



Using QE as a library The Sternheimer GW code

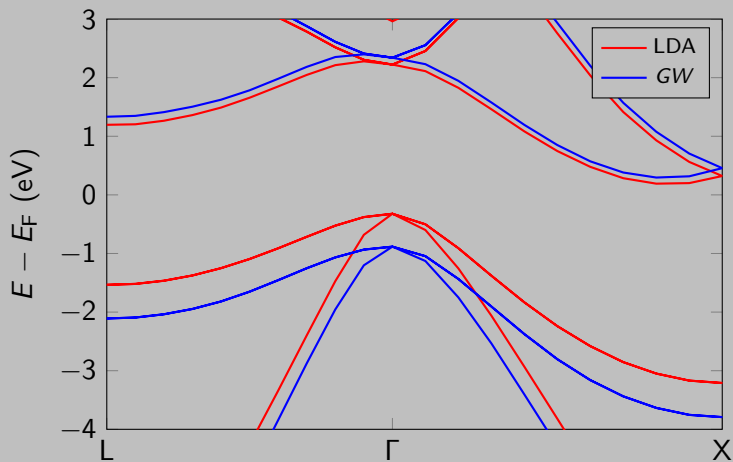
Martin Schlipf

QE developers meeting 2017-01-09



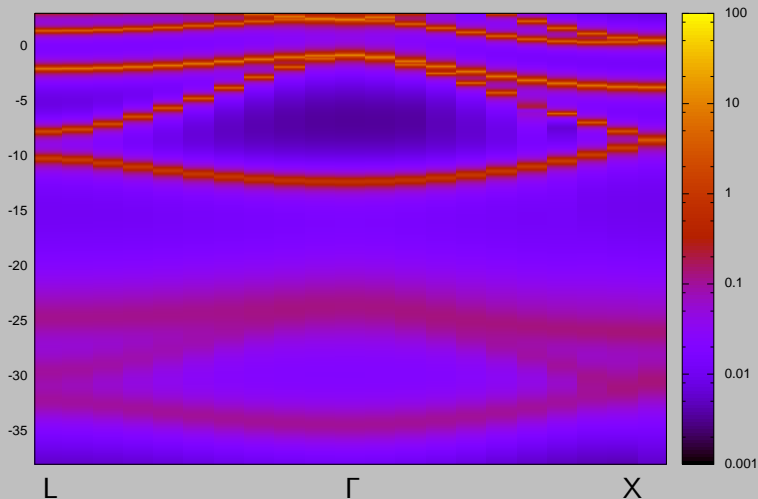


Band structure





Spectral function





The Sternheimer GW method

Green's function

$$(H - \omega)G = -\delta$$

Screened Coulomb interaction

$$(H - \epsilon \pm \omega)\Delta\psi_{v,\pm} = -(1 - P_v)V\psi_v$$

$$\epsilon = \delta - 2 \sum_{v,\pm} \psi_v^* \Delta\psi_{v,\pm}$$

$$W = \epsilon^{-1}V$$



SGW testsuite for buildbot

Silicon

- ▶ inversion symmetry
- ▶ determine W iteratively
- ▶ Godby-Needs PP model
- ▶ analytic continuation

LiCl

- ▶ determine ϵ on real axis

Diamond

- ▶ without inversion symmetry
- ▶ determine ϵ and invert
- ▶ full frequency (Padé)
- ▶ real frequency integration

BN film

- ▶ 2d truncation



Unit testing – pFUnit

- ▶ test individual units of program
- ▶ provide an exemplary input
- ▶ result of example is known

pFUnit – unit testing in Fortran

- ▶ developed by NASA
- ▶ testcode has pseudo-Fortran format

<http://pfunit.sourceforge.net/>



Example linear solver (1)

```
@test(npes=[1])  
SUBROUTINE test_linear_solver(this)  
  
USE pfunit_mod  
  
! ... other use module + variable definitions  
  
CLASS(MpiTestMethod) :: this
```



Example linear solver (2)

```
! solve the linear problem  $(A + w I) x = b$   
CALL linear_solver(config, A, b_in, w, x, ierr)  
  
! error flag must not be set  
@assertTrue(ierr == 0)  
  
! check if  $(A + w I) x = b$   
CALL A(w, x, b_out)  
  
! evaluate residual error  
residual = norm(b_in - b_out)  
  
! residual error < threshold  
@assertEqual(0.0_dp, residual, config%threshold)
```




The linear solver – multishift

- ▶ shifted linear problem $(A + \omega I)x = b$
- ▶ Krylov subspace based method

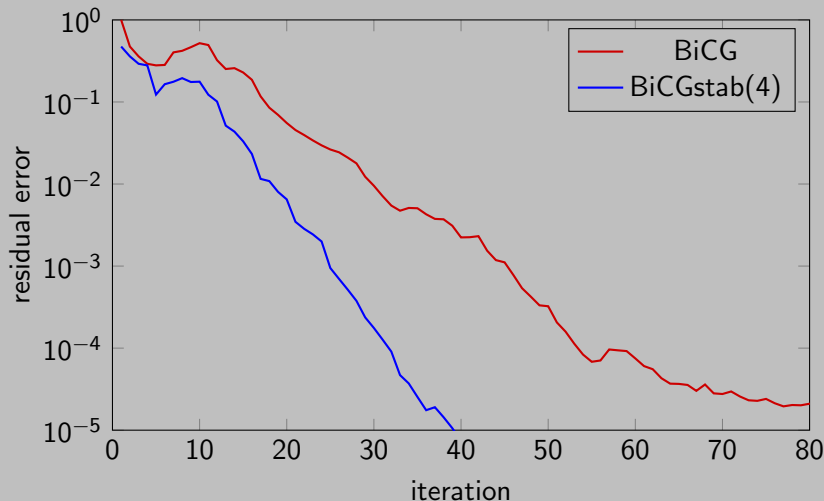
$$\mathcal{K}_n(A, b) = \text{span}\{b, Ab, A^2b, \dots, A^{n-1}b\}$$

- ▶ shifted operator spans the same Krylov subspace
- ▶ A expensive, ωI cheap
- ▶ shifted BiCGstab(I)

Frommer, Computing **70**, 87 (2003)



Performance of the linear solver





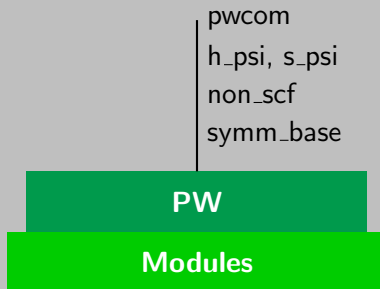
Interacting with QE as a library

fft_custom
mp_images, mp_pools
io_files, iotk
...

Modules

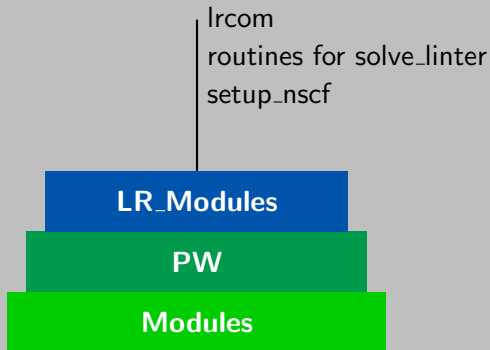


Interacting with QE as a library



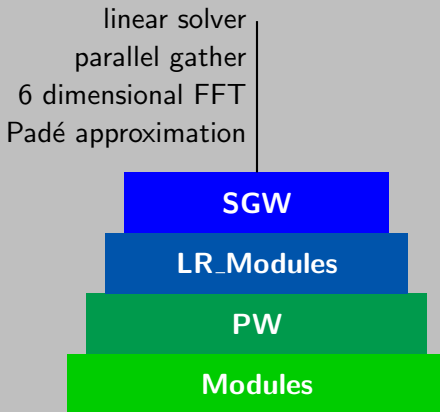


Interacting with QE as a library



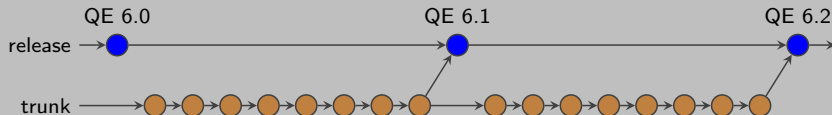


Interacting with QE as a library



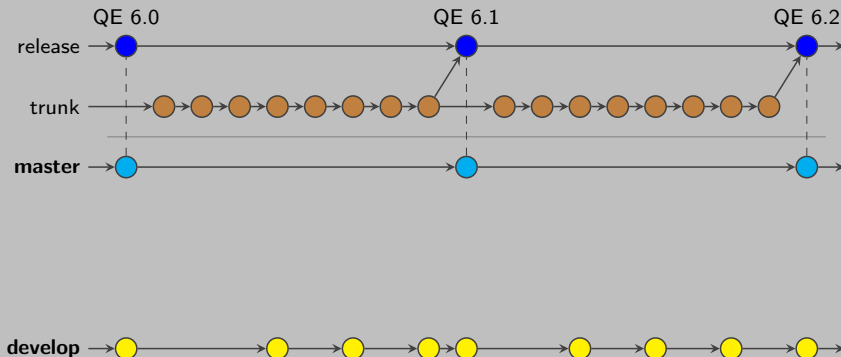


The Gitflow workflow





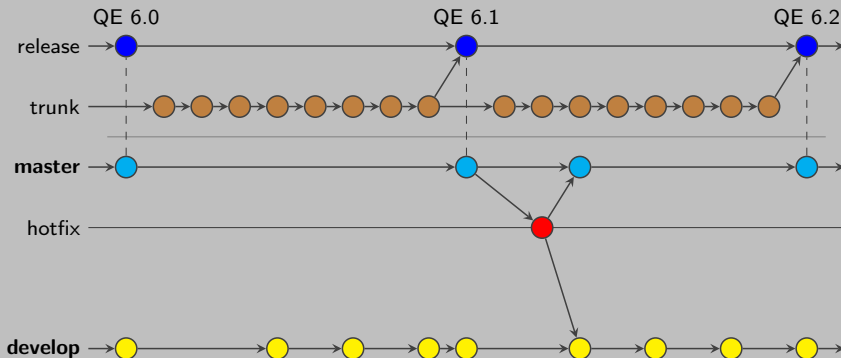
The Gitflow workflow



<http://nvie.com/posts/a-successful-git-branching-model/>



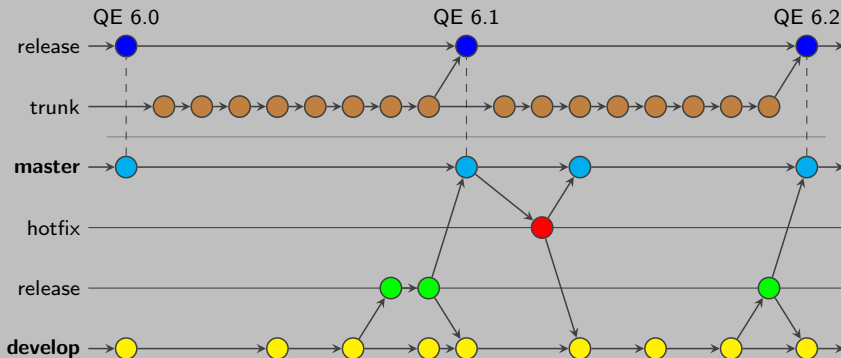
The Gitflow workflow



<http://nvie.com/posts/a-successful-git-branching-model/>



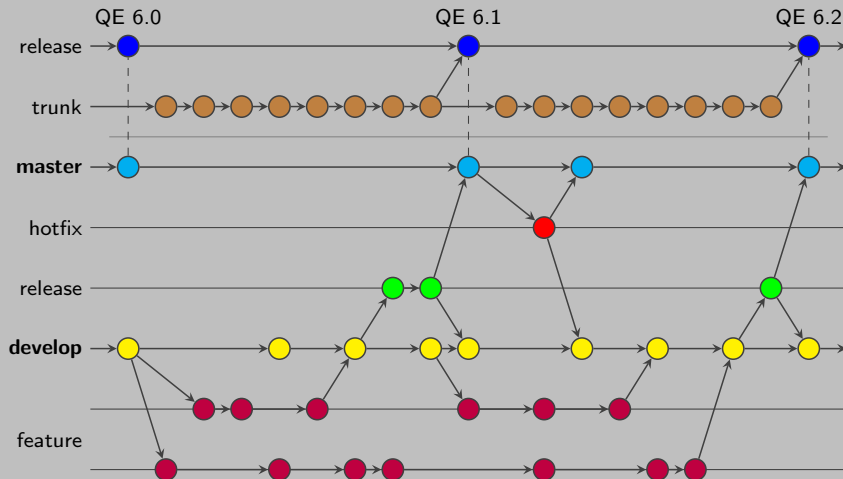
The Gitflow workflow



<http://nvie.com/posts/a-successful-git-branching-model/>



The Gitflow workflow



<http://nvie.com/posts/a-successful-git-branching-model/>



Future plans

- ▶ reduce the amount of global modules
- ▶ improve the user interface
- ▶ release the SGW code
- ▶ integrate SGW in QE



Acknowledgments



The Leverhulme Trust

