Overview of the coding rules in Arpege/Ifs/Aladin

R. El Khatib

Météo-France – CNRM/GMAP

September, 2002

Constraints

- Double aspect for the same code : operations & research => maintenance, flexibility, portability...
- Cross-collaborations : we need to speak the same language
- Portability on various platforms (various compilers, technologies)
- Perpetual scientific/technical evolutions to be achieved with minimum time & human resources

Solutions

- A set of ***agreed*** & ***justified*** coding rules
- The choice of a unique code language
- Further criterias to ensure the code quality :
  - Code analysis
  - Proper writing manners (???)
  - Documentation
  - Portability
  - Efficiency
  - Flexibility
  - Exchangeability

Historical highlights

- « DOCTOR » norms (mainly : variable prefixes) : ECMWF, 1986
- European standards for writing and documenting exchangeable Fortran 90 code, 1995
- Miscellaneous other documents/initiatives

Specifications for the documentation

- Documentation is ESSENTIAL !!!

  : to understand the spirit

- External documentation :
  - A scientific one,
  - A technical one,
  - A users guide
- Internal documentation :
  - Comments

Specifications for the code conception

- Future enhancements to be anticipated
- Modularity/hierachical design ; no duplication of code
- Simple modules & relashionships
- Confined dataflow (cf. portability)
- Non-standard aspects to be banned or confined

Specification for the code validation & maintenance

- Tests to be planned
- To control that all the rules have been followed
- Documentation to be UPDATED !!!

Overview of the code design

- Typewriting style
- Layout of the code
- Header comments
- Declaration of variables
- General coding rules
- Software-specific coding rules

About source code management

- Systematic use of a code management software to ensure the historization
- Further rules to clarify the code structure
- Makes merging operations more secure

Going further

- Rules can hardly be respected without an automatic verificator (Robocop syndroma !)
- New language features are suggesting new coding rules … to be tested first !
- Old rules/habits may be penalizing

Conclusion

- Read the *Coding rules tutorial*
- Do your best *for the others*
- *Write extensive documentation and update it !*