



## ***Coding Conventions and Practices***

03/08 – INFN-CNAF Bologna

Marco Emilio Poleggi – *INFN-CNAF*

Marco.Poleggi@cnafe.infn.ch



- ▣ Conventions
- ▣ Code quality assessment
- ▣ Open issues

- Recommendations for NCM components here  
<https://twiki.cern.ch/twiki/bin/view/ELFms/WriteNC>
- I still believe a general, high-level document is advisable
  - + Coding style
    - Indentation
      - New modules: 4 blanks instead of TABs
      - Old modules: clean-up before coding new parts and keep profile (always prefer blanks)
    - Fold very long lines
      - Ideally break at 80th column
  - + Use the build tool's facilities rather than direct CVS access
  - + Header with meta-data (author, maintainer, etc.) in source files
    - make header
  - + Other good recommendations here:
    - <http://cons.web.cern.ch/cons/perl/guide/>

## □ Documentation

- + Standard formats and nothing else to minimize maintenance
- + CVS
  - READMEs with minimal information about files and non-RPM-based installation
  - Technical documents for developers in LaTeX
  - Perl POD and similia for generating man pages
    - References to Web documents in section “MORE INFORMATION”
  - Convert, mark or remove obsolete files
- + WWW
  - Wiki pages
    - TWiki produces also PDFs, what about Trac?
  - No HTML, unless automatically generated
- + Synchronization: ideally, Web documents should reflect changes in man pages only

- Always use `strict`
- Use `LC::Exception` when possible
- Provide unit tests
- Try `B::Lint` (rel. 1.11)
  - + Still experimental but can spot common pitfalls and/or potentially dangerous practices
    - Cannot be enforced ;-)
  - + Stimulates deeper investigations

```
[poleggi@pcpoleggi:aia] perl -MO=Lint,all aia-shellfe
Bare sub name 'MODULEBASE' interpreted as string at aia-shellfe
line 284
Bare sub name 'USEMODULE' interpreted as string at aia-shellfe line
...
Use of $_ at aia-shellfe line 587
Implicit use of $_ in foreach at aia-shellfe line 594
...
```

## □ Call wrapper facilities

- + Enforce `LC::Process`?
  - With caveats (see bug #31194)
  - Maybe `IPC::Open3` instead (needs wrapper)
- + Wrap `open()` in order to log which files are touched?
  - It could rely on `CAF::Log`

## □ Code reviews? For checking, f.i.:

- + `exit()` calls in components
- + Ignored return values
- + Consistency of options/functionality with documentation
- + ...
- + But no deep dives, unless mission-critical ;-)

 **quattor**

<http://quattor.org/>