



Plenary discussion on plans to develop a portable AMV processing software package

Johannes Schmetz and Jaime Daniels





CGMS 37 recommends development of a portable AMV software package

Recommendation 37.22:

IWWG 10 is requested to address the development of a stand-alone software package for the derivation of AMVs from imaging satellite instruments.



What could be the benefits of a portable AMV package?

- 1) A common software tool for testing new methods/ideas in transparent way; i.e. The new result could be reproduced by others.
 - 2) Attractive to researchers 'outside' the IWW community to conduct research and test novel ideas. Attractive because the initial step to produce a useful result is much simpler. This in turn would benefit the progress of AMV research and applications
 - 3) ...
- It could also turn out to be a lot of work for relatively little gain !



Are there examples?

- **International ATOVS Processing Package**

The International ATOVS Processing Package (IAPP) has been developed to retrieve atmospheric temperature and moisture profiles, total ozone and other parameters in both clear and cloudy atmospheres from ATOVS radiance measurements. The IAPP algorithm, which operates on NOAA-15 and later data, retrieves the parameters in 4 steps: 1) cloud detection and removal; 2) bias adjustment; 3) regression retrieval; and 4) nonlinear iterative physical retrieval.

A publication by Li, Wolf, Menzel, Zhang, Huang and Achtor, *Journal of Applied Meteorology* (August 2000) provides details on the algorithm.

=> This has been and is widely used and fosters the use of ATOVS data



What can we achieve at this workshop ?

- An agreement and recommendation to go ahead or not
- If yes:
 - We should develop ideas towards a realisation
 - Task a team to define a way forward
 - Write a paper to CGMS (from Co-chairs and rapporteur) which defines the idea in detail and also addresses the needs for funding



Your Thoughts?...

- What do we hope to achieve by doing this? End goal=??
- Any additional benefits for doing this?
- How might we achieve this?
- Any lessons learned from other similar efforts?
- What are considerations for doing this? ie., design, development, management and coordination of, resource needs and \$\$ needs, maintenance of, etc, etc



Advice from other Similar Projects

- Agree which language it should be written in
- Have good configuration control for the software
- Advertise it well to get user take up
- Have beta testers for the code
- Always compile with the F90 NAG compiler (if using Fortran)
- Have really good administrator
- Have good documentation
- Provide and enforce coding standards / guidance
- Hold a regular discussion forum
- Avoid code becoming too 'native' i.e. it works great for one centre but is useless for others
- Main challenge with RTTOV is with integration (less of a problem for standalone) – simplify inputs/outputs as far as possible
- Will need to develop a test system and test multiple configurations (all takes time).
- Don't produce full updates too frequently (time overhead in testing), ? once per year at most
- Consider copyright issues (ROPP has some examples).