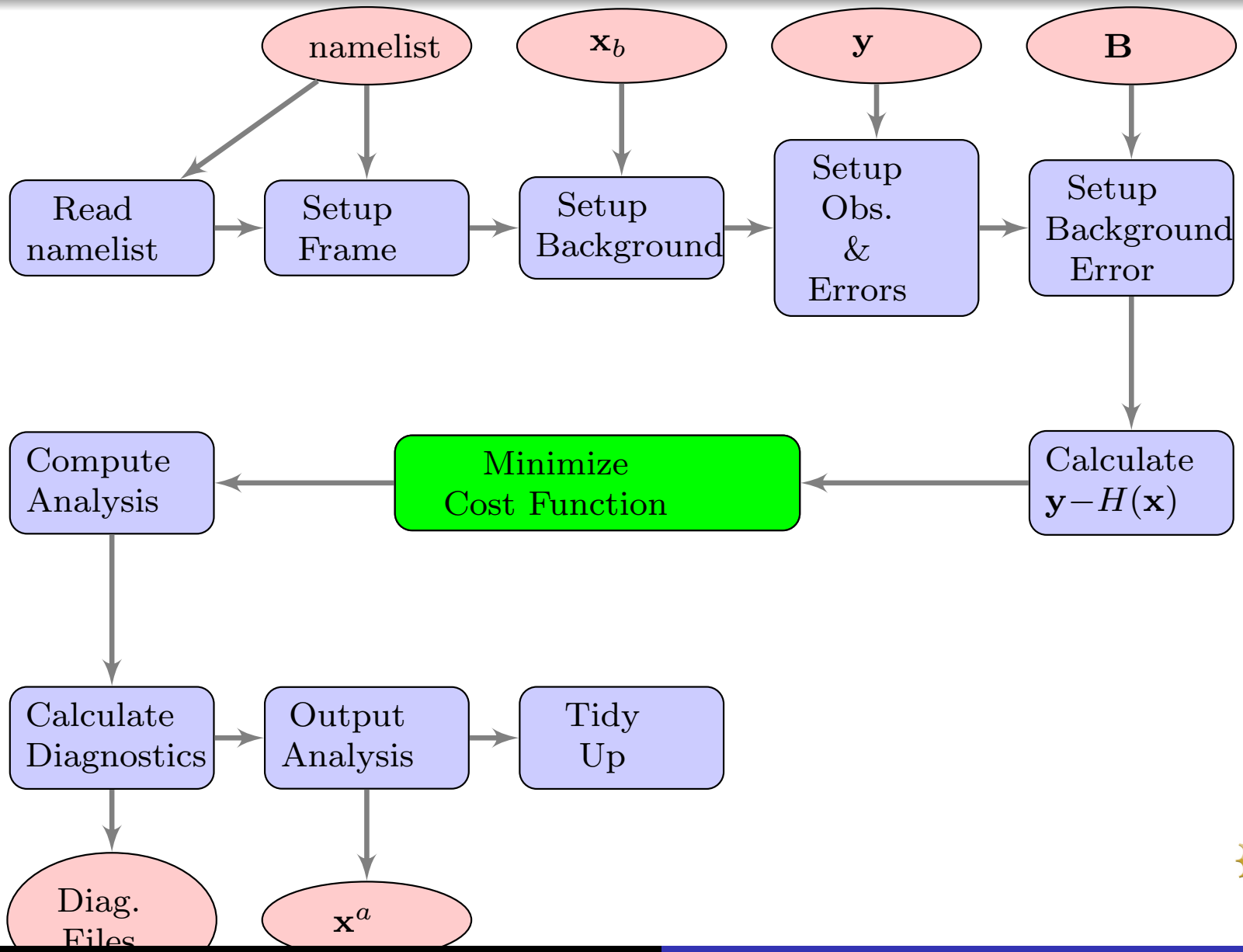

namelist
$\mathbf{x}_b$
$\mathbf{y}$
$\mathbf{B}$
Read namelist
Setup Frame
Setup Background
Setup Obs. & Errors
Setup Background Error
Compute Analysis
Minimize Cost Function
Calculate $\mathbf{y}-H(\mathbf{x})$
Calculate Diagnostics
Output Analysis
Tidy Up
Diag. Files
$\mathbf{x}^a$

Calculate Innovation

- Calculates model equivalent of observations through interpolation and change of variable
- Computes observation minus first guess ($\mathbf{y} - H(\mathbf{x})$) value
- Creates innovation vector FORTRAN 90 derived data type *iv*

```
Calling order:
da_wrfvar_main ==> call da_wrfvar_run ==> call da_wrfvar_interface ==> call da_solve
==>call da_get_innov_vector, da_allocate_y


Calling subroutines:
da_wrfvar_main.f90 ==> da_wrfvar_run.inc ==> da_wrfvar_interface.inc ==> da_solve.inc
==>da_get_innov_vector.inc, da_allocate_y.inc
```


namelist
$\mathbf{x}_b$
$\mathbf{y}$
$\mathbf{B}$
Read namelist
Setup Frame
Setup Background
Setup Obs. & Errors
Setup Background Error
Compute Analysis
Minimize Cost Function
Calculate $\mathbf{y}-H(\mathbf{x})$
Calculate Diagnostics
Output Analysis
Tidy Up
Diag. Files
$\mathbf{x}^a$

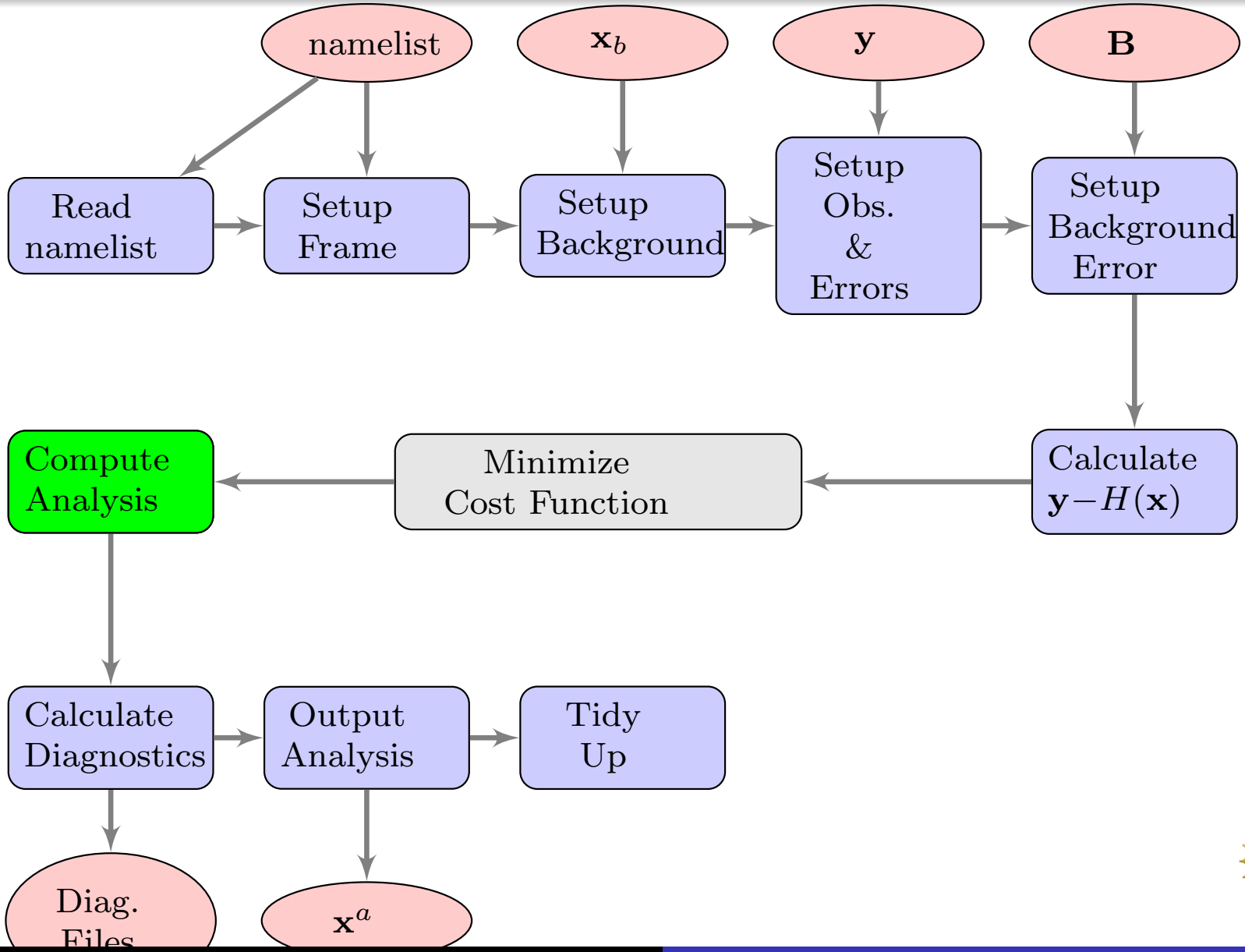
Minimization

Use conjugate gradient method

- Initializes analysis increments to zero
- Computes cost function (if desired)
- Computes gradient of cost function
- Uses cost function and gradient to calculate new value of analysis control variable

```
Calling order:
da_wrfvar_main ==> call da_wrfvar_run ==> call da_wrfvar_interface ==> call da_solve
==>call da_minimise_cg
```

```
Calling subroutines:
da_wrfvar_main.f90 ==> da_wrfvar_run.inc ==> da_wrfvar_interface.inc ==> da_solve.inc
==>da_minimise_cg.inc
```

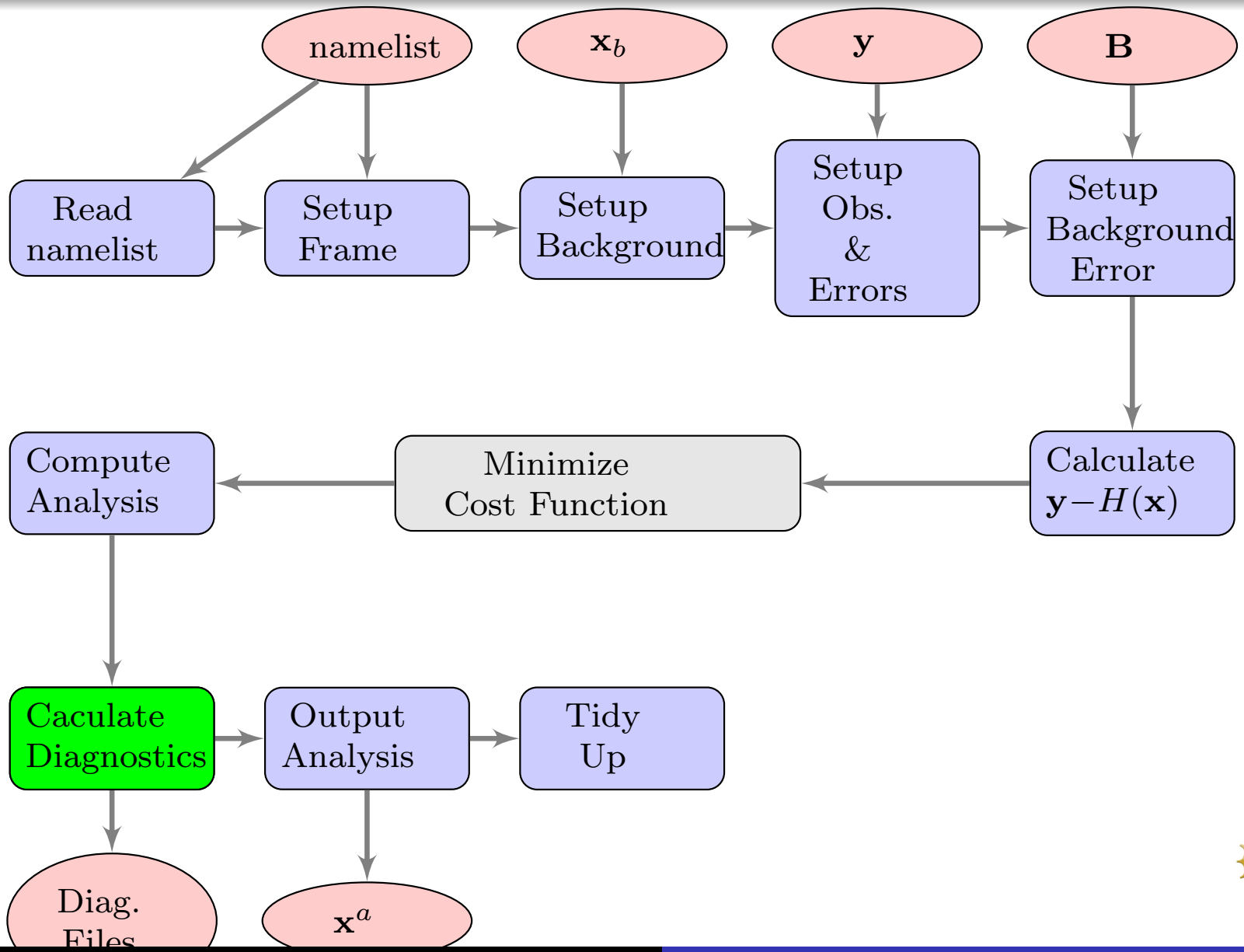

namelist
$\mathbf{x}_b$
$\mathbf{y}$
$\mathbf{B}$
Read namelist
Setup Frame
Setup Background
Setup Obs. & Errors
Setup Background Error
Compute Analysis
Minimize Cost Function
Calculate $\mathbf{y}-H(\mathbf{x})$
Calculate Diagnostics
Output Analysis
Tidy Up
Diag. Files
$\mathbf{x}^a$

Compute Analysis

- Once WRFDA has found a converged control variable, convert control variable to model space analysis increments
- Calculate:
  analysis = first-guess + analysis increment
- Performs consistency checks, e.g., remove negative humidity etc.
- Optionally, do outer loop

```
Calling order:
da_wrfvar_main ==> call da_wrfvar_run ==> call da_wrfvar_interface ==> call da_solve
==>call da_transfer_xatoanalysis
```

```
Calling subroutines:
da_wrfvar_main.f90 ==> da_wrfvar_run.inc ==> da_wrfvar_interface.inc ==> da_solve.inc
==>da_transfer_xatoanalysis.inc
```

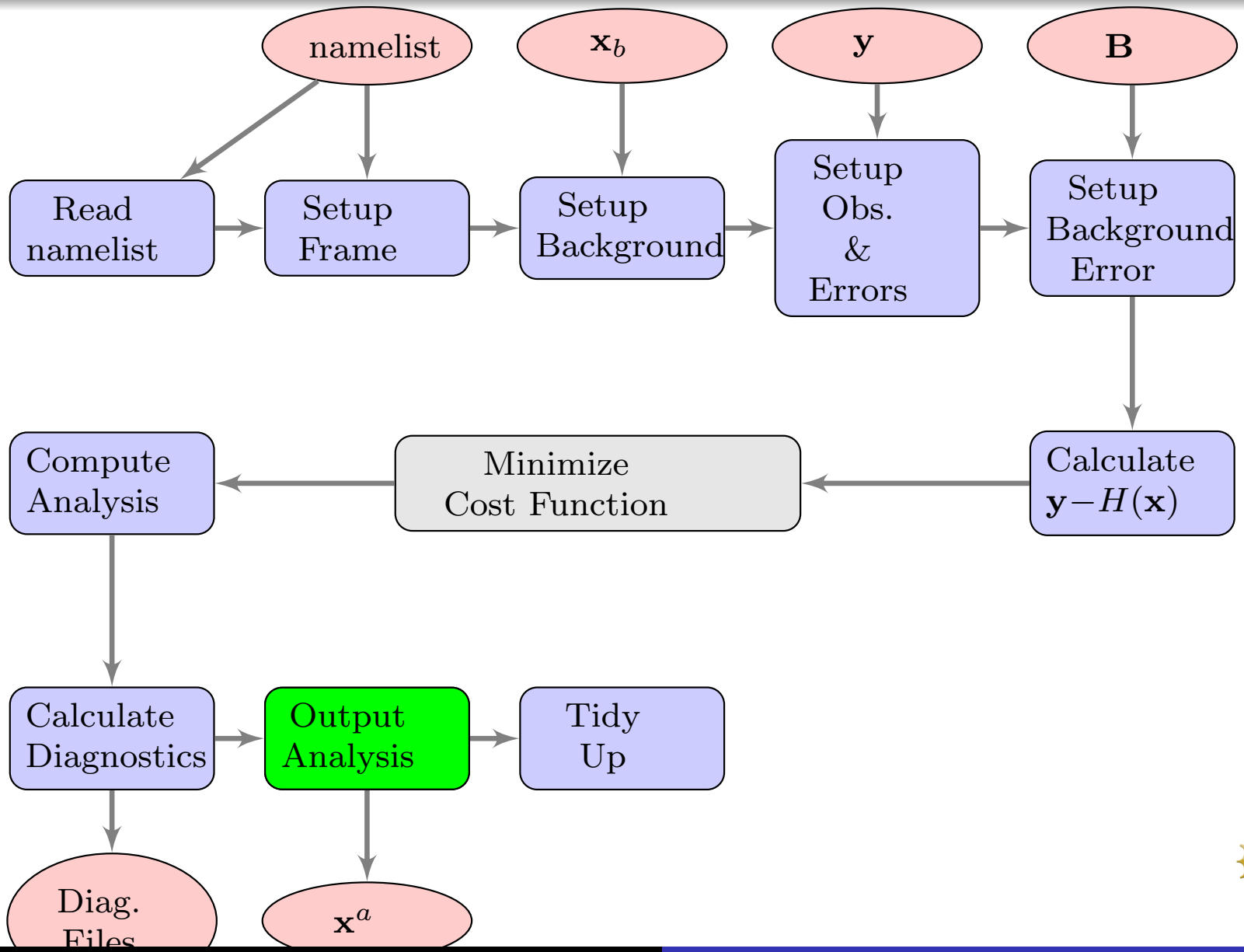

namelist
$\mathbf{x}_b$
$\mathbf{y}$
$\mathbf{B}$
Read namelist
Setup Frame
Setup Background
Setup Obs. & Errors
Setup Background Error
Compute Analysis
Minimize Cost Function
Calculate $\mathbf{y}-H(\mathbf{x})$
Caculate Diagnostics
Output Analysis
Tidy Up
Diag. Files
$\mathbf{x}^a$

Calculate Diagnostics

- Output $\mathbf{y} - H(\mathbf{x}_b)$, $\mathbf{y} - H(\mathbf{x}^a)$ statistics for all observation types and variables
- Compute $\mathbf{x}^a - \mathbf{x}_b$ (analysis increment) statistics for all model variables and levels
- Statistics include minimum, maximum (and their locations), mean and standard deviation.

```
Calling order:
da_wrfvar_main ==> call da_wrfvar_run ==> call da_wrfvar_interface ==> call da_solve
==> call da_write_diagnostics
```

```
Calling subroutines:
da_wrfvar_main.f90 ==> da_wrfvar_run.inc ==> da_wrfvar_interface.inc ==> da_solve.inc
==> da_write_diagnostics.inc
```

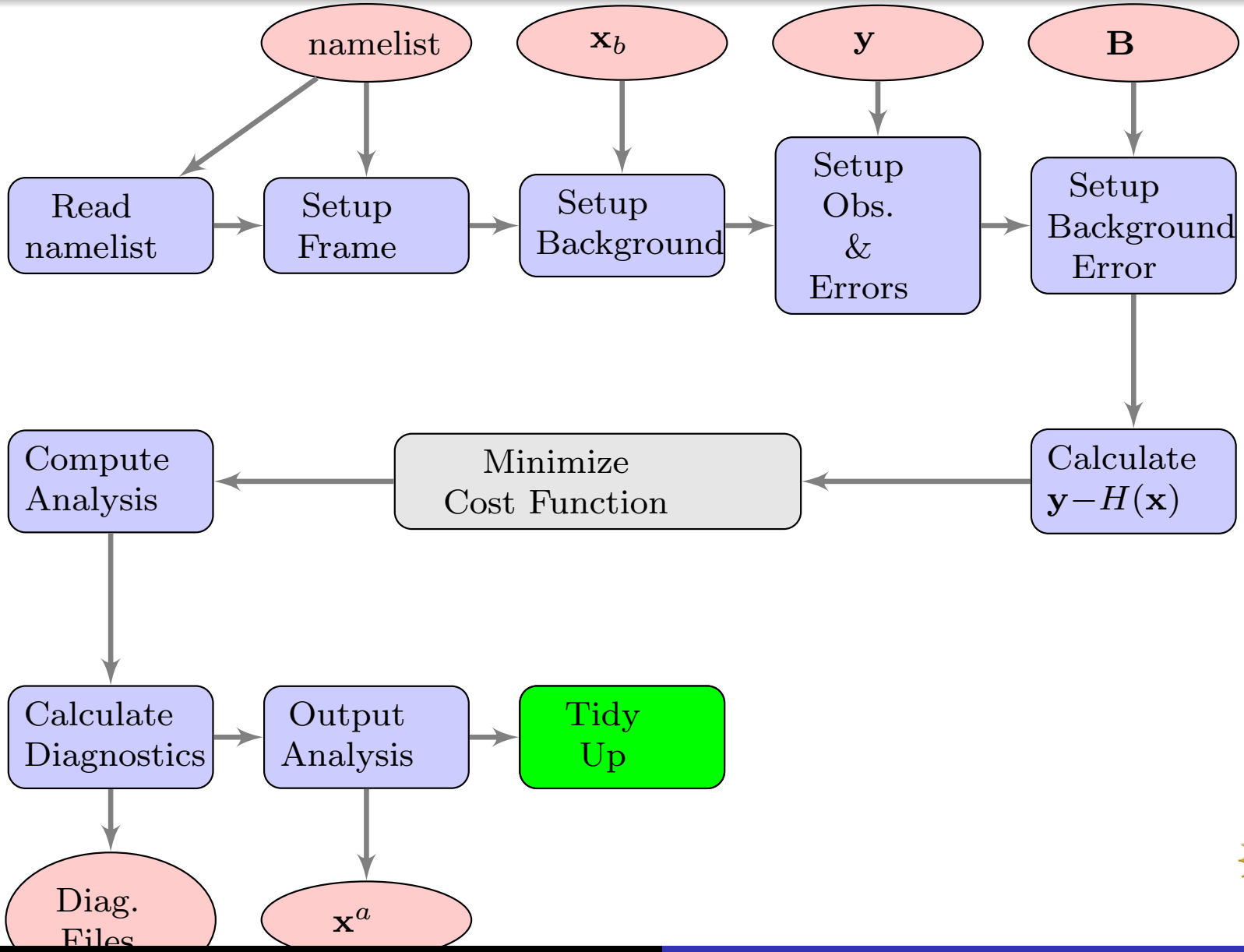

namelist
$\mathbf{x}_b$
$\mathbf{y}$
$\mathbf{B}$
Read namelist
Setup Frame
Setup Background
Setup Obs. & Errors
Setup Background Error
Compute Analysis
Minimize Cost Function
Calculate $\mathbf{y}-H(\mathbf{x})$
Calculate Diagnostics
Output Analysis
Tidy Up
Diag. Files
$\mathbf{x}^a$

Output Analysis

- Outputs analysis in native model format.

```
Calling order:
da_wrfvar_main ==> call da_wrfvar_run ==> call da_wrfvar_interface ==> call da_solve
==> call da_med_initialdata_output


Calling subroutines:
da_wrfvar_main.f90 ==> da_wrfvar_run.inc ==> da_wrfvar_interface.inc ==> da_solve.inc
==> da_med_initialdata_output.inc
```

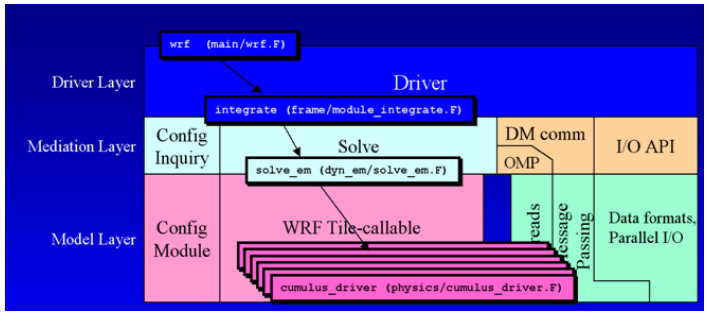

namelist
$\mathbf{x}_b$
$\mathbf{y}$
$\mathbf{B}$
Read namelist
Setup Frame
Setup Background
Setup Obs. & Errors
Setup Background Error
Compute Analysis
Minimize Cost Function
Calculate $\mathbf{y}-H(\mathbf{x})$
Calculate Diagnostics
Output Analysis
Tidy Up
Diag. Files
$\mathbf{x}^a$

Tidy Up

- Deallocate dynamically-allocated arrays, structures, etc.
- Timing information
- Clean end to WRFDA

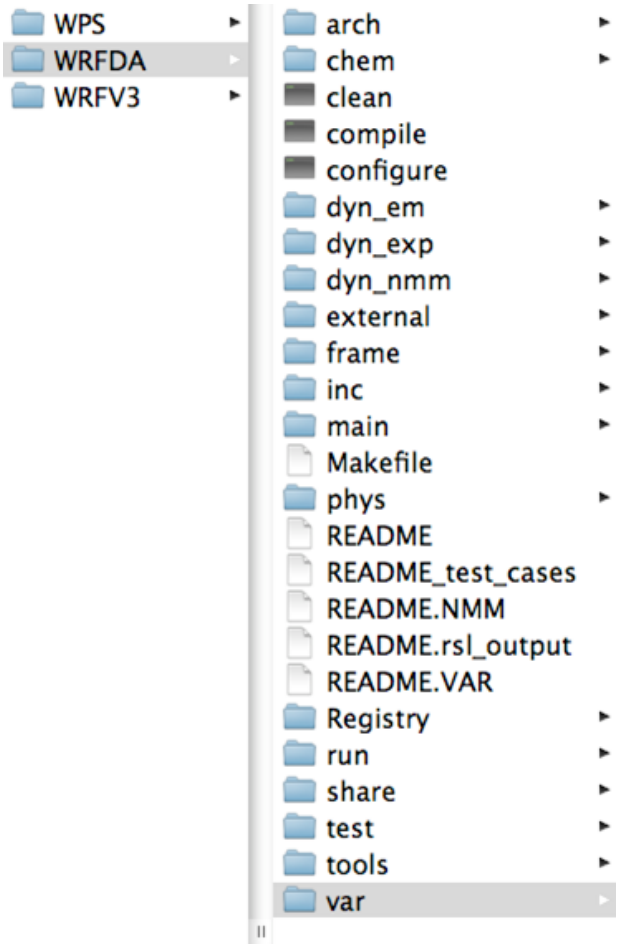
```
Calling subroutines:
da_wrfvar_main.f90 ==> da_wrfvar_run.inc ==> da_wrfvar_interface.inc ==> da_solve.inc
==> deallocate ....
```

WRFDA Software Framework



- WRFDA relies on the WRF Software framework for
  - Distributed memory parallelism (halo exchanges, etc.)
  - Input/Output of first guess and analysis files
  - Parallel transposes
- WRFDA also uses
  - The WRF Registry mechanism to handle definitions of fields, halos, type, package and transposes (Registry.wrfvar)
  - The WRF build system (clean, configure, compile)

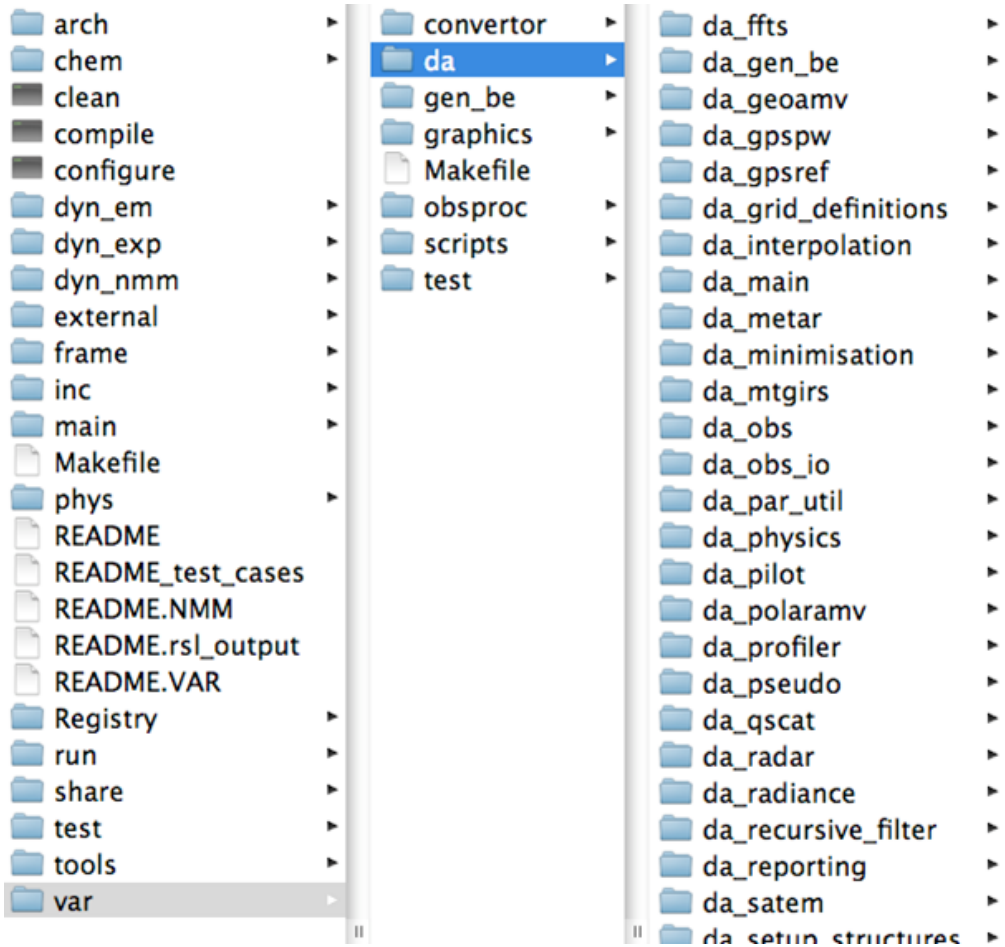
WRFDA Code Organization



Remarks

Besides the directories for WRF, the WRFDA tar file contains a *var* directory, which holds all of the WRFDA code

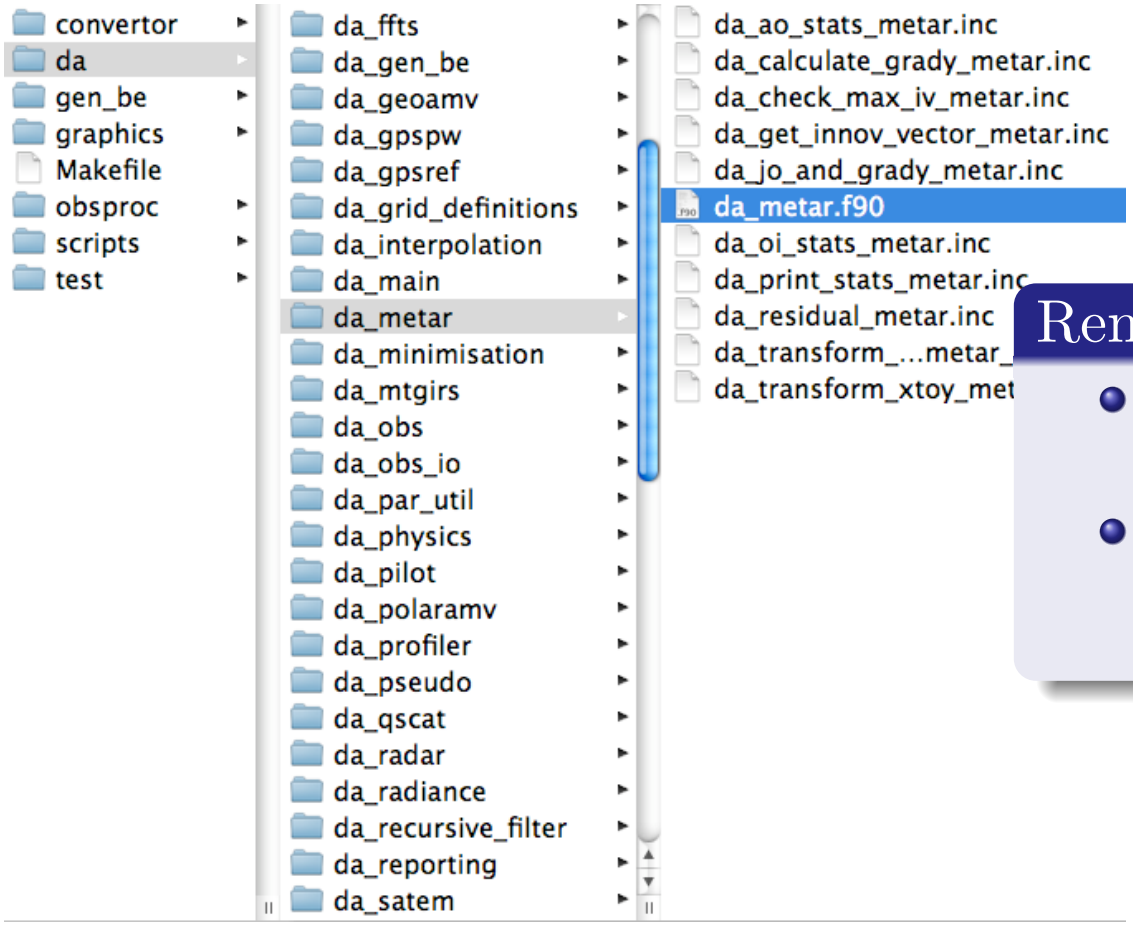
WRFDA Code Organization



Remarks

Generally, each subdirectory of *da* contains a Fortran module with the same name

WRFDA Code Organization



Remarks

- da_metar.f90 contains a Fortran module
- Each .inc file corresponds to a subroutine within the module

How to add a new observation in WRFDA

A example of adding TAMDAR data, contributed by Dr. Hongli Wang

1. define_structures
   - da/da_define_structures/da_deallocate_y.inc
   - da/da_define_structures/da_zero_y.inc
   - da/da_define_structures/da_deallocate_observations.inc
   - da/da_define_structures/da_allocate_y.inc
   - da/da_define_structures/da_allocate_observations.inc
   - da/da_define_structures/da_define_structures.f90
   - da/da_setup_structures/da_setup_obs_structures.inc
   - da/da_setup_structures/da_setup_structures.f90
2. da_obs_io
   - da/da_obs_io/da_search_obs.inc
   - da/da_obs_io/da_write_filtered_obs.inc
   - da/da_obs_io/da_read_obs_ascii.inc
   - da/da_obs_io/da_scan_obs_ascii.inc
   - da/da_obs_io/da_obs_io.f90
3. da_obs
   - da/da_obs/da_random_omb_all.inc
   - da/da_obs/da_add_noise_to_ob.inc
   - da/da_obs/da_obs.f90
   - da/da_obs/da_count_filtered_obs.inc
   - da/da_obs/da_transform_xtoy_adj.inc
   - da/da_obs/da_use_obs_errfac.inc
   - da/da_obs/da_fm_decoder.inc

How to add a new observation in WRFDA, cont'd

```
      da/da_obs/da_fill_obs_structures.inc
      da/da_obs/da_transform_xtoy.inc
4. da_tamdar
      da/da_tamdar/da_ao_stats_tamdar.inc
      da/da_tamdar/da_calculate_grady_tamdar.inc
      da/da_tamdar/da_check_max_iv_tamdar.inc
      da/da_tamdar/da_get_innov_vector_tamdar.inc
      da/da_tamdar/da_jo_and_grady_tamdar.inc
      da/da_tamdar/da_jo_tamdar_uvtq.inc
      da/da_tamdar/da_oi_stats_tamdar.inc
      da/da_tamdar/da_print_stats_tamdar.inc
      da/da_tamdar/da_residual_tamdar.inc
      da/da_tamdar/da_tamdar.f90
      da/da_tamdar/da_transform_xtoy_tamdar.inc
      da/da_tamdar/da_transform_xtoy_tamdar_adj.inc
5. da_minimisation
      da/da_minimisation/da_calculate_grady.inc
      da/da_minimisation/da_calculate_residual.inc
      da/da_minimisation/da_minimisation.f90
      da/da_minimisation/da_get_innov_vector.inc
      da/da_minimisation/da_get_var_diagnostics.inc
      da/da_minimisation/da_jo_and_grady.inc
      da/da_minimisation/da_write_diagnostics.inc
 6. control/regestry/compile
      da/da_control/da_control.f90
      Registry/Registry.wrfvar
      da/Makefile
```

How to add a new observation in WRFDA, cont'd

```
7. da_test
      da/da_test/da_check_xtoy_adjoint_tamdar.inc
      da/da_test/da_test.f90
      da/da_test/da_check_xtoy_adjoint.inc
      da/da_test/da_get_y_lhs_value.inc
8. obsproc
      obsproc/src/3dvar_obs.F90
      obsproc/src/fm_decoder.F90
      obsproc/src/sort_platform.F90
      obsproc/src/module_decoded.F90
      obsproc/src/module_write.F90
      obsproc/src/module_complete.F90
      obsproc/src/module_duplicate.F90
      obsproc/src/platform_interface.inc
      obsproc/src/module_namelist.F90
      obsproc/src/module_err_afwa.F90
      obsproc/src/module_per_type.F90
      obsproc/src/module_qc.F90
      Registry/Registry.wrfvar
tamdar_sfc
      da/da_define_structures/da_define_structures.f90
      da/da_control/da_control.f90
      da/da_define_structures/da_allocate_y.inc
      da/da_define_structures/da_deallocate_observations.inc
      da/da_define_structures/da_deallocate_y.inc
      da/da_define_structures/da_zero_y.inc
```

Quick Start

- Supported compilation mechanisms
  - Serial
  - Distributed-memory(dm)
  - Shared-memory(sm) (use with cautions, thread safe compiler only–IBM XLF)
  - hybrid (dm+sm) (use with cautions)
- Supported platforms
  - IBM: XLF
  - Linux: PGI, IFORT, GFORTRAN (higher version needed,V4.4.0 tested)
  - Macintosh intel: PGI, G95, GFORTRAN (higher version needed,V4.4.0 tested)
- Included libraries
  - CRTM 2.0.2
  - BUFR, BLAS and LAPACK

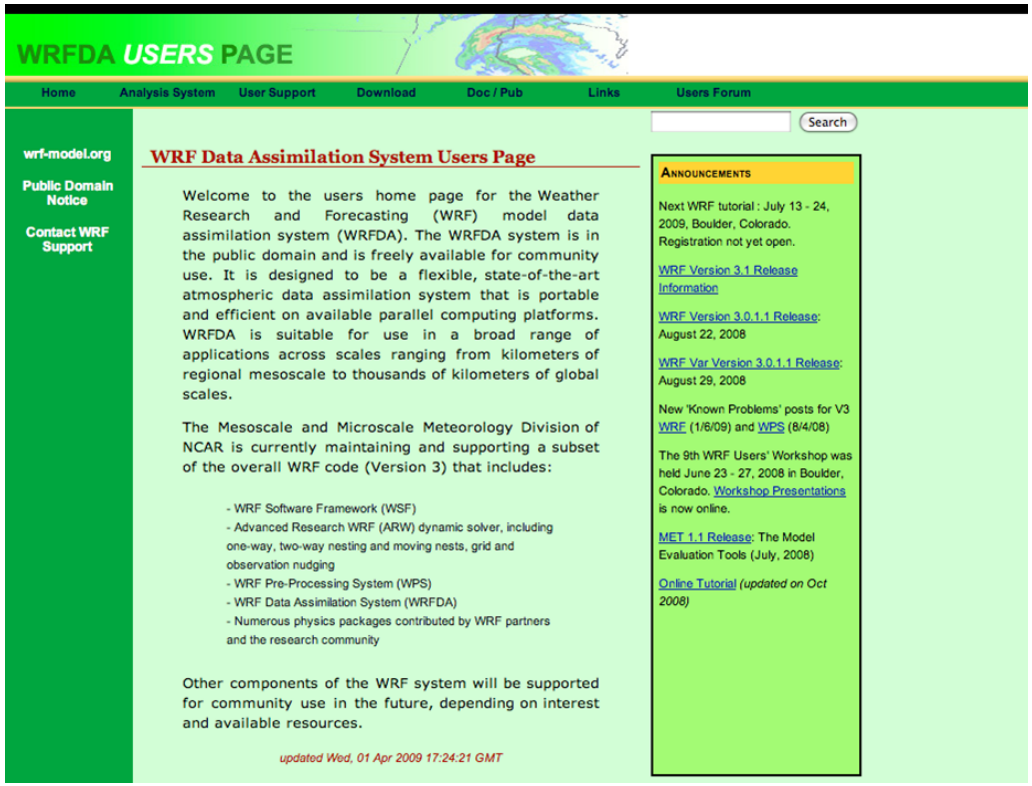
- Install NetCDF (V3.6 above) with THE SAME COMPILER you will choose to compile WRFDA codes.
- Setup the environmental variable
  - csh, tcsh : setenv NETCDF *your_netcdf_path*
  - bash, ksh : export NETCDF=*your_netcdf_path*
- cd WRFDA
- ./clean -a
- ./configure (-d) wrfda (-d : compile with debug mode)
- ./compile all_wrfvar
- 41 executables should be generated under var/build directory

Upgraded to Mac OS X Snow Leopard Users

- PGI (v10.3.0, 64-bit), G95(v4.0.3, 32-bit), GFORTRAN(v4.4.0, default is 32-bit, '-m64' needed for 64-bit) have been tested.
- gcc, g++ are verision 4.2.1, default is 64-bit, '-m32' needed for 32-bit.
- NetCDF library should be re-install with appropriate compiler and option.
- V3.3 configure is able to produce configure.wrf for PGI and G95.
- For GFORTRAN compiler, '-m64' should be added to SFC,SCC, CCOMP in configure.wrf manually.
- Prepbufr and Bufr format data have problem to be read on Snow Leopard system.

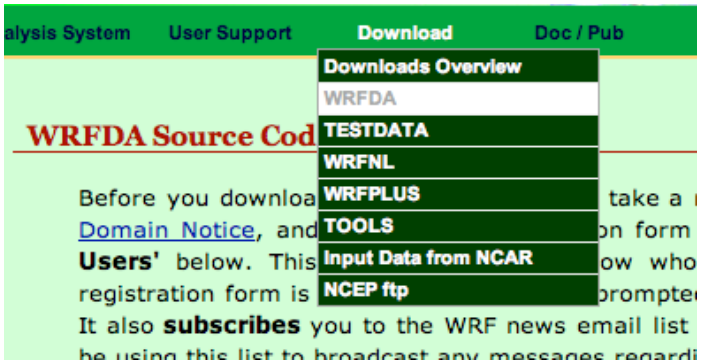
Online WRFDA Resources

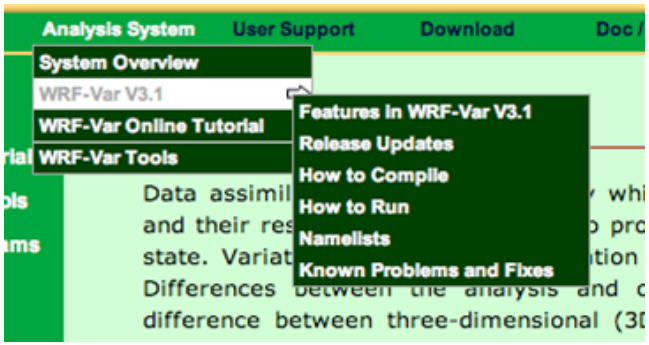
WRFDA has a dedicated page, similar to the ARW Users page:
WRFDA User Page



Online Resources

From the WRFDA page, one can access:





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