Collaborative software development for nanoscale physics

Yann Pouillon
mailto:yann.pouillon@ehu.es

Nano-Bio Spectroscopy Group
Universidad del País Vasco (UPV/EHU) & CSIC, Donostia-San Sebastián, Spain
European Theoretical Spectroscopy Facility (ETSF)

RMLL 2012
Genève, Suisse — 2012/07/12
1. Reaching the nanoscale

2. The European Theoretical Spectroscopy Facility (ETSF)

3. Software projects

4. Organizing cooperation
Outline

1. Reaching the nanoscale
2. The European Theoretical Spectroscopy Facility (ETSF)
3. Software projects
4. Organizing cooperation
## Quantum effects in research

### Classical view

*Separation, competition*

<table>
<thead>
<tr>
<th>Chemical Physics</th>
<th>Physics</th>
<th>Biophysics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chemistry</td>
<td>Biochem.</td>
<td>Biology</td>
</tr>
</tbody>
</table>

+ separation theory vs. experiment
Quantum effects in research

Classical view
Separation, competition

Quantum view
Union, cooperation

Chemical Physics | Physics | Biophysics
---|---|---
Chemistry | Biochem. | Biology

+ separation theory vs. experiment
A typical example

Green Fluorescent Protein (aka GFP)

\[\text{fluorescence of jellyfish}\]

- Multiscale approach
- Multiple viewpoints
- Coordination required
- Interoperability issues
Limits of quantum methods


http://arxiv.org/abs/1207.1075v1
Density-Functional Theory (DFT)

**Tractable way to model electrons**

\[ E_{tot}[\rho] = E_{kin}[\rho] + E_{ion}[\rho] + E_H[\rho] + E_{xc}[\rho] \]  

(1)

\[ \Psi(\mathbf{r}_1, \ldots, \mathbf{r}_N) = \varphi_1(\mathbf{r}) \times \cdots \times \varphi_N(\mathbf{r}) \]  

(2)

\[ \rho(\mathbf{r}) = \sum_{i=1}^{N} \varphi_i(\mathbf{r}) \]  

(3)

1. Write energy as functional of electronic density $\rho$
2. Order energy contributions by decreasing magnitude
3. Store complexity in the smallest term ($E_{xc}$)
4. Represent electrons by individual wavefunctions
5. Minimize energy of the system
6. Use it as a starting point for further explorations
Brief overview of DFT

Density-Functional Theory

Types
- Local Density Approximation (LDA)
- Generalized Gradient Approx. (GGA)
  - Meta-GGA, hyper-GGA
  - Exact exchange
  - Hybrid approximations
  - High-order kernels
  - Mathematical tricks
  - Error compensation
  - Inconsistencies
  - Self-interaction
  - Tractability

Exchange-correlation

Issues

Basis for:
- Molecular dynamics
- Time-Dependent DFT
  - Bethe-Salpeter
  - GW
- Tight-binding models

Real space
- Direct
- Complex implementation
- For molecules

Localized orbitals
- Non-systematic convergence
- Superposition errors
- For periodic systems

Plane waves
- Systematic convergence
- Resource-consuming
- Efficient use of resources
- Multiscale

Wavelets

Representation

Tuning

Pseudopotentials
- Types
  - Full potential
  - Effective potential
  - Resource-consuming
  - Universal
  - Norm-conserving
- Properties
  - Ultrasoft
  - Projector-Augmented Waves (PAW)
  - Less electrons
  - Transferability issues

All electrons
- Types
  - Pre-conditioning
  - Order N
  - Case by case

Properties
- Universal
- Norm-conserving
- Ultrasoft
- Projector-Augmented Waves (PAW)
- Less electrons
- Transferability issues
Nanoscale physics

**Macroscopic scale**
Classical physics
e$^- = \text{point charge}$
$N_{\text{atoms}} \approx 10^{23}$
0.1 $\mu$m–light years
Collective behaviour

**Nanoscale**

1–100nm
Every atom matters!

**Atomic scale**
Quantum physics
e$^- = \text{wavefunction}$
$N_{\text{atoms}} < 1000$
1 fm–1 nm
e$^- + e^- \neq 2e^-$!

**Accessing the nanoscale**

- Hybrid models
- Multiscale techniques
- Much higher complexity
Nanoscale physics

**Macroscopic scale**

Classical physics
\[ e^- = \text{point charge} \]
\[ N_{\text{atoms}} \approx 10^{23} \]

0.1 µm–light years
Collective behaviour

**Nanoscale**

1–100 nm
*Every atom matters!*

**Atomic scale**

Quantum physics
\[ e^- = \text{wavefunction} \]
\[ N_{\text{atoms}} < 1000 \]

1 fm–1 nm
\[ e^- + e^- \neq 2e^- ! \]

**Accessing the nanoscale**

- Hybrid models
- Multiscale techniques

\{ Build a theoretical instrument! \}

- Much higher complexity
Outline

1. Reaching the nanoscale
2. The European Theoretical Spectroscopy Facility (ETSF)
3. Software projects
4. Organizing cooperation
What is the ETSF?

- A network of ≈ 200 scientists in Europe (+ USA now)
- A knowledge center for theoretical spectroscopy
- An e-infrastructure bridging theory and experiment

- Our tools: state-of-the-art research, free software
- Our philosophy: cooperation, openness, autonomy & flexibility
- Our activities: collaborations, user projects, training, conferences

- Every 6 months: call for proposals
- Regularly: tutorials for users and developers (CECAM, $\psi_k$, . . .)
What is the ETSF?

- A network of \( \approx 200 \) scientists in Europe (+ USA now)
- A knowledge center for theoretical spectroscopy
- An e-infrastructure bridging theory and experiment

- Our tools: state-of-the-art research, free software
- Our philosophy: cooperation, openness, autonomy & flexibility
- Our activities: collaborations, user projects, training, conferences

- Every 6 months: call for proposals
- Regularly: tutorials for users and developers (CECAM, \( \Psi_k, \ldots \))

Resources

- Leading teams developing cutting-edge theories
- Coordinated software projects: codes, tools, libraries, standards
- Full autonomy of each project
Where is the ETSF?
A synchrotron-like e-infrastructure

photo-emission

time-resolved

X-ray

energy loss

transport

optics
Networking in the ETSF

Website: http://etsf.eu/

+ Wiki (MediaWiki)
+ Social network (Elgg)

- In addition: individual websites of projects
- In progress: Single Sign-On (SSO)
Software development in the ETSF

- Forge (Bazaar, Subversion, Loggerhead) + Test Farm (Buildbot)
- > 20 [ arch + compilers + libraries ] combinations available
- Nightly builds + on-demand operations
- ≈ 100 developers involved in 15 projects
- Experiments on Launchpad
Outline

1. Reaching the nanoscale
2. The European Theoretical Spectroscopy Facility (ETSF)
3. Software projects
4. Organizing cooperation
Free software in the ETSF
LibXC project (2005 – now)
- gather existing routines
- unify calling interfaces
- implement & test in Octopus
- deploy in other codes

Implementation
- language: C, with Fortran 90 interface
- all LDA parametrizations available
- most GGA already implemented
- meta-GGA still work in progress
- status: stable
- APIs available: 1.0, 1.1, 1.2

License: LGPL
Libpspio: get rid of the data conversion headache!

Goal: seamless exchange of pseudopotential data
- up to now: many formats existing
- XML file format available
- no consensus yet on its use

Take two: create a library
- 2009–2010: design of precise specifications
- September 2011: coding party in Coimbra, Portugal
- status: under development
- first stable version: expected 2013

License: LGPL
Outline

1. Reaching the nanoscale
2. The European Theoretical Spectroscopy Facility (ETSF)
3. Software projects
4. Organizing cooperation
ETSF File Format

- Consensus rapidly reached on a common file format
- Extensions regularly discussed
- Implementation: NetCDF-based / Fortran 90

- Extensible specification + ETSF_IO library (from 2005)
- Library licence: LGPL
- Library status: stable
- API version: 1.0 (Fortran 90 only)
- Bindings implemented and tested in 7 codes (including V_Sim)

- Peculiarity: source code self-generated from specifications
- In progress: version 2.0 (C + Fortran bindings)
ETSF Coding Standards

- Most developments done by PhDs and Post-Docs \(\implies\) very high turnover
- Format: short & easy-to-read document
- Main goal: long-term maintainability of software
- Facilitate adoption by not being too constraining
- Common sense and genericity

- Companion wiki
  \(\implies\) *ETSF Developers’ Reference*

- Biennial developer schools (with CECAM)
Developer schools

- Purpose: help developers conform to coding standards
- Duration: one week
- Mornings: interactive lectures / Afternoons: hands-on

- Version Control Systems (Subversion & Bazaar)
- Python scripting
- Libraries & code reuse
- Language bindings
- File formats
- Debugging
- Profiling & optimizing
- Building source code with the autotools
Packaging efforts: strategy

- Kick-off: discussion w/ Andreas Tille @ RMLL 2009
- Team work → light and flexible management
- Prior agreement from upstream developers
- Many exchanges back and forth with upstream

- Experimental packages: Ubuntu PPA
  https://launchpad.net/~pouillon/+archive/etsf-ubuntu

- Debian metapackages: nanoscale-physics, nanoscale-physics-dev
  http://blends.alioth.debian.org/science/tasks/nanoscale-physics

- Fedora, Gentoo: volunteer initiatives

- Slowed down by GSL 1.14 → 1.15
## Packaging efforts: status

<table>
<thead>
<tr>
<th>Code</th>
<th>Version</th>
<th>Debian</th>
<th>Ubuntu</th>
<th>Fedora</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abinit</td>
<td>6.12.3</td>
<td>In progress</td>
<td>In progress</td>
<td>—</td>
</tr>
<tr>
<td>APE</td>
<td>1.1.0</td>
<td>—</td>
<td>In progress</td>
<td>14, 15, 16</td>
</tr>
<tr>
<td>ASE</td>
<td>3.6.0</td>
<td>In progress</td>
<td>In progress</td>
<td>—</td>
</tr>
<tr>
<td>AtomPAW</td>
<td>3.0.1.2</td>
<td>In progress</td>
<td>PPA</td>
<td>—</td>
</tr>
<tr>
<td>BigDFT</td>
<td>1.2.0.4</td>
<td>In progress</td>
<td>PPA</td>
<td>—</td>
</tr>
<tr>
<td>ETSF.IO</td>
<td>1.0.3</td>
<td>Squeeze</td>
<td>Natty</td>
<td>—</td>
</tr>
<tr>
<td>FoX</td>
<td>4.1.0</td>
<td>In progress</td>
<td>PPA</td>
<td>—</td>
</tr>
<tr>
<td>Libpspio</td>
<td>1.0</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>LibXC</td>
<td>1.1.0</td>
<td>In progress</td>
<td>PPA</td>
<td>—</td>
</tr>
<tr>
<td>Octopus</td>
<td>???</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>SPGLib</td>
<td>1.2</td>
<td>—</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>V_Sim</td>
<td>3.6.0</td>
<td>Wheezy</td>
<td>Oneiric</td>
<td>—</td>
</tr>
<tr>
<td>Wannier90</td>
<td>1.2</td>
<td>In progress</td>
<td>PPA</td>
<td>—</td>
</tr>
</tbody>
</table>
Acknowledgments

- ETSF/Nanoquanta Research Network
- ETSF EU Project (contract 211956)
- MEC: project FIS2007-65702-C02-01 (2009-2011)
- Gobierno Vasco: projects IT-319-07 + ETORTEK-inanoGUNE (2009-2011)
Credits

- \LaTeX\ (beamer, tikz, babel, url, inputenc, fontenc, times)
- Vim, Evince
- VYM
- GIMP, Inkscape, ImageMagick
- GNU Make, Bazaar
- Debian & Ubuntu ecosystems
- Wikipedia, Wikimedia Commons
Thank you for your time!