Lecture 2: PWV Processing from GNSS using GAMIT/GLOBK

DARA Training School · September 2026 · Workshop lecture

Lecture agenda

- 1. PWV science: ZTD → ZWD → PWV
- 2. Workshop setup, data, and script overview
- 3. Walk through Steps 1–8 (live / guided)
- 4. Interpreting met plots and QC
- 5. Extensions, troubleshooting, wrap-up

Part A · Why GNSS is a meteorological sensor

FROM TROPOSPHERIC DELAY TO PWV

Learning outcomes

- Define TZD/ZTD, ZHD, ZWD, and PWV and how they relate
- Configure GAMIT sestbl. for meteorological products
- Execute the DARA workshop pipeline with dara_gamit_globk_steps.py
- Run sh_metutil and plot PWV / delay time series for GBGA
- Know which files to check when a step fails

Definitions

- GNSS — Global Navigation Satellite Systems (here: primarily GPS processing)
- TZD / ZTD — Tropospheric / Zenith Total Delay (hydrostatic + wet)
- ZHD — Zenith Hydrostatic Delay (from surface pressure / models)
- ZWD — Zenith Wet Delay (ZTD − ZHD); sensitive to water vapour
- PWV — Precipitable Water Vapour (column water; mm), derived from ZWD
- GAMIT / GLOBK — processing + combination software from Lecture 1

Physical chain (conceptual)

- Signal delay in the neutral atmosphere ≈ hydrostatic + wet components
- GAMIT estimates zenith delay parameters (and optionally gradients)
- With reliable surface pressure → remove ZHD → residual ZWD
- Conversion ZWD → PWV uses a dimensionless factor Π(Tm)
- Mean temperature Tm from models or surface temperature parameterisations
- Workshop uses sh_metutil to combine o-file delays with RINEX met

PWV quality inherits GAMIT delay quality and the accuracy of pressure / Tm.

Why GAMIT settings matter for PWV

- Met obs source — where pressure/temperature come from (RNX + GPT, etc.)
- Output met = Y — write meteorological products needed downstream
- Mapping functions / ionosphere / magnetic field / yaw — consistency & accuracy
- I-file output helps inspection of intermediate products
- Elevation cutoff and gradient estimation influence delay biases
- Workshop sestbl. edits are mandatory — defaults are not PWV-ready

Part B · GBGA · 2023 DOYs 277–278 · interactive Python driver

WORKSHOP CONTEXT

Workshop data & environment

- Station: GBGA (local African site used in DARA exercises)
- Days: 2023 DOY 277–278 (directory naming often 277-278/2023)
- Server workflow: SSH to the course GNSS machine (see steps sheet)
- Source RINEX / tables may live under a shared path (e.g. DARA2025/2023/…)
- Goal: clean run → GAMIT → GLOBK → metutil → PWV plots
- Work sequentially: the script enforces order (complete prior steps first)

Companion docs: Workshop_Exercise_Steps.pdf and dara_gamit_globk_steps.py

Example network partner residuals — WIND

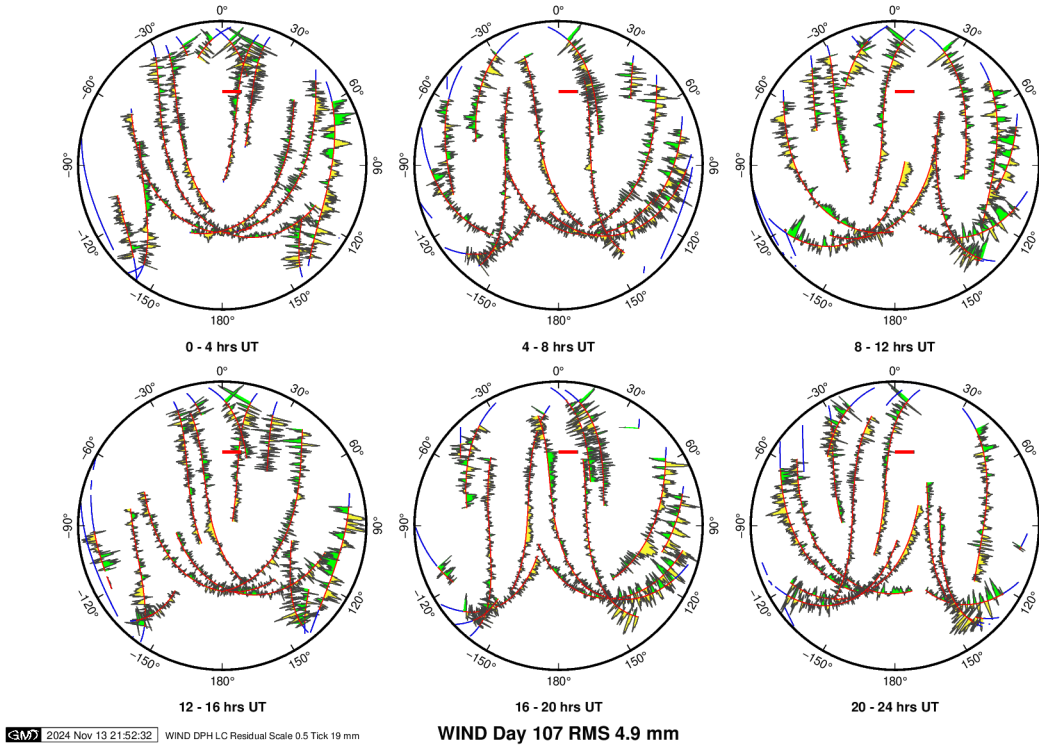
Workshop networks include local GBGA plus IGS/reference sites

Compare residual behaviour across sites

Healthy partners support ambiguity resolution and frame

Same skyplot style you will generate with -pres ELEV

Use differences between sites to spot local multipath / met issues



DARA example skyplot: WIND day 107 (reference/partner site).

The interactive menu (script)

- 1. Clean up Directories
- 2. Data Pre-processing
- 3. Set up Tables
- 4. Update Station Info
- 5. Run Gamit
- 6. Run Globk
- 7. Run Metutil
- 8. Plot Met data
- 0. Exit

Run: python3 dara_gamit_globk_steps.py — choose 1→8 in order.

Manual steps sheet ↔ script

Workshop Exercise Steps (manual)	Script menu automation
gzip -d / CRX2RNX / gfzrnx	Option 2 Data Pre-processing
Edit sites.defaults + IGS map	(prepare before / during 1–3)
sh_setup -yr 2023	Option 3 Set up Tables
sed edits to sestbl. (met-related)	Option 3 (embedded in setup_tables)
sh_upd_stnfo -files ../rinex/*23o	Option 4 Update Station Info
sh_gamit -expt dara ...	Option 5 Run Gamit
sh_glred ...	Option 6 Run Globk
cp met + sh_metutil	Option 7 Run Metutil
plot met products	Option 8 Plot Met data

Use the script for the practical; the PDF remains your human-readable checklist.

Part C · Prepare a clean, PWV-ready experiment

GUIDED STEPS 1–4

Workshop Step 1: Clean up Directories

Menu option 1

Resets the working experiment to a known starting state

Removes previous day/processing debris (keeps rinex/ & tables/ scaffolding)

Copies GBGA RINEX for the DOY window from the shared source tree

Restores sites.defaults from the shared tables template

Extracts start/end DOY from the path name (expects …/277-278/…)

Key commands / script actions

```
# Script: clean_up_dir()
# Copies GBGA00NAM_R_2023{doy}*.rnx.gz / *.crx.gz into rinex/
# Re-copies sites.defaults; strips stale table links/files
```

Workshop Step 2: Data Pre-processing

Menu option 2

Unzip .gz archives in rinex/

Hatanaka decompress .crx → .rnx (CRX2RNX -d)

Convert RINEX3 → RINEX2 GPS-only with gfzrnx

Move gbga*.23* products into rinex/ with expected short names

Infers start_doy / end_doy from resulting gbga files (277 & 278)

Key commands / script actions

```
gzip -d rinex/*.gz
CRX2RNX -d rinex/*.crx
gfzrnx -finp <file> -vo 2 -satsys G -fout .::RX2::
```

Step 2 notes from a real test run

- gfzrnx may warn: MARKER NAME GBGA vs file name GBGA00NAM — using GBGA
- RINEX3 → 2.11 drops some modern header labels (expected)
- Confirm outputs like gbga2770.23o / gbga2780.23o (and .23m met if present)
- If no .gz / .crx / .rnx found, stop and check Step 1 copy paths
- Do not proceed to tables until short-name RINEX2 files exist

Workshop Step 3: Set up Tables

Menu option 3

Runs sh_setup -yr 2023 to populate tables/

Then applies PWV-oriented sestbl. sed edits automatically

Links / copies station.info, sestbl., sittbl., autcln.cmd, apr files, …

Verify Met obs source and Output met after the script returns

Still review sites.defaults for your network selection

Key commands / script actions

```
sh_setup -yr 2023
# sestbl. edits inside tables/:
Met obs source = RNX GPT 50
Output met = Y
Ion model = GMAP
Mag field = IGRF13
Yaw Model = Y
I-file = Y
```

Choosing the processing network

- Update sites.defaults: local GBGA + surrounding IGS/reference sites
- Use the IGS map to pick geometrically sensible partners
- https://igs.org/imaps/map.html#!/?network=igs&view=horizontal
- Too few sites → weak geometry / frame; too many → slower, harder QC
- Public sites can be flagged for download; local site must already be in rinex/
- Keep experiment name consistent (dara) across GAMIT and GLOBK

Workshop Step 4: Update Station Info

Menu option 4

Merges metadata using sh_upd_stnfo

First pass can use sites.defaults list (-l sd)

Then updates from local RINEX headers for GBGA days

Produces station.info.new → becomes station.info

Read warnings about early session end times carefully

Key commands / script actions

```
cd tables/
sh_upd_stnfo -l sd ; mv station.info.new station.info
sh_upd_stnfo -files ../rinex/gbga2770.23o
sh_upd_stnfo -files ../rinex/gbga2780.23o
```

Part D · Estimate delays (GAMIT) and frame coordinates (GLOBK)

GUIDED STEPS 5–6

Workshop Step 5: Run GAMIT

Menu option 5

Launches full daily processing for the DOY range

Uses IGS final orbits and elevation residual plots

Creates day directories 277/ and 278/ with o-files / h-files

Expect makex / model / autcln / solve log traffic — normal

Watch for station.info window warnings and missing-met warnings

Key commands / script actions

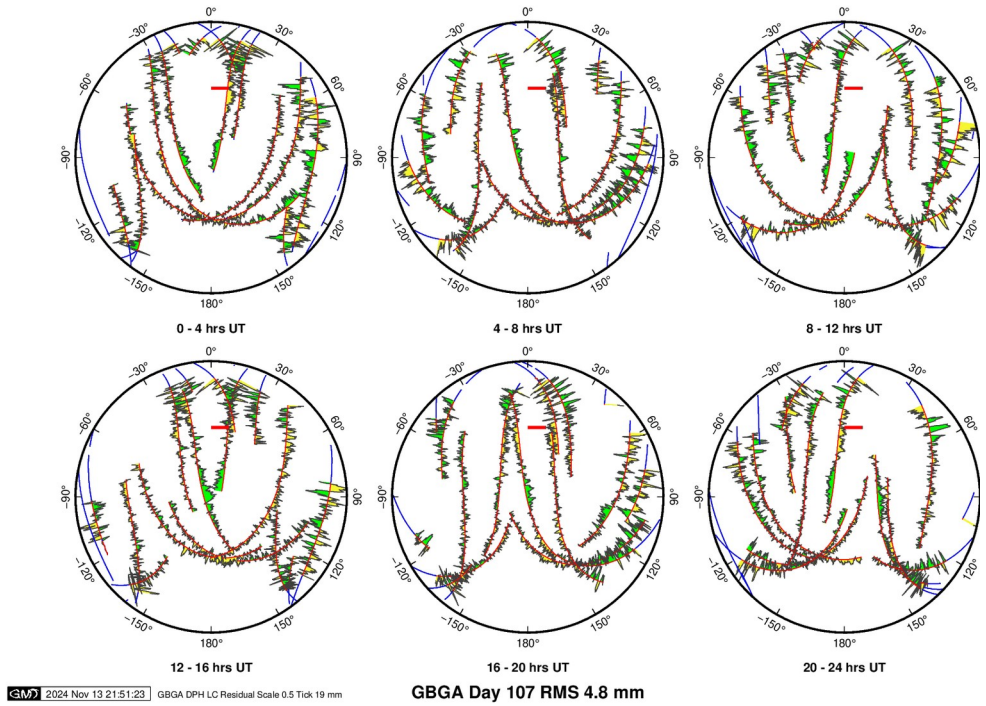
```
sh_gamit -expt dara -c -s 2023 277 278 -pres ELEV \
         -orbit IGSF -copt x k p -dopt c ao
```

During / after Step 5 — what success looks like

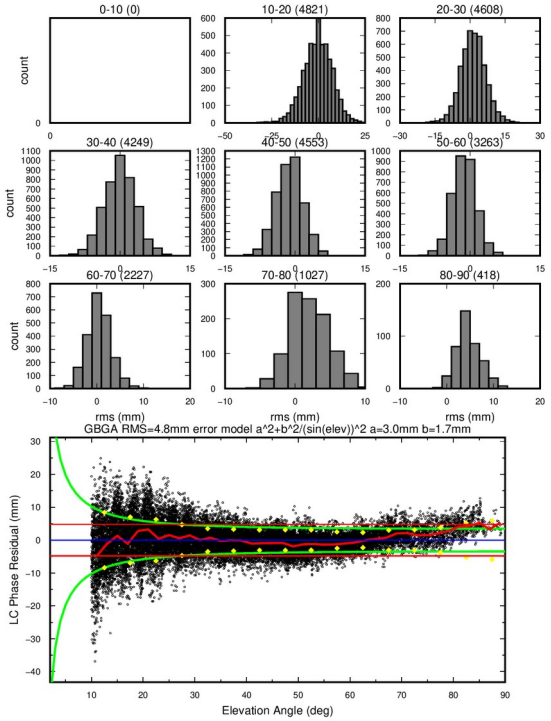
- Day directories exist with summaries and residual .ps plots
- ASCII h-files produced for each day / experiment
- GBGA present in the site list of the solution
- Phase residuals not dominated by huge systematics
- If sh_gamit aborts: read the first ERROR in the log, fix, then re-run
- Do not start GLOBK until both days finish cleanly

What to inspect after GAMIT — GBGA residuals

Skyplot DPH LC residuals



Residual vs elevation



Trust PWV only after residual QC looks physically sensible (example from DARA practicals).

Workshop Step 6: Run GLOBK

Menu option 6

Initialises command files if needed (sh_glred -cmd)

Converts h-files, runs daily combinations for the window

Builds short time series / plots for QC of coordinates

Confirms the geometric solution before trusting troposphere use

Workshop window is only 2 days — interpret series cautiously

Key commands / script actions

```
sh_glred -cmd
sh_glred -s 2023 277 2023 278 -expt dara -opt H G T
```

Step 6 outputs to glance at

- glbf/ binary h-files (e.g. h2310041200_dara.glx)
- gsoln/ print/log/org products from glred
- pos_…/GBGA.*.pos and plots_… residual postscript
- Confirm GBGA (and stabilisation sites) appear as expected
- Large coordinate outliers → revisit station.info / RINEX / day QC

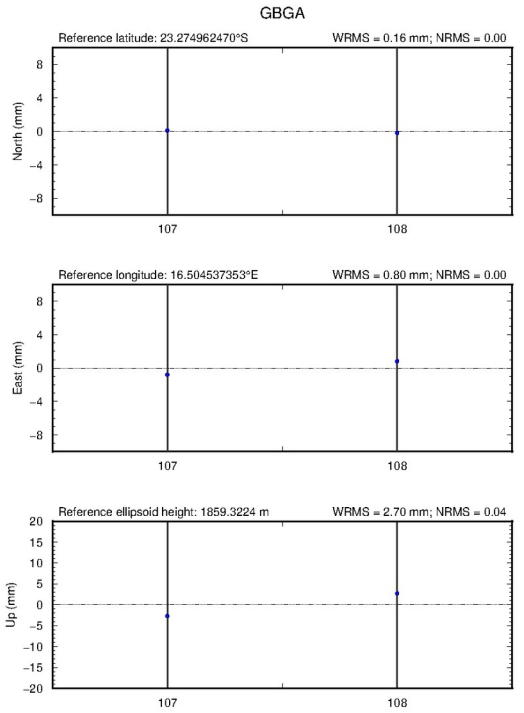
Example coordinate QC before PWV

Short NEU series from sh_glred / sh_plot_pos

Stable horizontal + plausible Up scatter

If Up is wild, troposphere (and PWV) may be compromised

Only then proceed to sh_metutil



DARA example: GBGA NEU residuals after a two-day combination.

Part E · Convert delays to PWV and plot

GUIDED STEPS 7–8

Workshop Step 7: Run Metutil

Menu option 7

For each DOY directory: copy the RINEX met file into the day folder

Run sh_metutil against the GAMIT o-file and the met file

-i 300 → 300 s (5 min) sampling of the met util products

Writes met_* products / logs (e.g. metutil.odaraa.277.log)

Requires Output met path from GAMIT to have succeeded

Key commands / script actions

```
cd 277/
cp ../rinex/gbga2770.23m .
sh_metutil -f odaraa.277 -m gbga2770*.23m -i 300
# repeat for 278/
```

What sh_metutil is doing

- Reads tropospheric estimates from the GAMIT solution (o-file)
- Combines with meteorological observations (pressure, temperature)
- Forms delay / PWV time series suitable for plotting and comparison
- Interval flag balances noise vs temporal resolution
- If met RINEX is missing, fix data prep — GPT alone may be insufficient for the exercise
- Check the metutil.*.log for “Finished” and for skipped epochs

Workshop Step 8: Plot Met data

Menu option 8

Copies met_gbga.YYDOY files into a met/ directory

Runs plot_metutil.py on each available met file

Inspect PWV and delay behaviour across the two days

Compare diurnal pattern, magnitude, and gaps

Save / export plots for your workshop report

Key commands / script actions

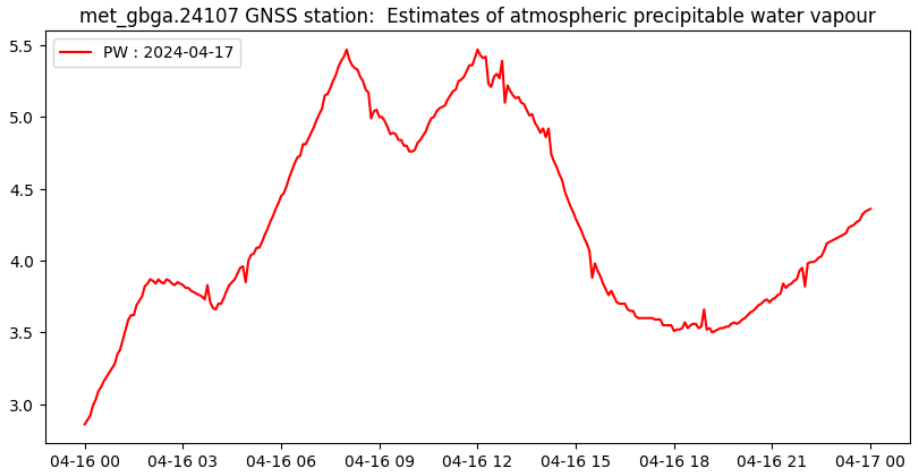
```
cp 277/met_gbga.23277 met/
cp 278/met_gbga.23278 met/
cd met/
plot_metutil.py met_gbga.23277
plot_metutil.py met_gbga.23278
```

Interpreting the plots

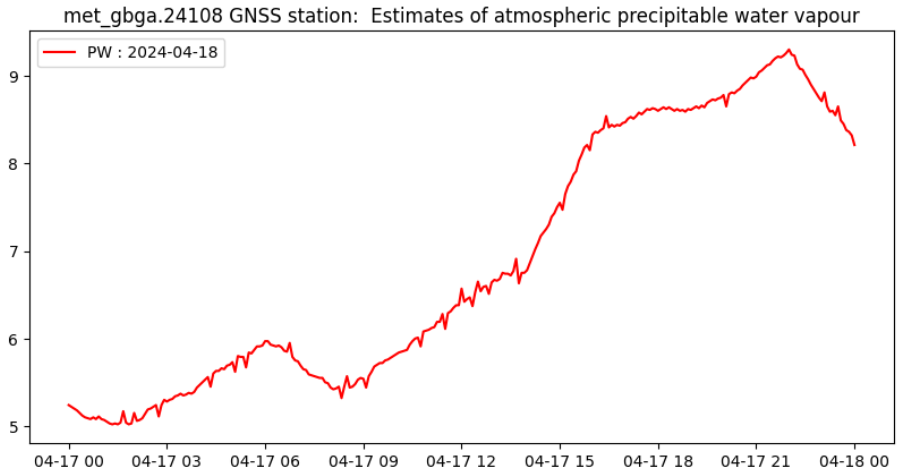
- PWV in mm: typical tropical/subtropical values often tens of mm — context matters
- Look for smooth physical evolution; spikes may be solution or met outliers
- Gaps often track missing epochs, cleaning edits, or met dropouts
- Compare day 277 vs 278: weather change vs processing artefact?
- If coordinates from Step 6 look poor, treat PWV as suspect
- Optional: compare against radiosonde / ERA5 / nearby GNSS if available later

Example PWV products — met_gbga

met_gbga.24107



met_gbga.24108



DARA example outputs from plot_metutil.py — diurnal PWV evolution for GBGA.

Part F · When the menu says no — or the plots look wrong

TROUBLESHOOTING & GOOD PRACTICE

Script behaviour to remember

- Options must be run in order; skipping triggers “complete previous steps”
- step_done gating starts after a successful Step 1 in a session
- If you restart Python, begin again from Step 1 or know your directory state
- clean_up_dir expects DOY range in the path (e.g. 277-278)
- Hard-coded year 2023 and station GBGA in the provided script
- Read stdout carefully — many “warnings” are informational (gfzrnx headers)

Troubleshooting matrix

- No files in rinex after Step 1 → shared source path / DOY pattern / permissions
- gfzrnx fails → input not true RINEX3, or tool not on PATH
- sh_setup missing tables → ~/gg link or year tables incomplete
- sh_gamit site missing → sites.defaults / station.info / RINEX name
- sh_metutil fails → missing .23m, wrong o-file name (odaraa.DOY), wrong cwd
- Empty plots → met_* not created; re-check Step 7 logs

Good practice beyond the exercise

- Document sestbl. and sites.defaults with the PWV product
- Archive raw RINEX + met + key summaries, not only final plots
- Use longer span and more sites for research-grade PWV
- Validate against independent humidity observations when possible
- Keep GAMIT/GLOBK tables updated (antennas, DCBs, metadata)
- Separate “tutorial success” from “climate-quality reprocessing”

Part G · You now have a repeatable PWV path on GAMIT/GLOBK

WRAP-UP

Checklist — end of workshop

- Experiment directory cleaned and GBGA RINEX present
- RINEX2 short names for DOYs 277–278
- sestbl. shows Output met = Y and RNX GPT met source
- station.info contains GBGA with sensible antenna / height
- sh_gamit finished for both days
- sh_glred produced framed solutions / plots
- sh_metutil logs show Finished for both days
- plot_metutil.py figures inspected and saved

Key takeaways

- PWV from GNSS is a by-product of high-quality geodetic delay estimation
- Lecture 1 workflow + met-aware sestbl. + metutil = workshop chain
- The Python menu mirrors the written exercise steps — use both
- QC the geodesy before trusting the meteorology
- You can extend the same recipe to more days, sites, and comparisons

Resources

- Workshop_Exercise_Steps.pdf — manual command checklist
- dara_gamit_globk_steps.py — interactive automation for Steps 1–8
- MIT GAMIT/GLOBK documentation and course notes (geoweb.mit.edu/gg)
- IGS network / products portals for reference sites and orbits
- Course facilitators: ask early when a step fails — logs first, then re-run

Questions & hands-on time

- Start the script on the server and complete Steps 1–3 together
- Then proceed at your own pace through GAMIT → met plots
- Flag blockers with the failing step number
- Open discussion: applications of GNSS-PWV in your country / project