

# Debian and RPM packages for ABINIT?

Yann Pouillon, X. Gonze

*Physico-Chimie et Physique des Matériaux  
Université Catholique de Louvain  
1348 Louvain-la-Neuve (Belgium)*

# Overview

---

- Why packages?
- Conforming to standards
- Using Autotools
- Packaging ABINIT
- Version management

# Why packages?

---

- Show that ABINIT has reached a certain maturity
- Promote original structure of source tree
- Enhance visibility within free software community
- Foster free software development model
- Simplify access and use for a broader audience
- Ease education and training
- Get help from outside more easily

# Conforming to standards

---

- Follow free software development guidelines
    - GNU coding style + autotools
  - Improve consistency, compilation, and debugging
    - **Assign correct file extensions**
  - Enable system-wide installation (for packages)
    - Filesystem Hierarchy Standard (UNIX)
- ⇒ Modify source code structure
- ⇒ Transfer configure process to autotools

# Using GNU Autotools

---

- Autoconf: generate a *configure* script  
→ Fortran 90/95 support not yet complete
- Automake: in each subdir: *Makefile.am* → *Makefile.in*
- Libtool: build (shared) libraries
- Automatic test suite generation
- Build process:  
*./configure [options]*  
*make*  
*[make check]* (important for ABINIT)  
*make install* or *make dist*

# GNU Autotools in ABINIT

---

**Constraint: preserve backward compatibility  
as long as possible**

- Assign correct file extensions (.f, .F90)
- Write *configure.ac* for autoconf
- Write *Makefile.am* files (currently: mostly automatic)
- Move some files and add some other
- Write a script to restore previous structure

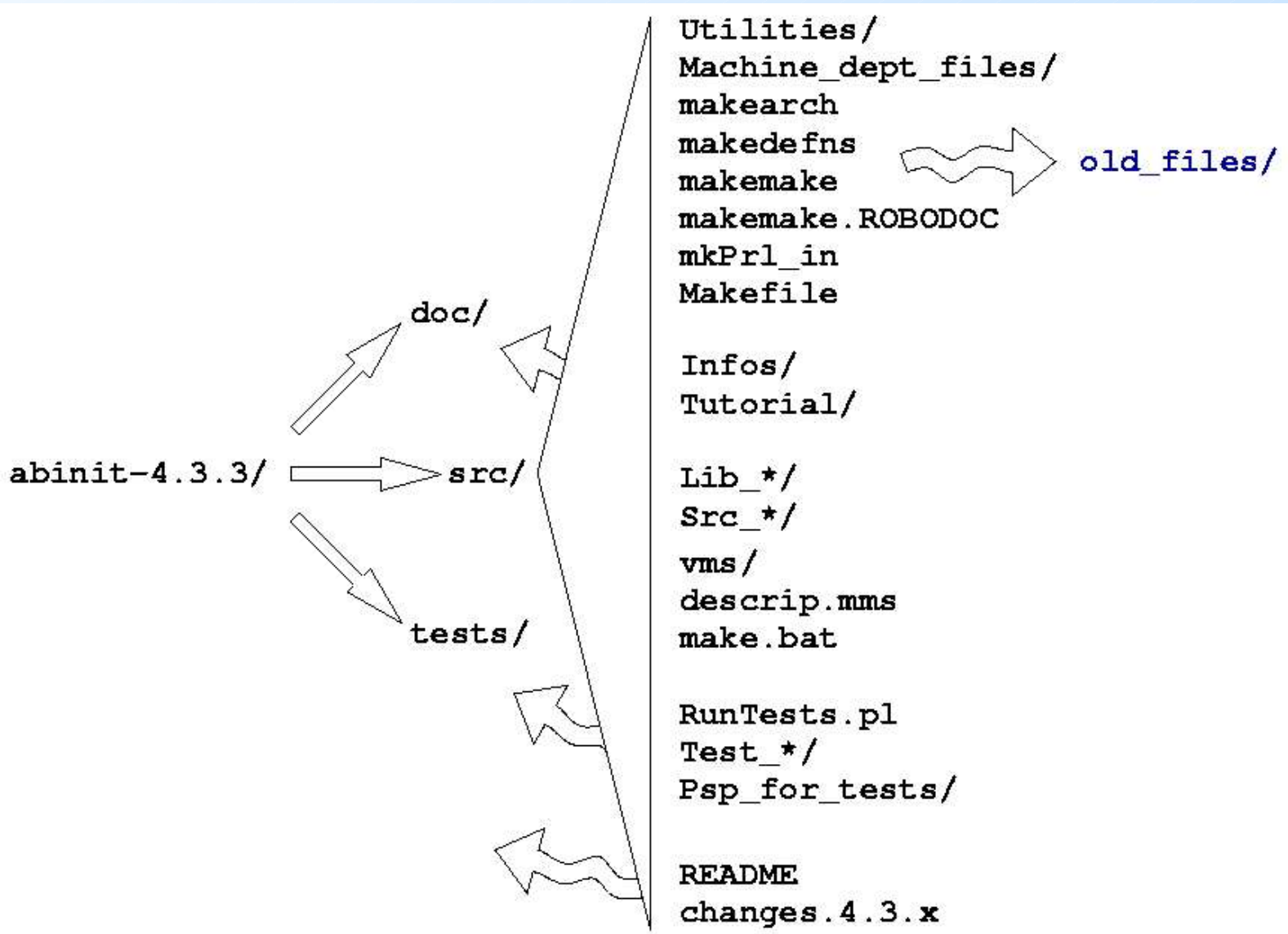
# Packaging ABINIT

---

- Requirements
  - write one man page for each executable
    - SGML / DocBook
  - adopt strict naming conventions for files
  - define quantity and composition of packages
  - establish a release policy for the packages
- Package info in *debian/* and *rpm/* subdirectories
- Package building can be done automatically

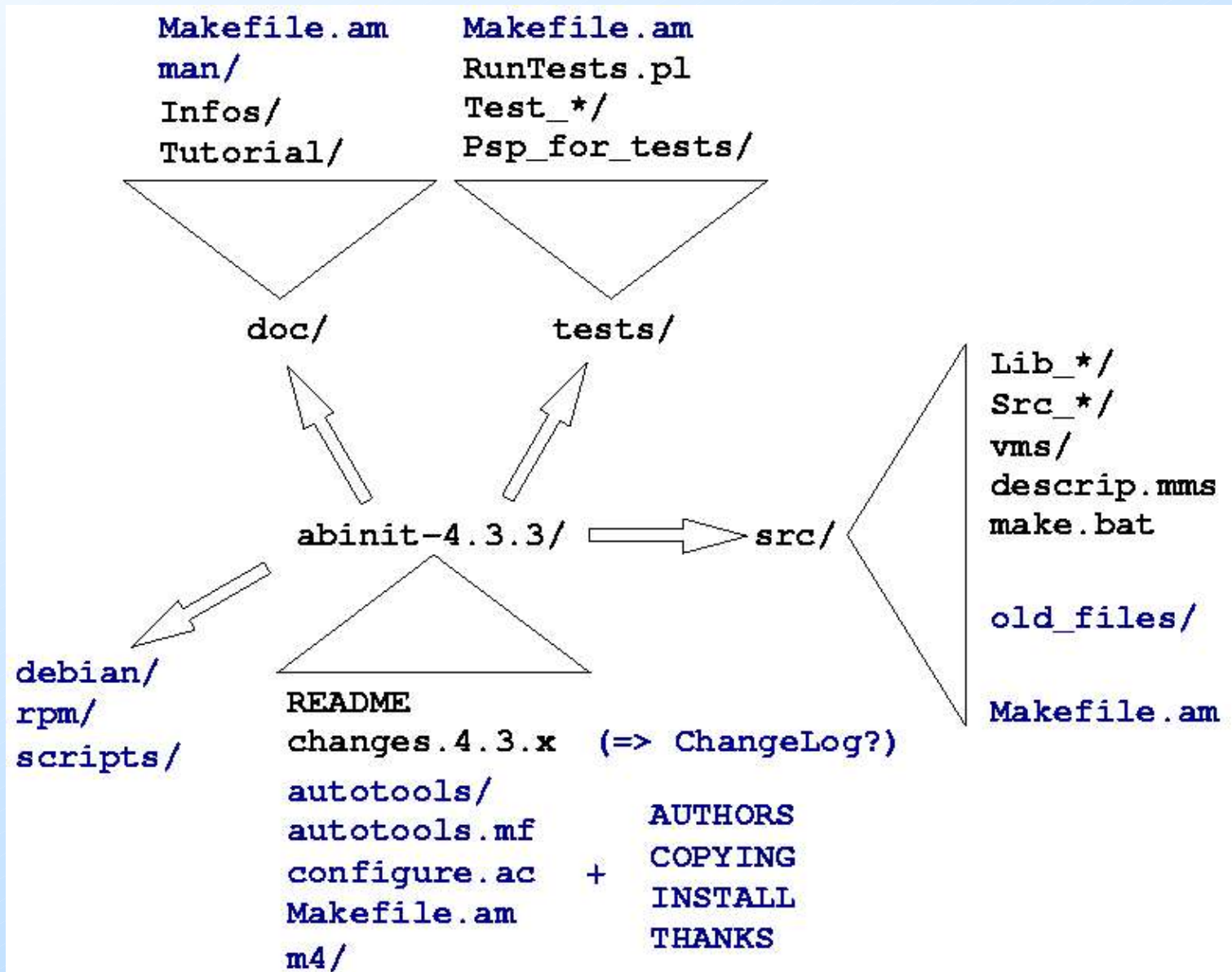


# Current source tree

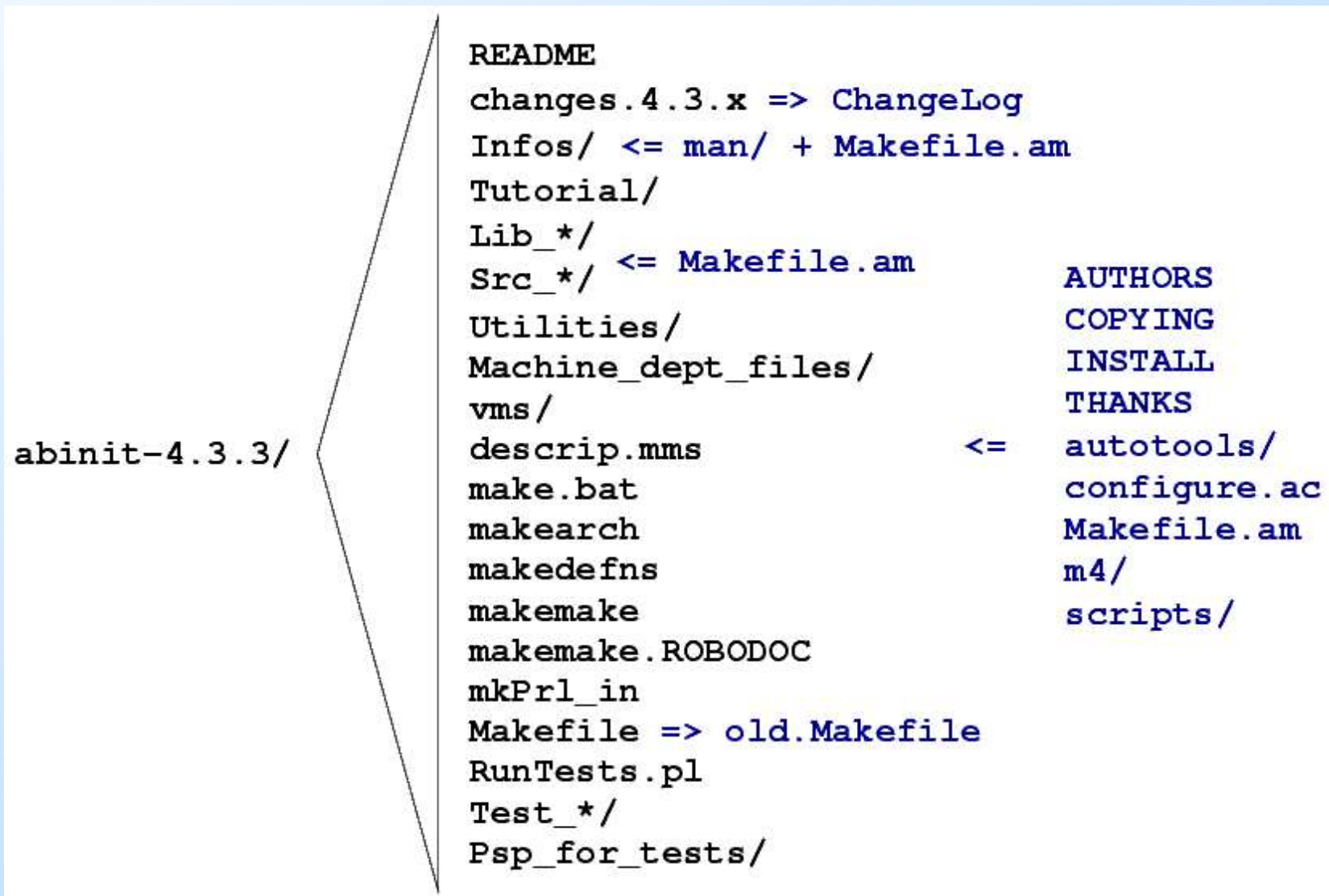




# Proposed source tree



# Transitional source tree



# ABINIT packages

---

- Proposed ABINIT packages
  - *abinit-source*: whole source tree
  - *abinit-doc*: whole documentation
  - *abinit-bin*: sequential and parallel binaries
  - *abinit-testsuite*: all available tests
  - *abinit-data*: data used by the binaries

# *abinit-data*: auxiliary data

---

- Data needed by ABINIT
  - pseudopotentials (~ 80 Mb raw / 30 Mb zipped)
  - PAW atomic data (~ 15 Mb raw / 5 Mb zipped)
  - atomic density files (~ 3 Mb raw)
- Open issue: how to package data?
  - atom per atom (may be a maintainer's nightmare)
  - by column or group of interest
  - other (suggestions?)

# Version management

---

- GNU Arch: give it a try! (<http://www.gnuarch.org/>)
  - Text and binary data handled transparently
  - file naming constraints / renames handed cleanly
  - advanced merging features (history-sensitive)
  - web-browsable raw repository format
  - distributed and private repositories
  - easy server administration
  - bidirectional communication with CVS

# Summary and outlook

---

The discussion is open ...



# Getting GNU Arch

---

- Thomas Lord's implementation
  - Debian packages: *tla*, *tla-doc*, *tla-buildpackage*, *tla-load-dirs*, *tla-tools*
  - RPM packages: *tla*
- Suggestion: use version 1.2 or above
  - support for MD5 checksums
  - support for GnuPG-signed commits
- See also: <http://wiki.gnuarch.org/>