Introduction to EPWpy and MatCSSI

Sabyasachi Tiwari
Oden Institute for Computational Engineering and Sciences
UT Austin

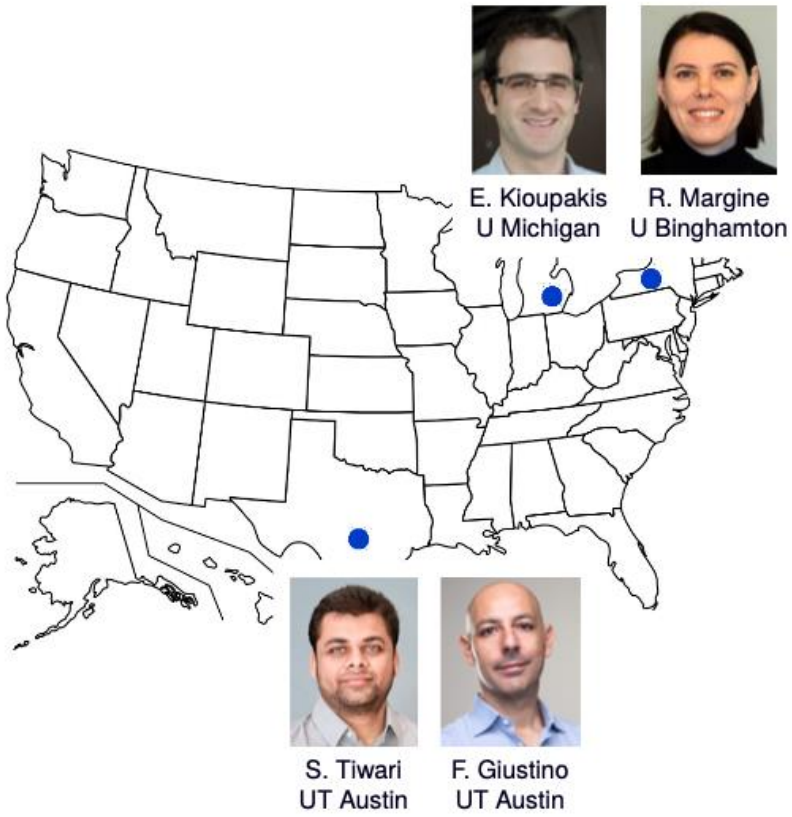
Outline

- Electron-Phonon-Physics with EPW
- Programmable abstraction using EPWpy
- Integrated MATCSSI portal
- Conclusion

Outline

- Electron-Phonon-Physics with EPW
- Programmable abstraction using EPWpy
- Integrated MATCSSI portal
- Conclusion

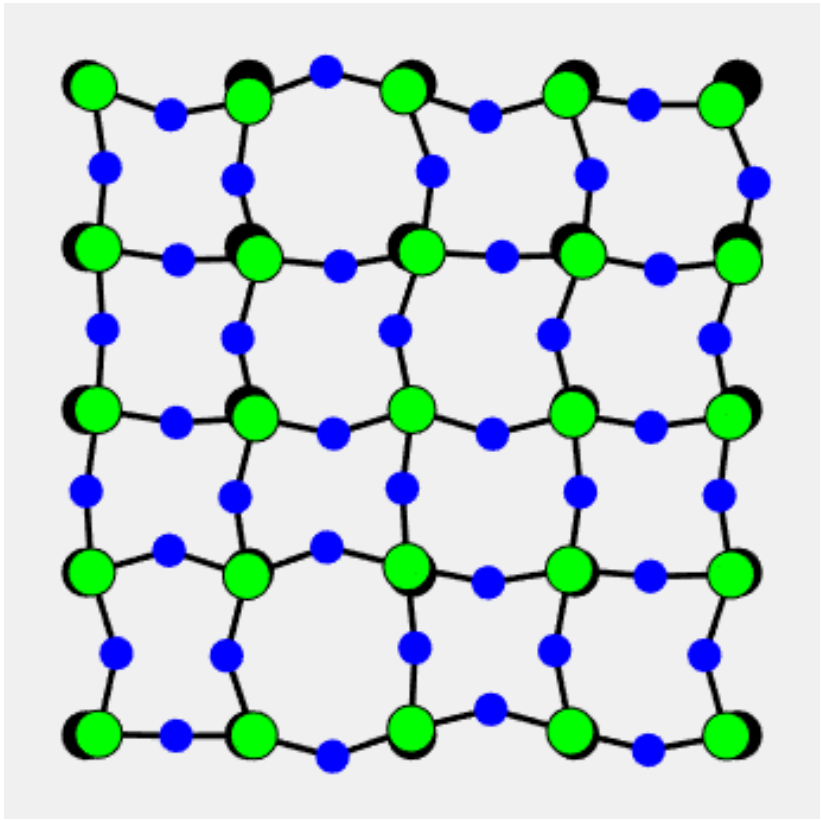
EPW collaboration



Current developers

Kyle Bushick
Fabio Caruso
Jie-Cheng Chen
Zhenbang Dai
Feliciano Giustino
Viet-Anh Ha
Emmanouil Kioupakis
Jon Lafuente-Bartolomé
Tae Yun Kim
Chao Lian
Jae-Mo Lihm
Zhe Liu
Roxana Margine
Hitoshi Mori
Yiming Pan
Samuel Poncé
Danylo Radevych
Young-Woo Son
Sabyasachi Tiwari
Aidan Thorn
Shashi Mishra
Amanda Wang
Wooil Yang
Marios Zacharias
Xiao Zhang

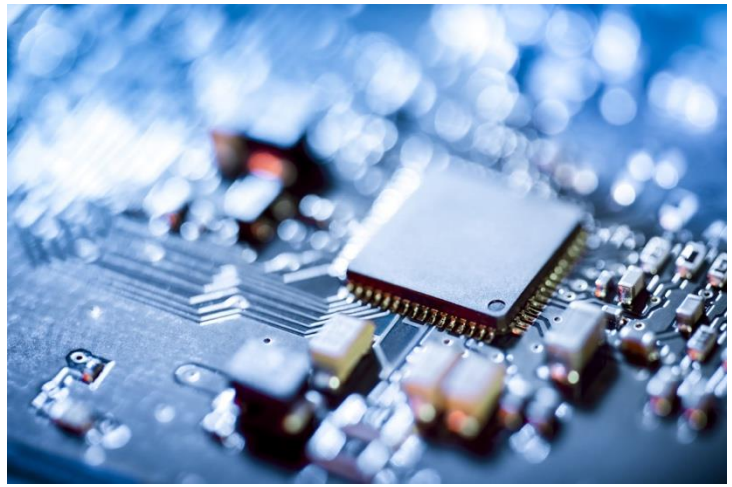
Why Electron-Phonon Physics?



- Lattice vibrations : Phonons
- Electrons/excited particles
  - Interact with lattice
  - E-P coupling

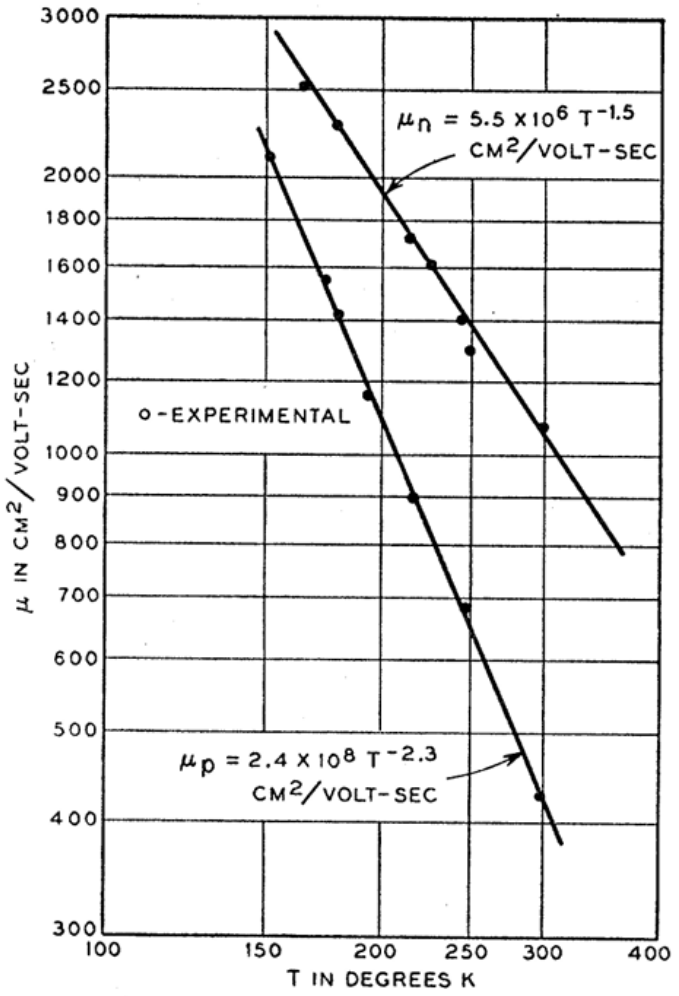
Wendt et.al., Sci. Adv. 10, 1126 (2019)

Why Electron-Phonon Physics?



With more usage (higher heat) they become slow

Why Electron-Phonon Physics?



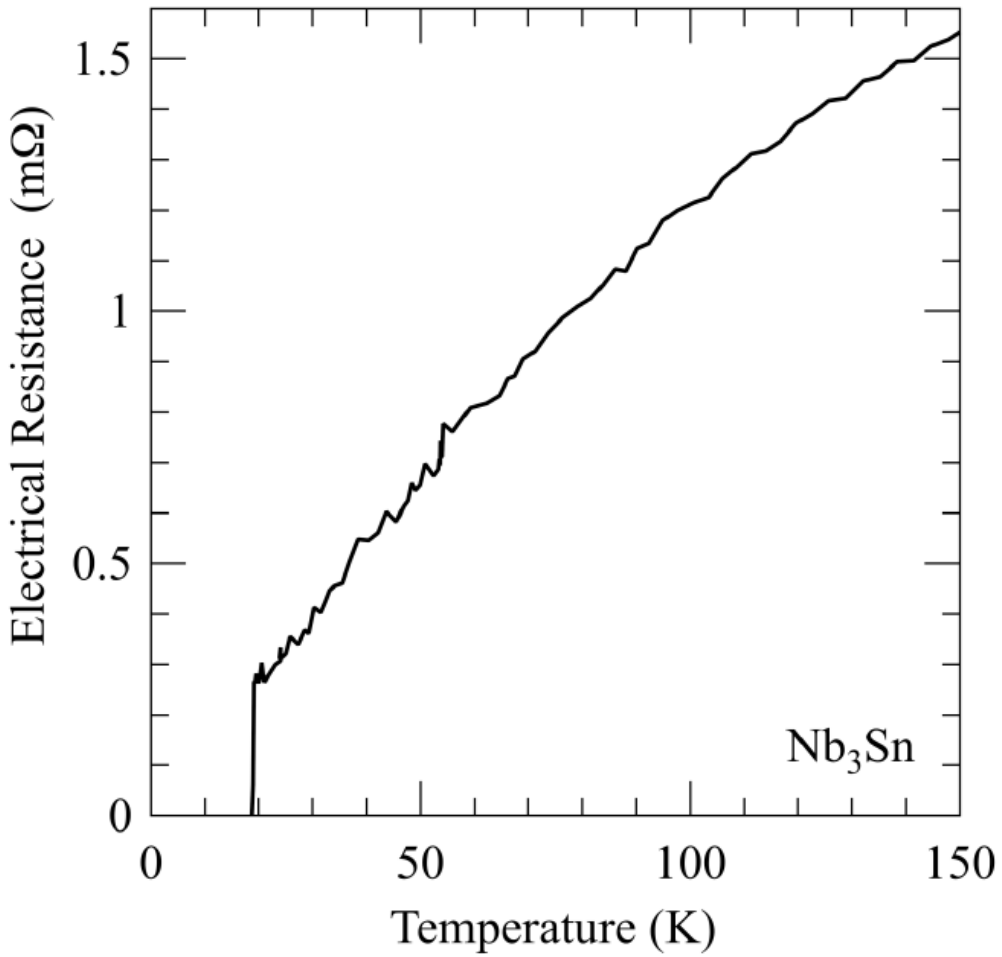
- Transport
  - With temperature mobility goes down

Prince et al., Phys. Rev. B. 93.1204 (1954)

Why Electron-Phonon Physics ($T \approx 0$ K)?

MRI machines use superconductors to generate magnetic field

Why Electron-Phonon Physics (T ≈ 0 K)?



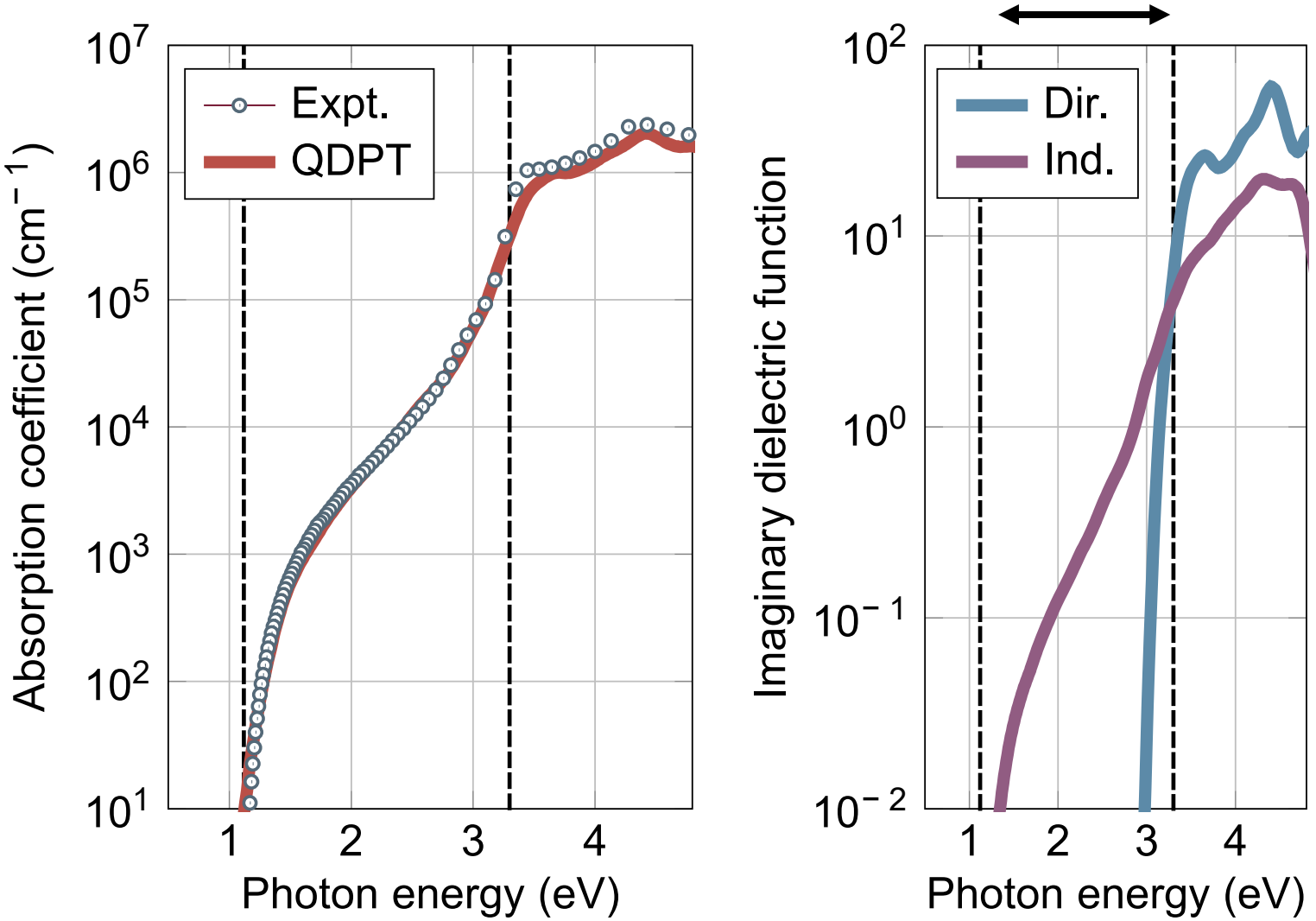
- Superconductivity

Resistance of Nb3Sn and superconducting transition
Akimitsu group, in “Superconductors: New Developments”, 2015

Why Electron-Phonon Physics (Light-Matter interaction)?

Solar farm using silicon photovoltaics which absorb sunlight and convert to electricity

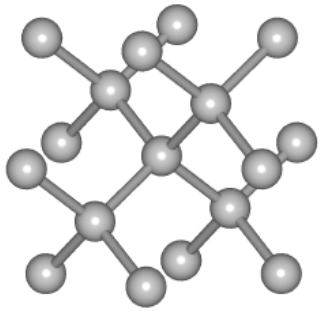
Why Electron-Phonon Physics (Light-Matter interaction)?



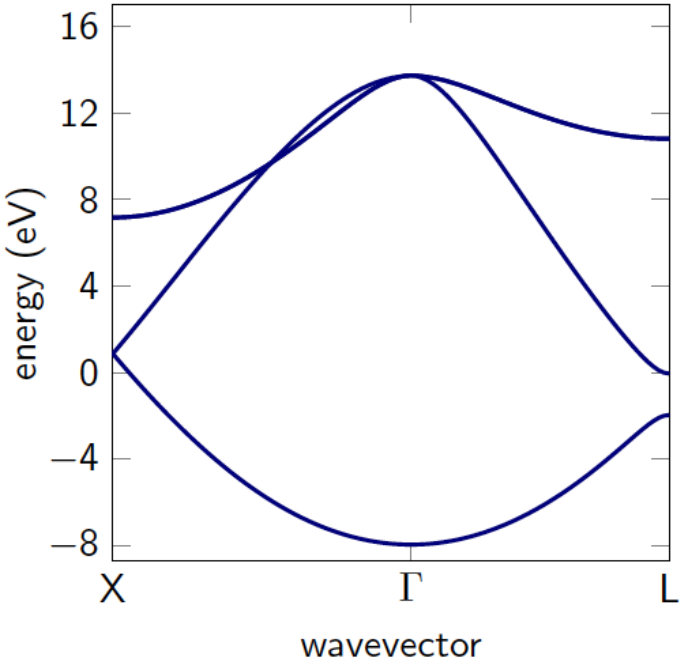
- Visible range: Phonons

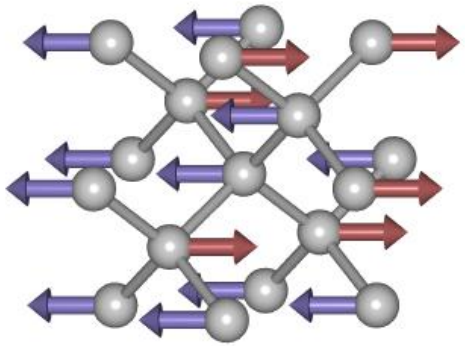
Tiwari et.al., Phys. Rev. B. 109. 195137 (2024)

Where do phonons come from?



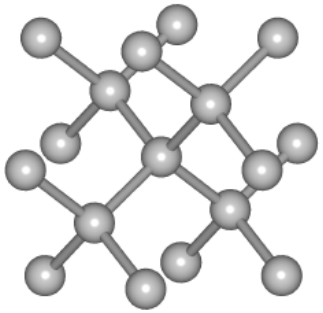
diamond



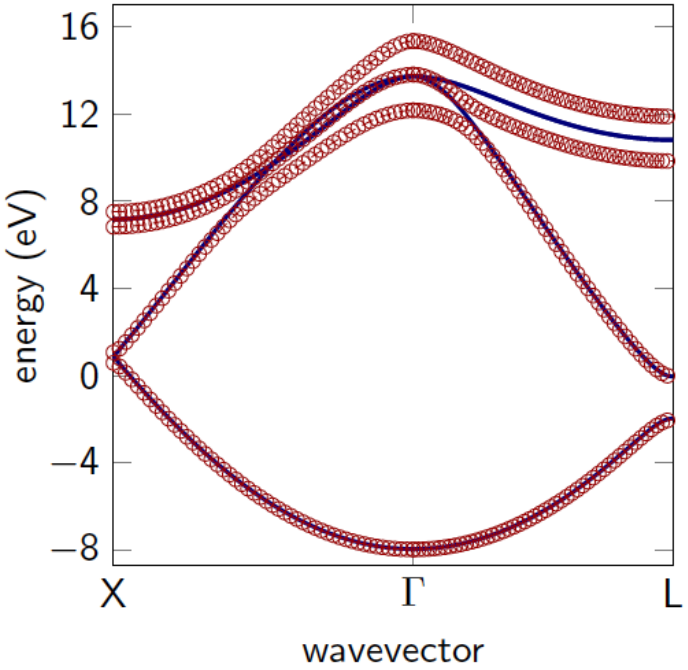


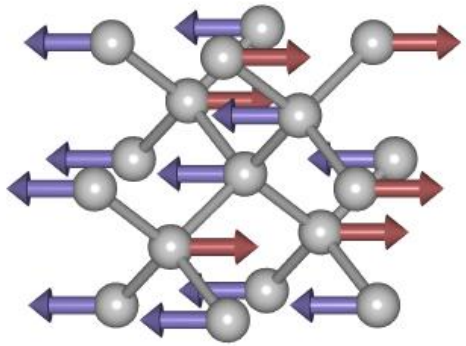
Move atoms along a mode

Where do phonons come from?



diamond

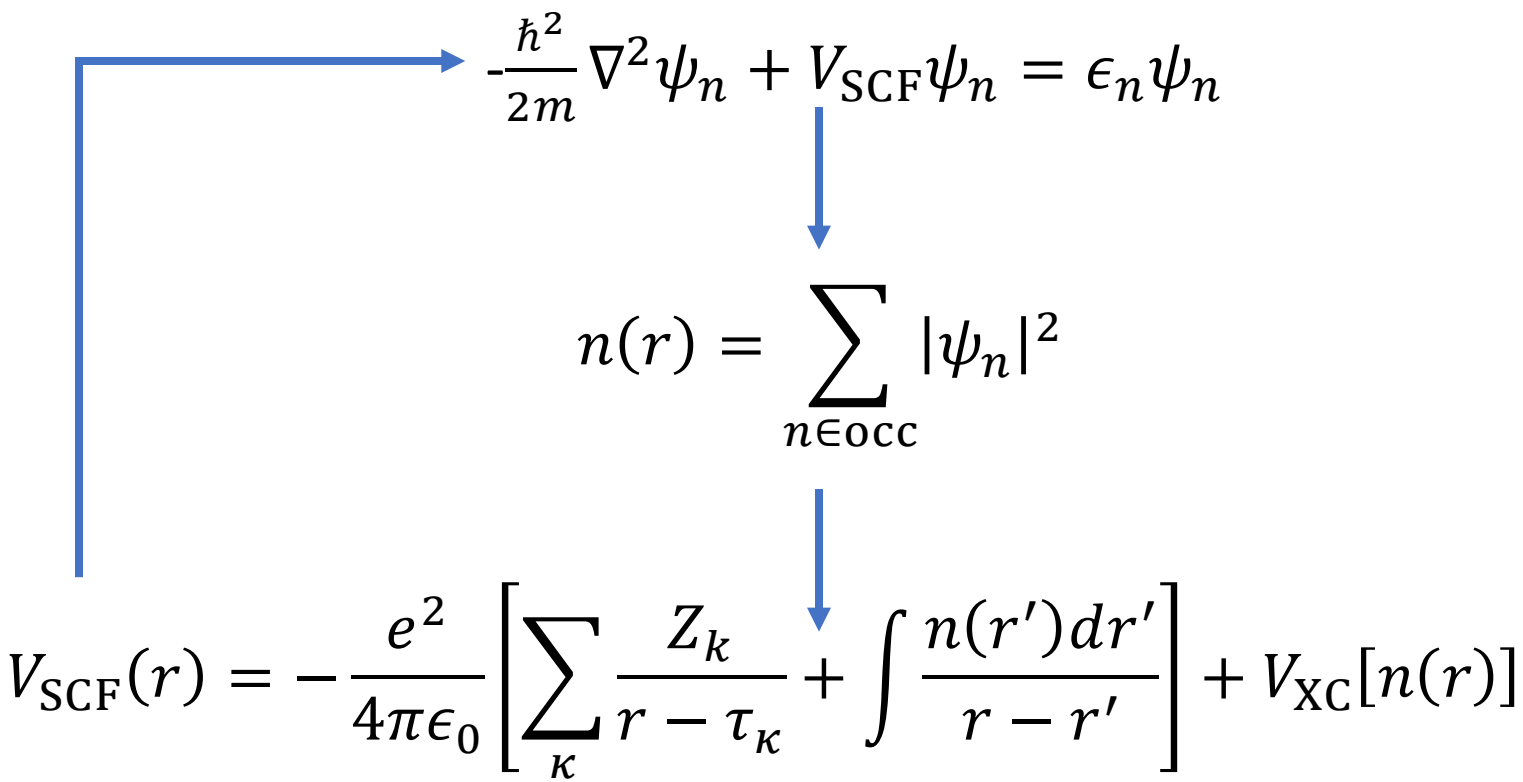




Move atoms along a mode

- Change in band structure

Where do phonons come from?



- V_{SCF}: Depends on atomic positions (τ_κ)

Where do phonons come from?

$$V_{\mathrm{SCF}} = V_{\mathrm{SCF}}(r; \tau_{\kappa_1}, \tau_{\kappa_2}, \tau_{\kappa_3} .. \tau_{\kappa_N})$$

Displace atoms from equilibrium position: $\tau = \tau_0 + \Delta\tau$

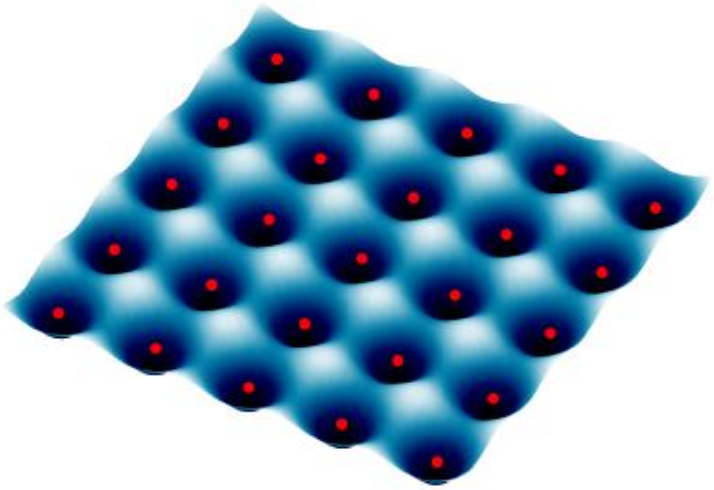
$$V_{\mathrm{SCF}}(\tau_0 + \Delta\tau) = V_{\mathrm{SCF}}(\tau_0) + \frac{\partial V_{\mathrm{SCF}}}{\partial \tau}\Delta\tau + \frac{1}{2}\frac{\partial^2 V_{\mathrm{SCF}}}{\partial \tau^2}\Delta\tau^2 + \ldots\ldots$$

- Perturbation to atomic positions
  - Temperature

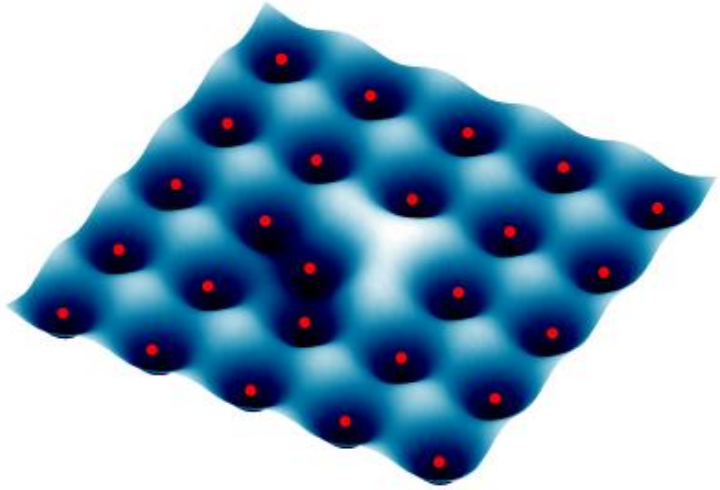
Baroni et.al., Rev. Mod. Phys. 73, 515 (2001)

Where do phonons come from?

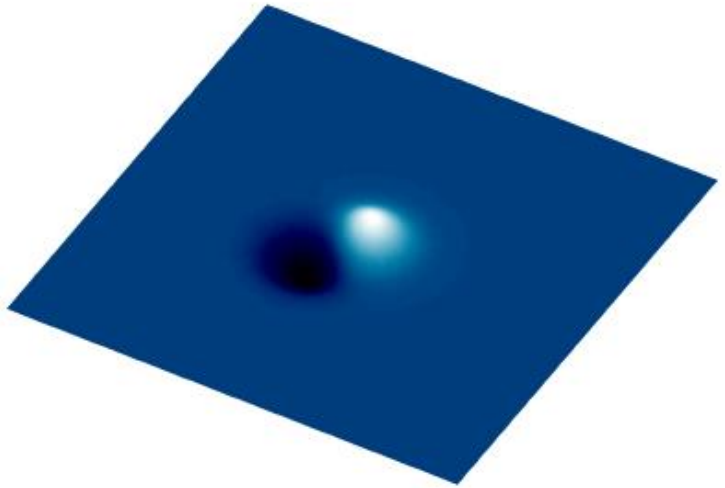
$V_{\mathrm{SCF}}(r;\tau)$



$V_{\mathrm{SCF}}(r;\tau+\Delta\tau)$



$V_{\mathrm{SCF}}(r;\tau)-V_{\mathrm{SCF}}(r;\tau+\Delta\tau)$



Perturbing potential

Use of perturbing potential

Energy $\Delta E = \left\langle \psi_m \left| \frac{\partial V_{\mathrm{SCF}}}{\partial \tau} \Delta\tau \right| \psi_n \right\rangle$ Band renormalization

Wavefunction $\Delta\psi = \sum_{m \neq n} \frac{\left\langle \psi_m \left| \frac{\partial V_{\mathrm{SCF}}}{\partial \tau} \Delta\tau \right| \psi_n \right\rangle}{E_m - E_n} \psi_m$ Optical transitions

Transition Rate $\Gamma_{n \to m} = \frac{2\pi}{\hbar} \left| \left\langle \psi_m \left| \frac{\partial V_{\mathrm{SCF}}}{\partial \tau} \Delta\tau \right| \psi_n \right\rangle \right|^2 \delta(E_m - E_n)$ Transport

- Calculation of perturbation potential is key to calculating many properties

Electron-Phonon matrix element

$$\left\langle \psi_m \left| \frac{\partial V_{\mathrm{SCF}}}{\partial \tau_{\kappa\alpha p}} \right| \psi_n \right\rangle \rightarrow \left\langle u_{m\mathbf{k}+\mathbf{q}} \left| \Delta_{\mathbf{q}\nu} v_{\mathrm{SCF}} \right| u_{n\mathbf{k}} \right\rangle_{\mathrm{uc}} = g^{\nu}_{mn}(\mathbf{k},\mathbf{q})$$

Supercell

Unit-cell

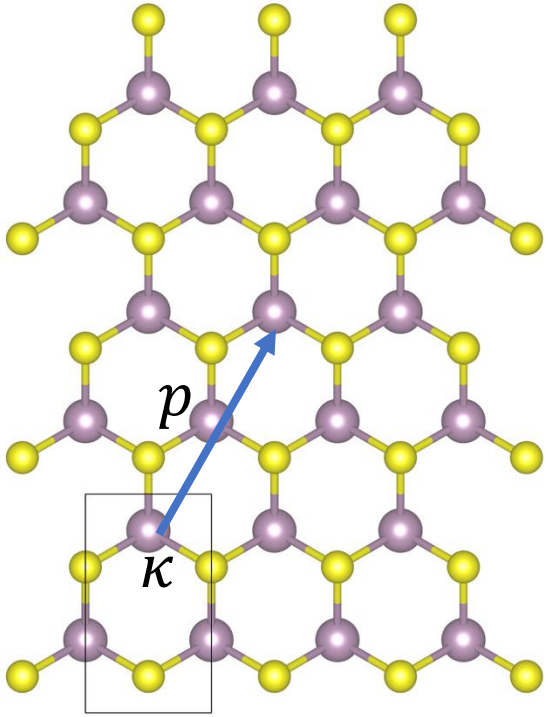
Potential variation from ion displacement

$$\Delta_{\mathbf{q}\nu} v_{\mathrm{SCF}} = \sum_{\kappa\alpha p} e^{-i\mathbf{q}.(\mathrm{r}-\mathbf{R}_p)} \sqrt{\frac{\hbar}{2M_\kappa \omega_{\mathbf{q}\nu}}}\, e_{\kappa\alpha,\nu}(\mathbf{q}) \frac{\partial V_{\mathrm{SCF}}(\mathbf{r})}{\partial \tau_{\kappa\alpha p}}$$

Incommensurate modulation

Zero-point amplitude

Phonon polarization



- Go from supercell to unit-cell

Giustino et.al., Rev. Mod. Phys. 89, 015013 (2017)

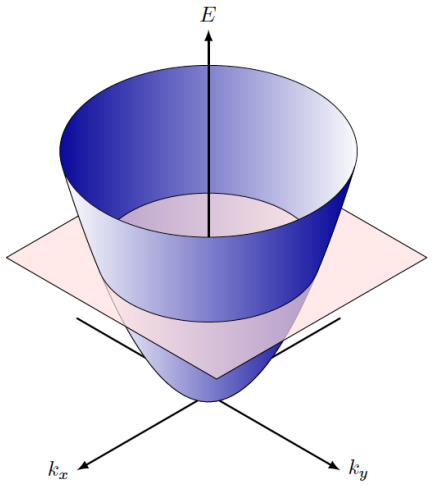
Electron-Phonon matrix element (practical calculation)

$$g^{\nu}_{mn}(\mathbf{k},\mathbf{q}) = \left\langle u_{m\mathbf{k}+\mathbf{q}} \middle| \Delta_{\mathbf{q}\nu} v_{\mathrm{SCF}} \middle| u_{n\mathbf{k}} \right\rangle$$

Giustino et.al., Rev. Mod. Phys. 89, 015013 (2017)

Electron-Phonon matrix element (practical calculation)

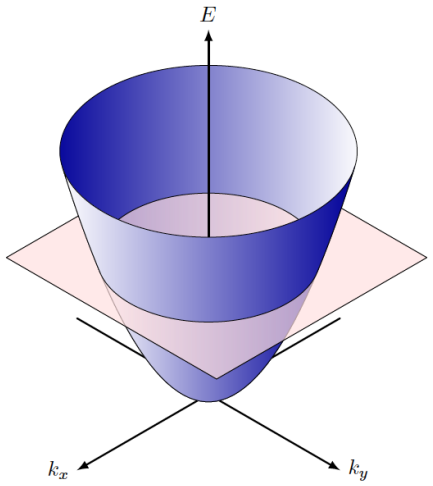
$$g^{\nu}_{mn}(\mathbf{k},\mathbf{q}) = \langle u_{m\mathbf{k}+\mathbf{q}} | \Delta_{\mathbf{q}\nu} v_{\mathrm{SCF}} | u_{n\mathbf{k}} \rangle$$

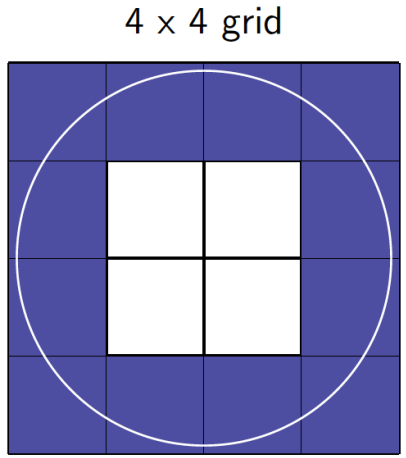


Giustino et.al., Rev. Mod. Phys. 89, 015013 (2017)

Electron-Phonon matrix element (practical calculation)

$$g^{\nu}_{mn}(\mathbf{k},\mathbf{q}) = \langle u_{m\mathbf{k}+\mathbf{q}} | \Delta_{\mathbf{q}\nu} v_{\mathrm{SCF}} | u_{n\mathbf{k}} \rangle$$

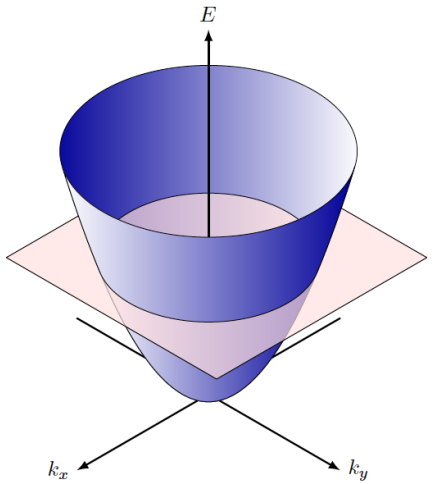




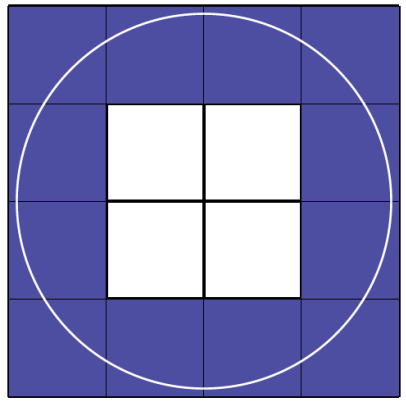
Giustino et.al., Rev. Mod. Phys. 89, 015013 (2017)

Electron-Phonon matrix element (practical calculation)

$$g^{\nu}_{mn}(\mathbf{k},\mathbf{q}) = \langle u_{m\mathbf{k}+\mathbf{q}} | \Delta_{\mathbf{q}\nu} v_{\mathrm{SCF}} | u_{n\mathbf{k}} \rangle$$

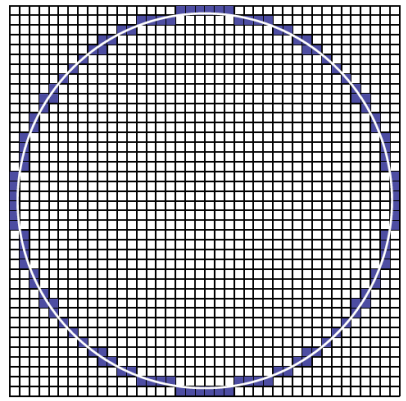


4 x 4 grid



Coarse BZ sampling

40 x 40 grid

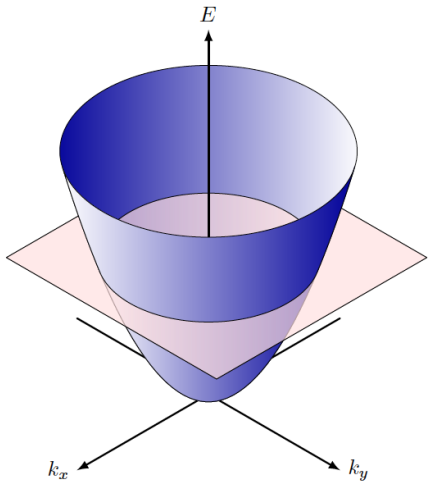


Fine BZ sampling

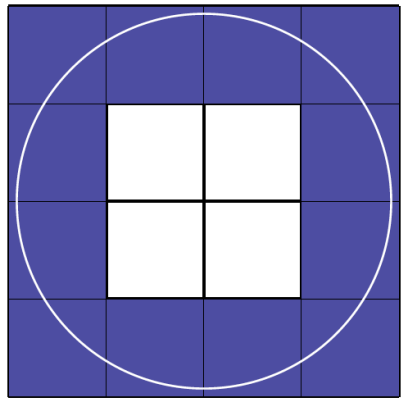
Giustino et.al., Rev. Mod. Phys. 89, 015013 (2017)

Electron-Phonon matrix element (practical calculation)

$$g^{\nu}_{mn}(\mathbf{k},\mathbf{q}) = \langle u_{m\mathbf{k}+\mathbf{q}} | \Delta_{\mathbf{q}\nu} v_{\mathrm{SCF}} | u_{n\mathbf{k}} \rangle$$

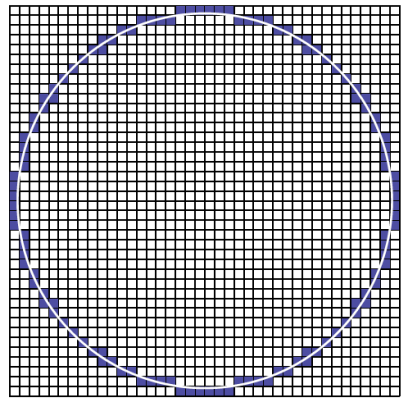


4 x 4 grid



Coarse BZ sampling

40 x 40 grid



Fine BZ sampling

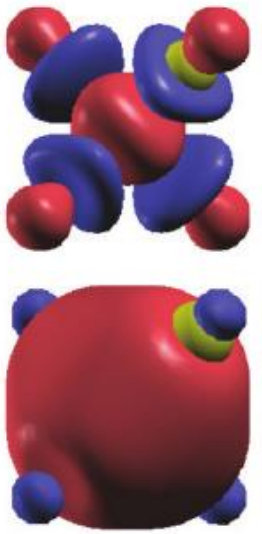
- Dense **k, q** grids needed for calculating observables
- Wavefunctions: NSCF calculations (DFT)
  - Cheap for unit-cells
- Displacements/derivatives V_{SCF}: Phonon calculations
  - Expensive
  - Dense grids not possible

Giustino et.al., Rev. Mod. Phys. 89, 015013 (2017)

Wannier functions

$$w_m(r;\mathbf{R}_p) = \frac{1}{N_p}\sum_{nk} \exp(i\mathbf{k}.\mathbf{R}_p) U_{nm}^{\mathbf{k}} \, |\psi_{n\mathbf{k}}\rangle$$

$$|\psi_{n\mathbf{k}}\rangle = \sum_{m\mathbf{R}_p} \exp(-i\mathbf{k}.\mathbf{R}_p) U_{mn}^{\mathbf{k}*} \, w_m(r;\mathbf{R}_p)$$



- Wavefunctions can be localized using Wannier representation
- Interpolated to arbitrary wavevectors using Fourier transform
  - $U_{mn}^{\mathbf{k}}$: Obtained by diagonalizing Wannier Hamiltonian

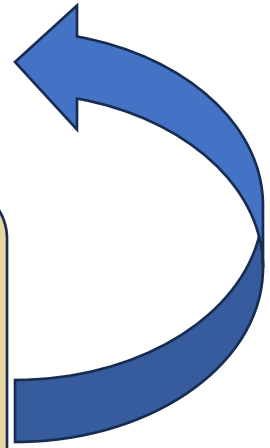
Marzari et.al., Rev. Mod. Phys. 84, 1419 (2012)

Wannier interpolation

$$g_{mn}^{\nu}(\mathbf{k},\mathbf{q}) = \langle u_{m\mathbf{k}+\mathbf{q}} | \Delta_{\mathbf{q}\nu} v_{\mathrm{SCF}} | u_{n\mathbf{k}} \rangle$$

$$w_m(r;\mathbf{R}_p) = \frac{1}{N_p} \sum_{nk} \exp(i\mathbf{k}.\mathbf{R}_p) U_{nm}^{\mathbf{k}} \, |\psi_{n\mathbf{k}}\rangle$$

$$|\psi_{n\mathbf{k}}\rangle = \textstyle\sum_{m\mathbf{R}_p} \exp(-i\mathbf{k}.\mathbf{R}_p) U_{mn}^{\mathbf{k}*} \, w_m(r;\mathbf{R}_p)$$



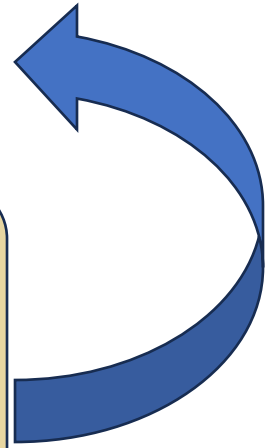
Giustino et.al., Phys. Rev. B. 76, 165108 (2007)

Wannier interpolation

$$g^{\nu}_{mn}(\mathbf{k},\mathbf{q}) = \langle u_{m\mathbf{k}+\mathbf{q}} | \Delta_{\mathbf{q}\nu} v_{\mathrm{SCF}} | u_{n\mathbf{k}} \rangle$$

$$w_m(r;\mathbf{R}_p) = \frac{1}{N_p}\sum_{nk} \exp(i\mathbf{k}.\mathbf{R}_p) U^{\mathbf{k}}_{nm} \, |\psi_{n\mathbf{k}}\rangle$$

$$|\psi_{n\mathbf{k}}\rangle = \textstyle\sum_{m\mathbf{R}_p} \exp(-i\mathbf{k}.\mathbf{R}_p) U^{\mathbf{k}*}_{mn} \, w_m(r;\mathbf{R}_p)$$

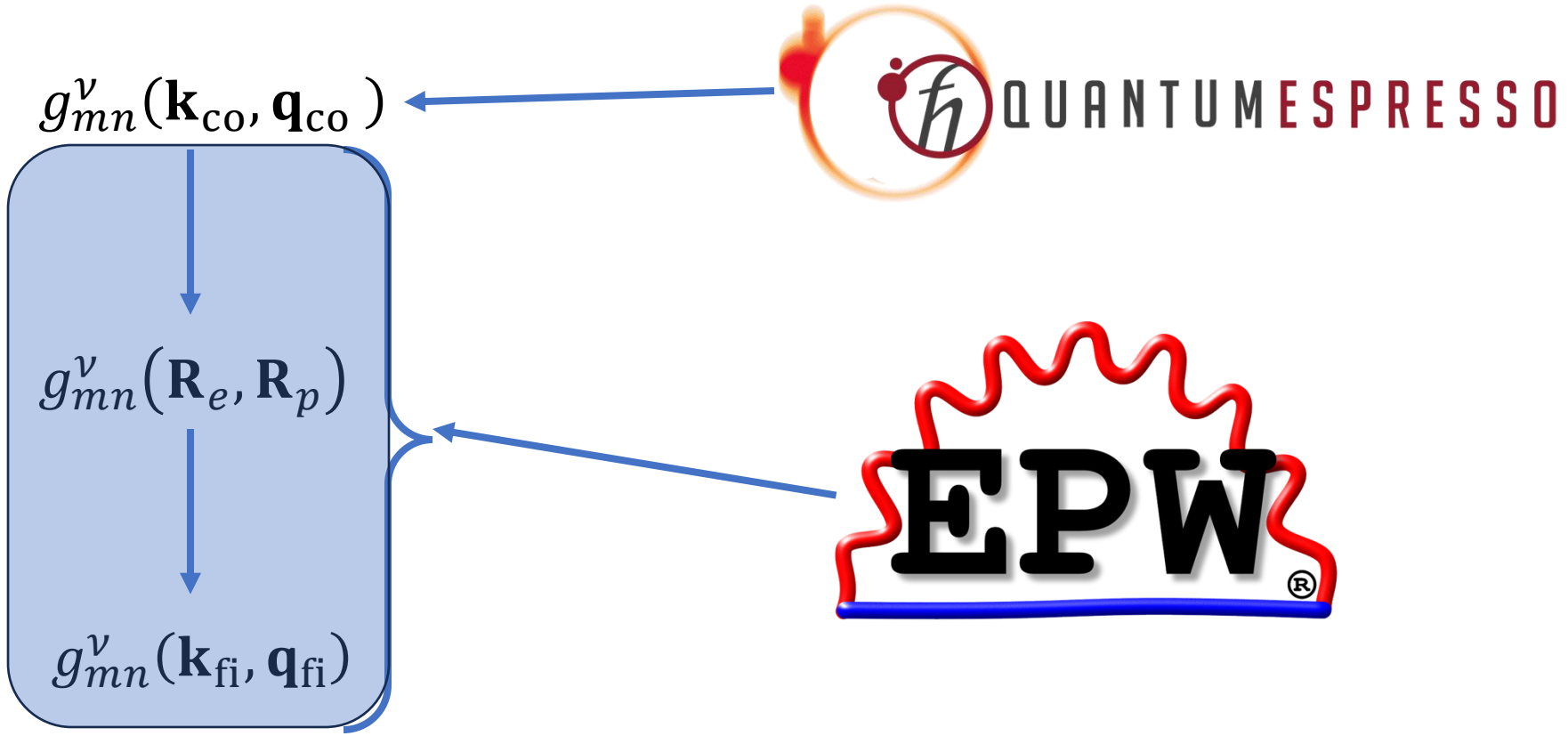


$$g^{\nu}_{mn}(\mathbf{k}_{\mathrm{co}},\mathbf{q}_{\mathrm{co}}) \rightarrow g^{\nu}_{mn}(\mathbf{R}_e,\mathbf{R}_p) \rightarrow g^{\nu}_{mn}(\mathbf{k}_{\mathrm{fi}},\mathbf{q}_{\mathrm{fi}})$$

- A method for reliably interpolating e-p matrix element

Giustino et.al., Phys. Rev. B. 76, 165108 (2007)

Electron-Phonon-Wannier method (EPW)



- QE: $\Delta_{\mathbf{q}\nu} v_{\mathrm{SCF}}; \Delta\tau$
- EPW: Wannierization and interpolation

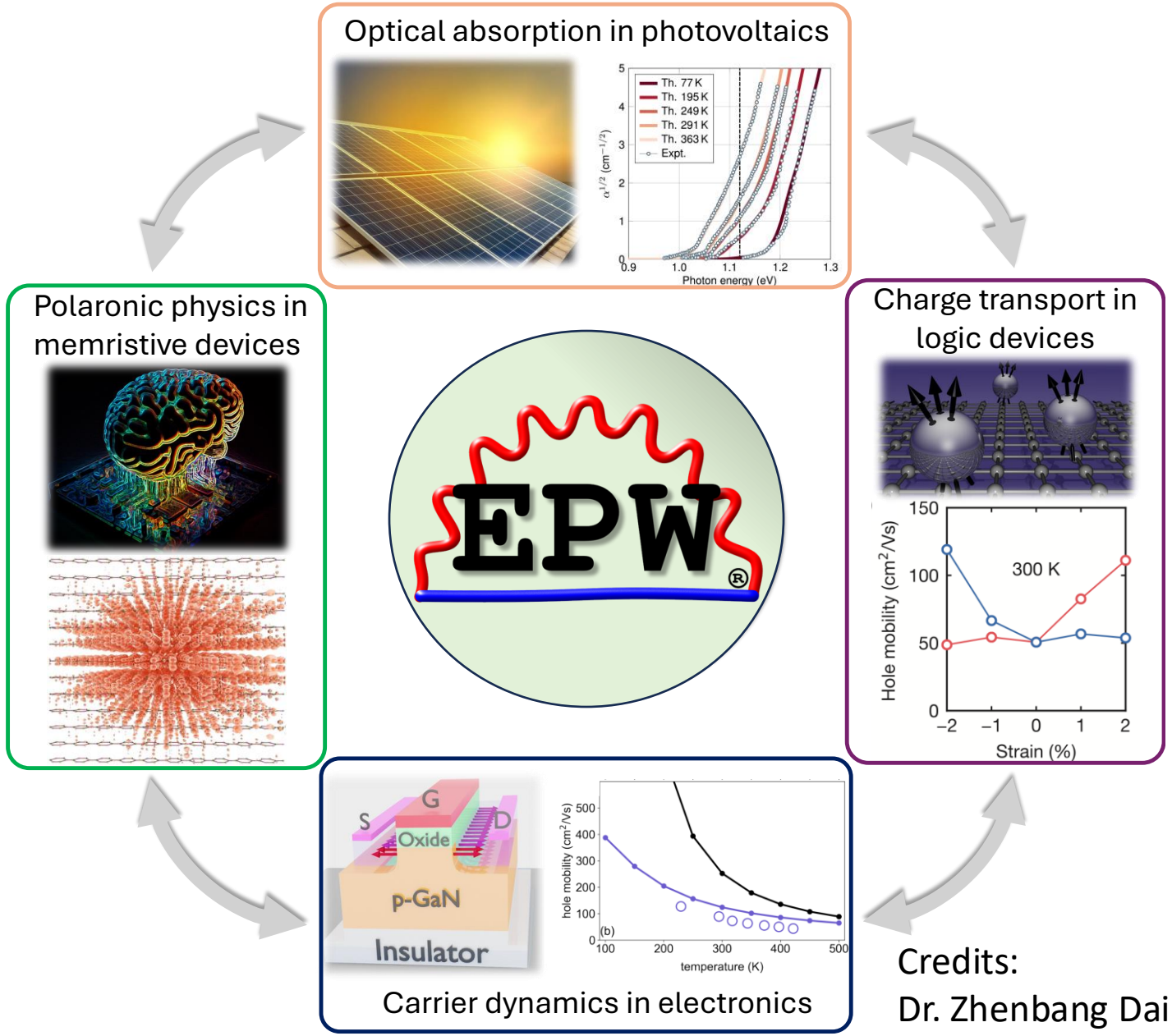
Ponce et.al., Comput. Phys. Comm, 209, 116-133 (2016)
Lee, … Tiwari et.al., npj comput. mater., 9, 156 (2023)

EPW

- Electron-Phonon Wannier (EPW) code
  - A sub-package within the Quantum Espresso DFT package
  - Interpolates e-p matrix elements obtained from DFPT (QE)
  - Allows calculations on very fine **k** and **q** grids
    - **Transport (IBTE)**
    - **Polarons**
    - Superconductivity
    - Temperature dependent Bandgap
    - Optical absorption

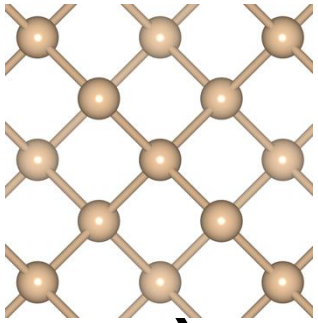
Current computational materials design landscape (EPW)

- Computational materials design impacts all walks of life today
- These calculations require very complex workflows and **user intervention**
- High-throughput study of large materials databases is **currently out of reach**
- Challenge is two-fold
  - Streamlining calculations
  - Analysis



Credits:
Dr. Zhenbang Dai

A typical workflow for EPW



$\mathbf{q}$-grids

SCF

PH

NSCF

$e_{\mathbf{q}\nu}$
$\partial_{\mathbf{q}\nu}V$

$\epsilon_{n\mathbf{k}}$
$\psi_{n\mathbf{k}}$

WANNIER90

$U_{n\mathbf{k}}$

EPW
(Wannier)

$U_{n\mathbf{k}}$
$g_{mn\nu}(\mathbf{R}_e, \mathbf{R}_p)$

EPW
(interpolation)

Wavefunction cutoff
Pseudopotentials
Bands

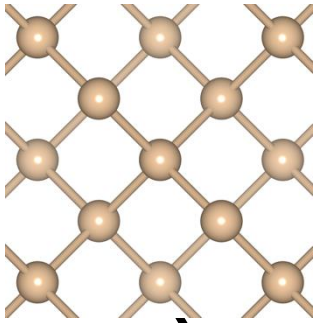
Bands
$\mathbf{k}$-grid

Wannier functions
Check for
correctness

Physical property
related inputs
Optical range
Fermi-surface thickness

Lee et. al., NPJ. Comput. Mat 9, 156 (2023)

A typical workflow for EPW (more complex)



$\mathbf{q}$-grids

SCF

PH

NSCF

GW

BSE

WANNIER90

$e_{\mathbf{q}\nu}$
$\partial_{\mathbf{q}\nu}V$

$U_{n\mathbf{k}}$

$\epsilon_{n\mathbf{k}}$
$\psi_{n\mathbf{k}}$

$\epsilon_{n\mathbf{k}}$

$A_{n\mathbf{k}}$

EPW
(Wannier)

$U_{n\mathbf{k}}$
$g_{mn\nu}(\mathbf{R}_e, \mathbf{R}_p)$

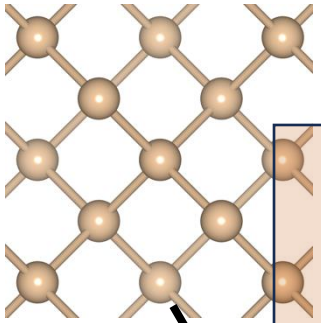
EPW
(interpolation)

Wavefunction cutoff
Pseudopotentials
Bands

Since we deal with $|N \pm 1\rangle/|N + 2\rangle$ state

Dai et. al., Phys. Rev. Lett 32, 036902 (2024)

A typical workflow for EPW



SCF

PH

NSCF

$e_{\mathbf{q}\nu}$
$\partial_{\mathbf{q}\nu}V$

$\epsilon_{n\mathbf{k}}$
$\psi_{n\mathbf{k}}$

WANNIER90

W90

$U_{n\mathbf{k}}$

EPW
(Wannier)

$U_{n\mathbf{k}}$
$g_{mn\nu}(\mathbf{R}_e, \mathbf{R}_p)$

EPW
(interpolation)

EPW®

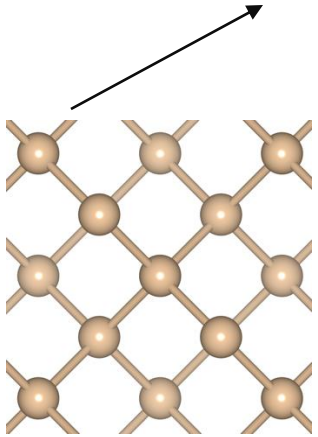
QUANTUMESPRESSO

- Multiple codes communicating with each-other
- Data transfer and user intervention
- There is a need for a unified method to streamline calculations

Outline

- Electron-Phonon-Physics with EPW
- Programmable abstraction using EPWpy
- Integrated MATCSSI portal
- Conclusion

Functional abstraction



```
silicon=EPWpy.EPWpy({'prefix':'si',
                     'calculation':'\'scf\'',
                     'structure_mp':"mp-149",
                     'ecutwfc':'40',
                     'celldm(1)':'10.262',
                     'pseudo_auto':True,
                    },
                    env='mpirun')
```

SCF → Object{material attributes, scf attributes}

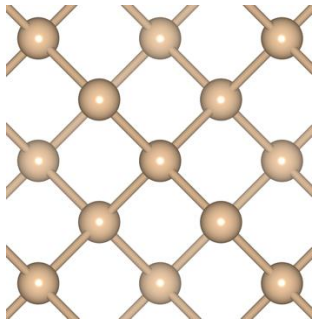
```
silicon.scf(name = 'scf',
            kpoints = {'kpoints':[[6,6,6]]})
```
→ Method to define inputs

```
silicon.prepare('scf')
```
→ Method to prepare the calculation

```
silicon.run(4,'scf')
```
→ Method to run the calculation

- Material: **Object**
- Calculations: **Methods**
- Each calculation can be seen as a **functional block**
- These blocks **can take user input** else **automated**

Functional abstraction



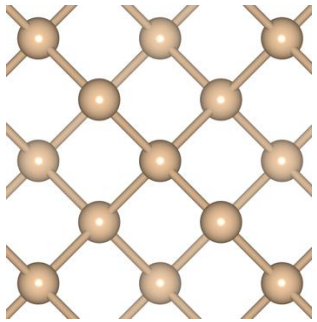
```
silicon.scf(name = 'scf',
            kpoints = {'kpoints':[[6,6,6]]})
silicon.prepare('scf')
silicon.run(4,'scf')
```

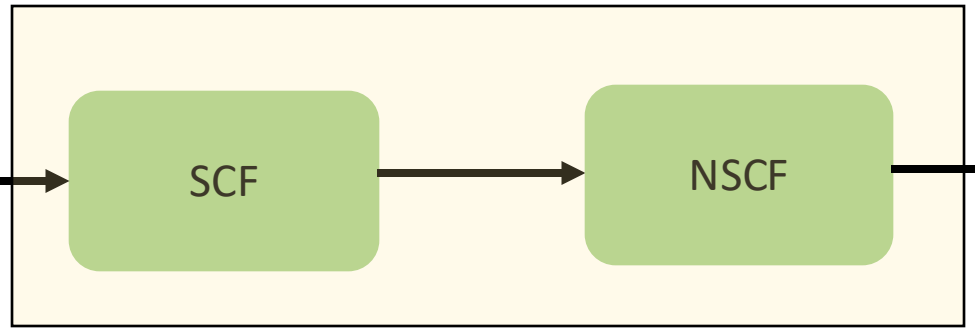
SCF → NSCF → Object{material attributes, scf_attributes, nscf_attributes}

```
silicon.nscf(system = {'nbnd':8},
             kpoints = {'grid':[4,4,4],
                        'kpoints_type': 'crystal'})
silicon.prepare('nscf')
silicon.run(4,'nscf')
```

- Block to block connection **through position**
- Data transfer is handled accordingly

Functional abstraction





Object{material attributes,
scf attributes,
nscf attributes}

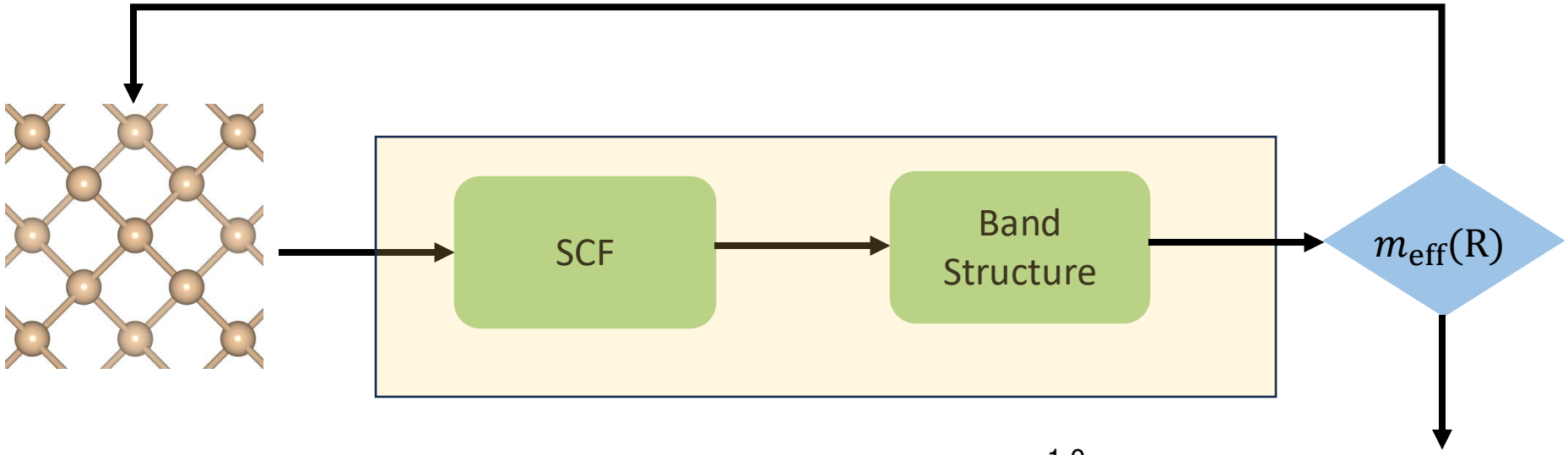
```
silicon.scf(name = 'scf',
            kpoints = {'kpoints':[[6,6,6]]})
silicon.prepare('scf')
silicon.run(4,'scf')

silicon.nscf(system = {'nbnd':8},
             kpoints = {'grid':[4,4,4],
                        'kpoints_type': 'crystal'})
silicon.prepare('nscf')
silicon.run(4,'nscf')
```

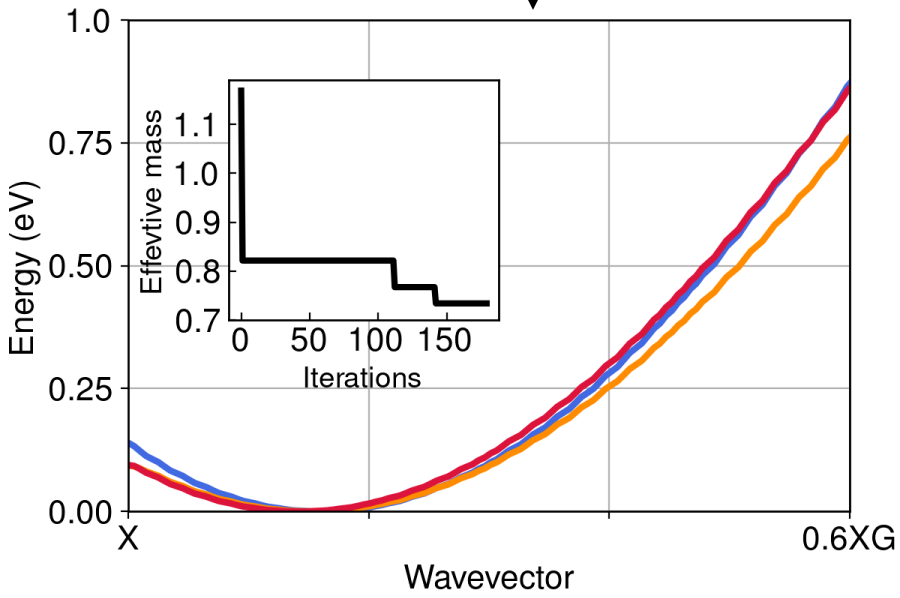
- Blocks combined to give single abstract function

Optimization of these abstract functions

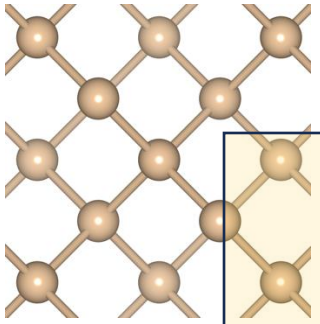
$$\mathbf{R}_{\text{new}} = \mathbf{R}_{\text{old}} + \epsilon$$

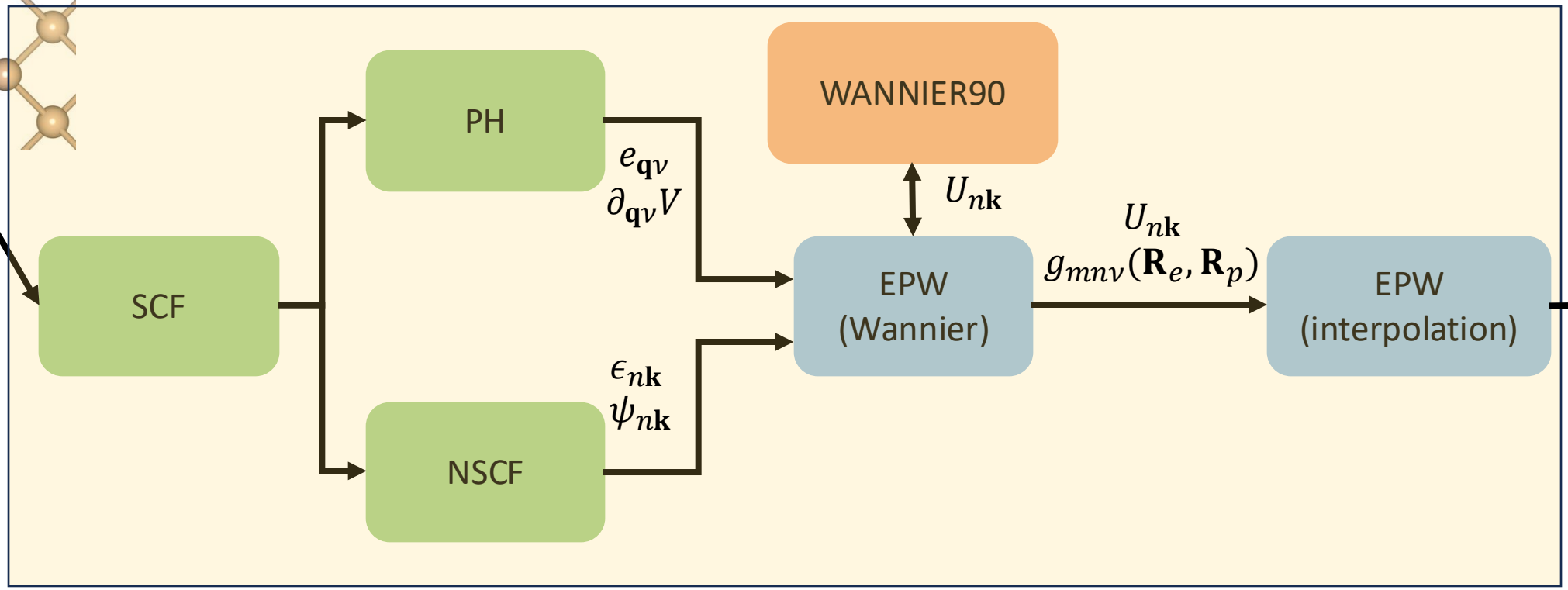


- Entire physical calculation can be encapsulated
- These abstract functions can be optimized using top level optimizers



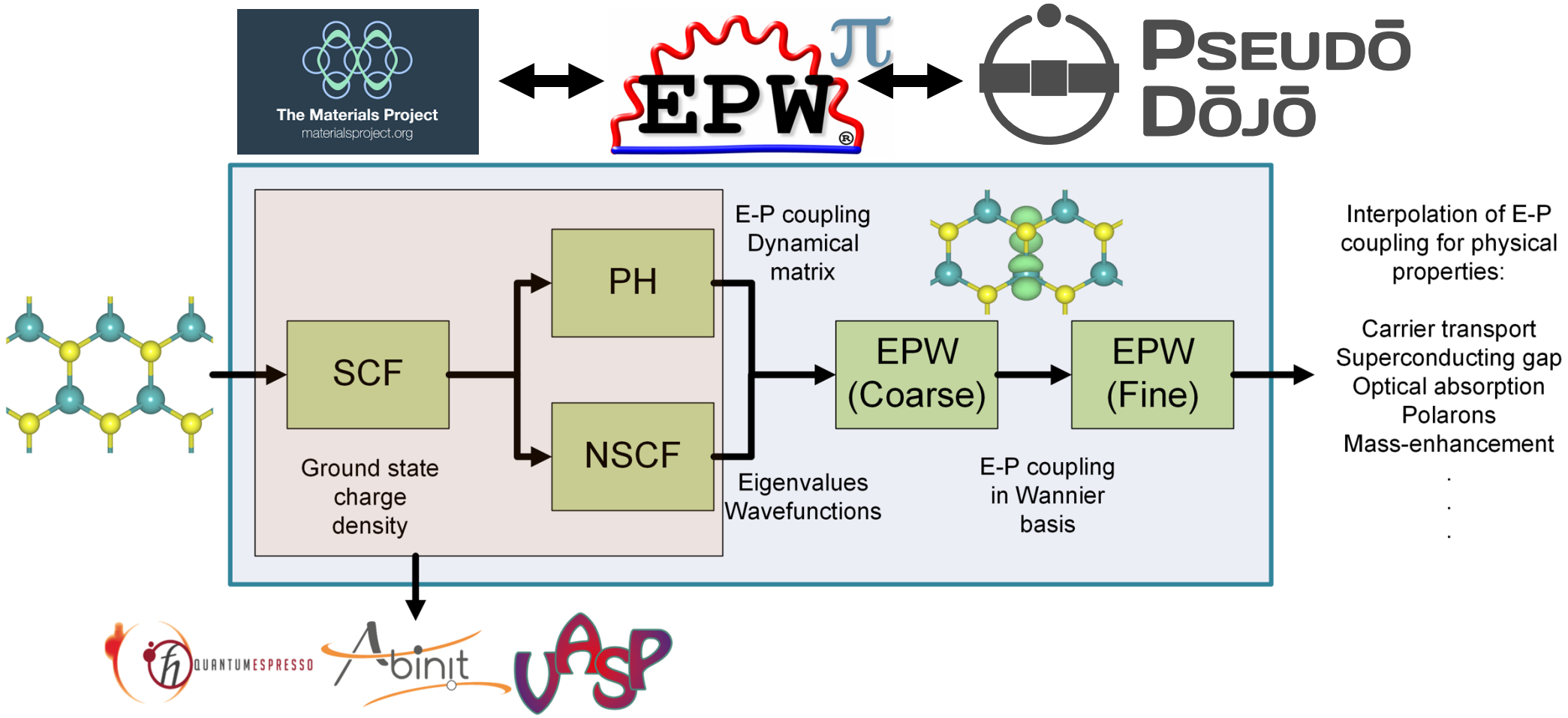
Functional abstraction





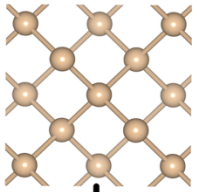
- Entire physical calculation can be encapsulated

EPWpy



- EPWpy is a python program allowing such object-oriented abstractions
- Currently, interface with BGW, partial interface with VASP and Abinit
- Comes with interface with Materials project and PseudoDojo

A typical EPWpy script



```
silicon=EPWpy.EPWpy({'prefix':'si',
                     'calculation':'\'scf\'',
                     'structure_mp':"mp-149",
                     'ecutwfc':'40',
                     'celldm(1)':'10.262',
                     'pseudo_auto':True,
                    },
                    env='mpirun')
```

The Materials Project
materialsproject.org

PSEUDŌ DŌJŌ

SCF

```
silicon.scf(name = 'scf',
            kpoints = {'kpoints':[[6,6,6]]})
silicon.prepare('scf')
silicon.run(4,'scf')
```

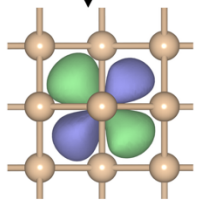
NSCF

```
silicon.nscf(system = {'nbnd':8},
             kpoints = {'grid':[4,4,4],
                        'kpoints_type': 'crystal'})
silicon.prepare('nscf')
silicon.run(4,'nscf')
```

PH

```
silicon.ph(phonons={'nq1':2,
                    'nq2':2,
                    'nq3':2}
           )
silicon.prepare('ph')
silicon.run(4,'ph')
```

EPW (Coarse)



```
silicon.epw(epwin={'wdata':['guiding_centres = .true.',
                            'dis_num_iter = 500',
                            'num_print_cycles  = 10',
                            'dis_mix_ratio = 1',
                            'use_ws_distance = T'],
                   'proj':['\'Si : sp3\''],
                   'band_plot':'.true.',
                   'etf_mem':1,
                   'wannierize':'.true.',
                   'elph':'.true.',
                   'num_iter':500,
                   'prtgkk':'.true.',
                  },
            name='epw1')
silicon.prepare(type_run='epw1')
silicon.run(4,type_run='epw1')
```

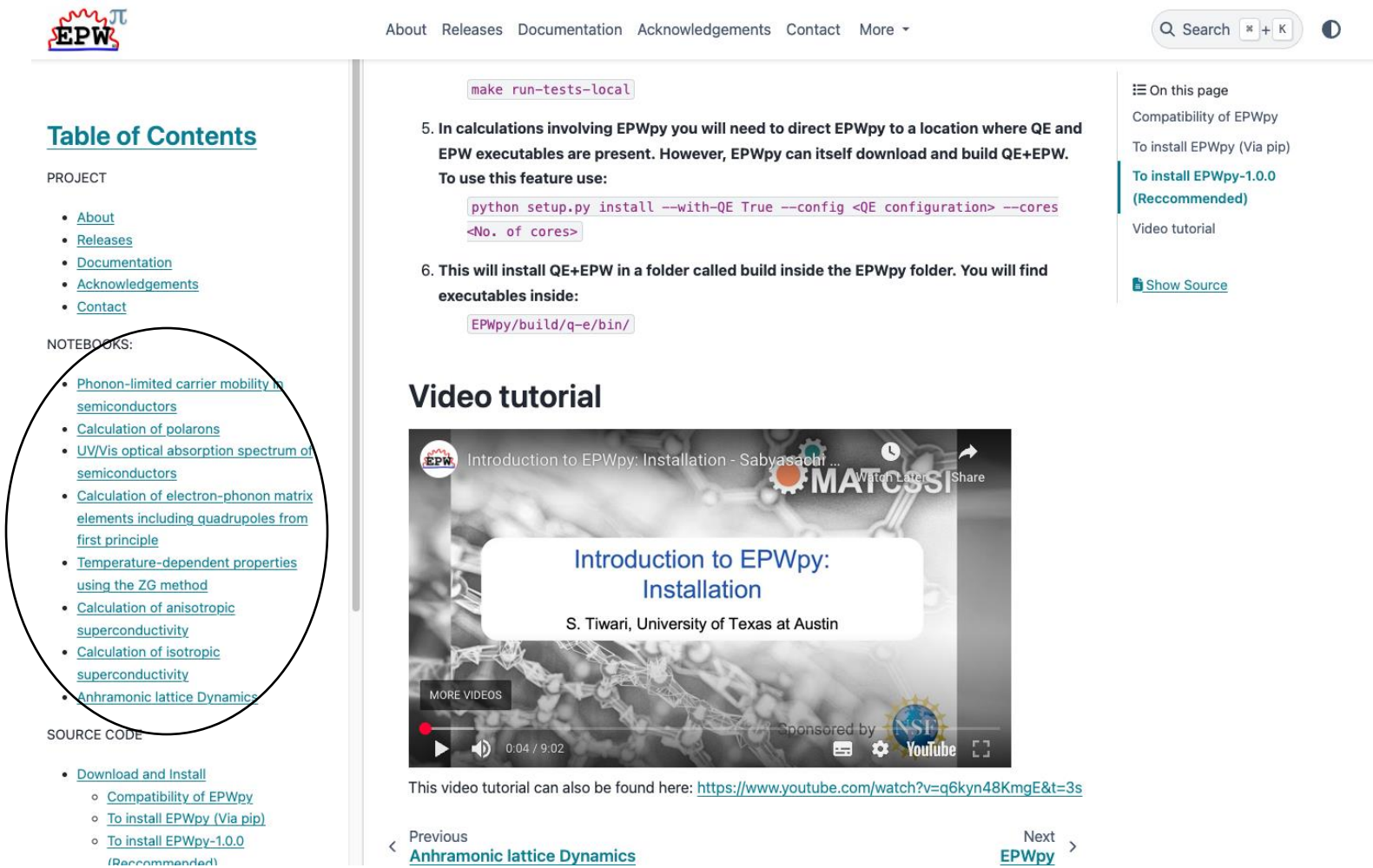
```
silicon.GW(GW={'nbnd':80})
silicon.epsilon(epsilon={'degeneracy_check_override':' '})
silicon.prepare('epsilon')
silicon.run(192,'epsilon')

silicon.sigma(sigma={'degeneracy_check_override':' ',
                     'no_symmetries_q_grid':' ',
                     'screened_coulomb_cutoff':20.0,
                     'band_index_min': 1,
                     'band_index_max':12})
silicon.prepare('sigma')
silicon.run(192,'sigma')
```

- Within a few lines we perform a complex EPW calculation
- All data transfer is handled internally while maintaining full user control

EPWpy resources

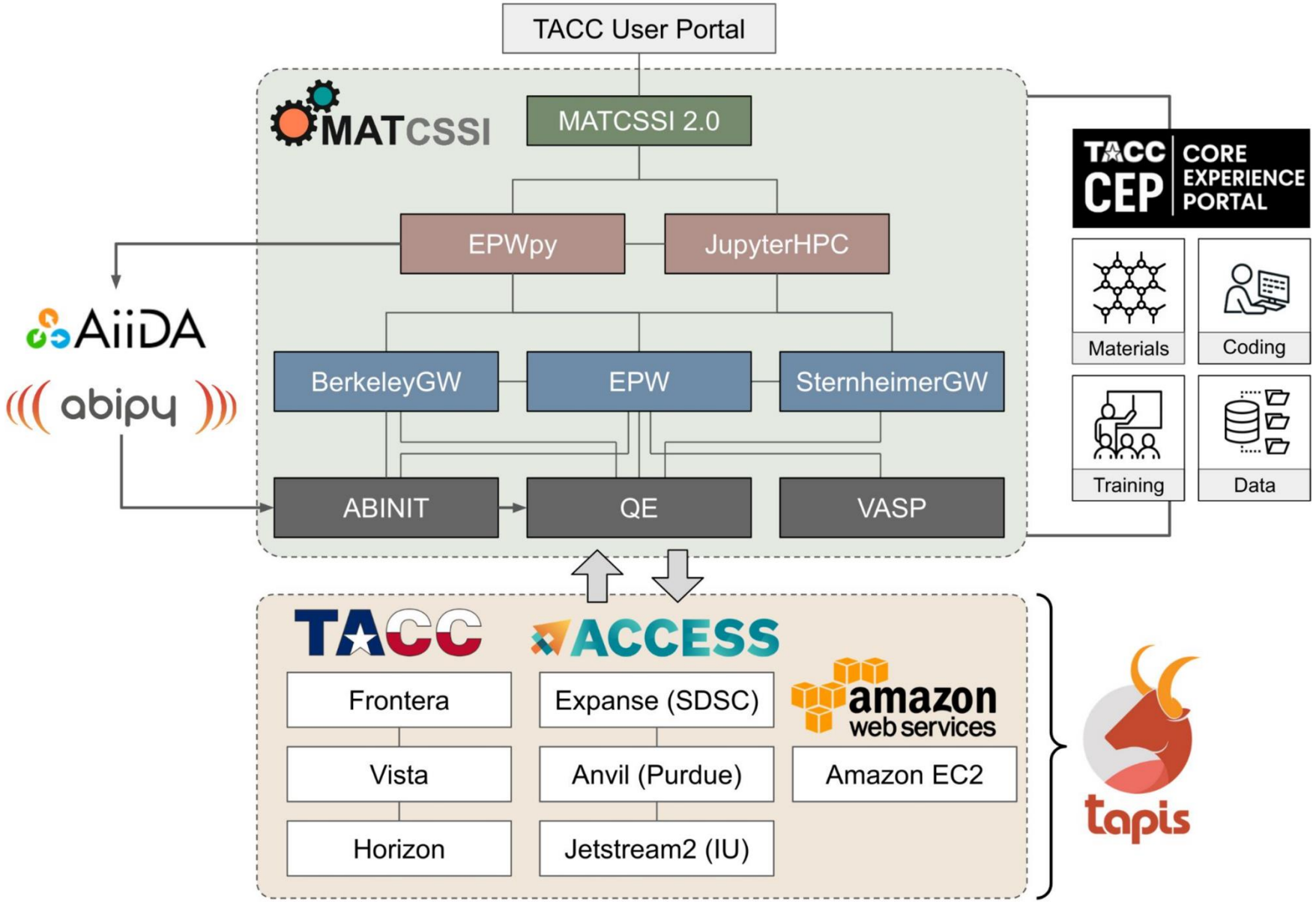
- EPWpy has a dedicated website
- Contains tutorials
- Notebooks can be run on TACC infrastructure



Outline

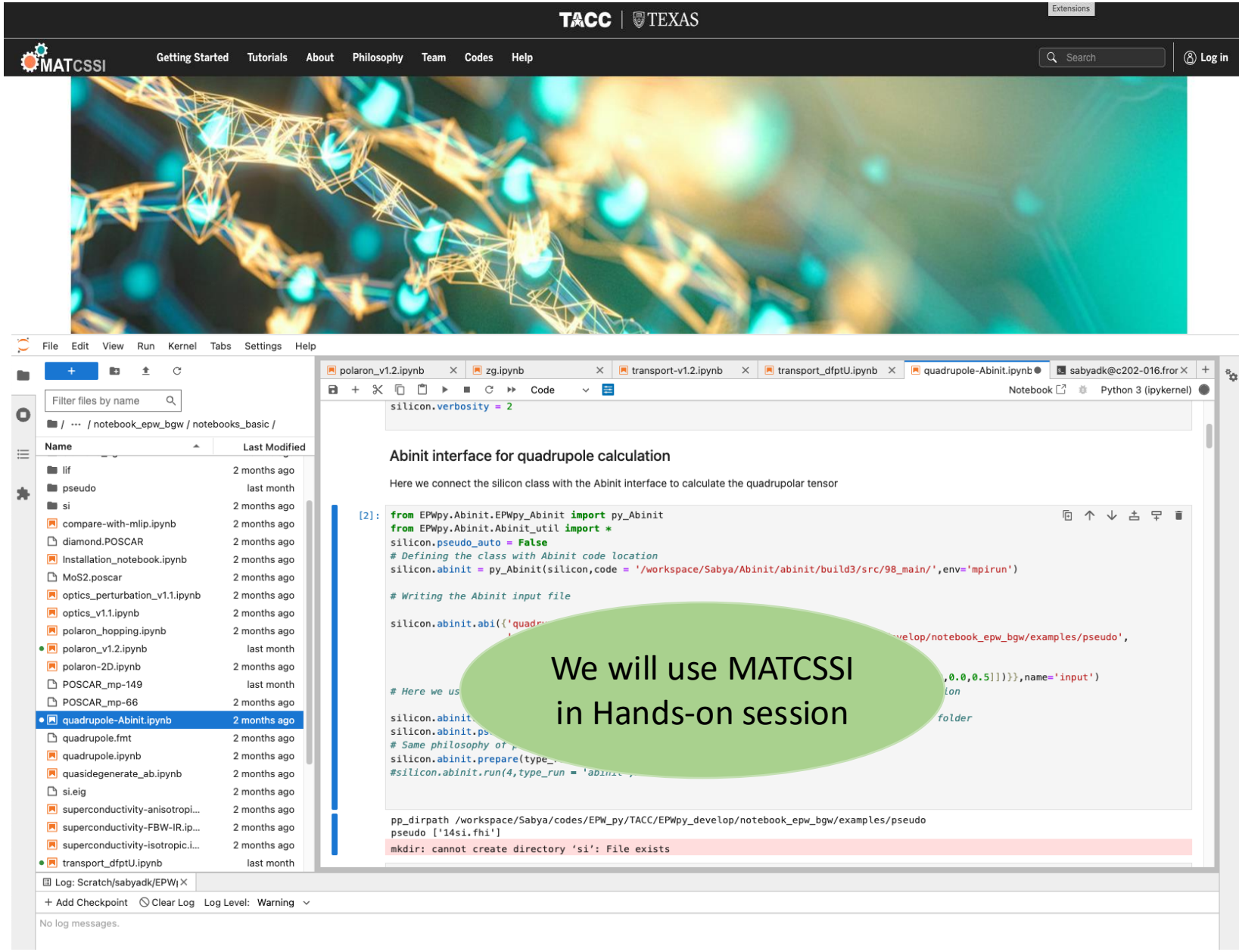
- Electron-Phonon-Physics with EPW
- Programmable abstraction using EPWpy
- Integrated MATCSSI portal
- Conclusion

Enabling interoperable *ab initio* simulation environment



MATCSSI

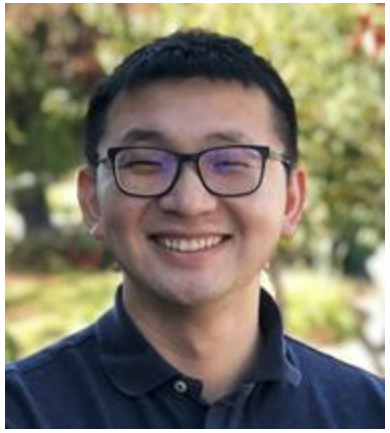
- MATCSSI is a portal powered by TACC
- Large scale calculations using EPWpy can be performed using Jupyter notebook (MPI-TACC)
- Allocations can be requested



Feliciano Giustino
UT Austin

Dan Stanzione
TACC

Maytal Dahan
TACC



Zhenglu Li
USC

Steven Louie
UC Berkeley

Roxana Margine
Binghamton U

This initiative is supported by the Office of Advanced Cyberinfrastructure and the Condensed Matter and Materials Theory division of the National Science Foundation under the CSSI Frameworks program, Award #2513830

Outline

- Electron-Phonon-Physics with EPW
- Programmable abstraction using EPWpy
- Integrated MATCSSI portal
- Conclusion

Conclusion

- Electron-Phonon physics is important for many physical properties
- EPW and EPWpy provide a convenient tool for exploring electron-phonon physics
- EPW GitLab: https://gitlab.com/epw/q-e
- EPWpy GitLab: https://gitlab.com/epwpy/epwpy

Thank You


MATCSSI
Questions?