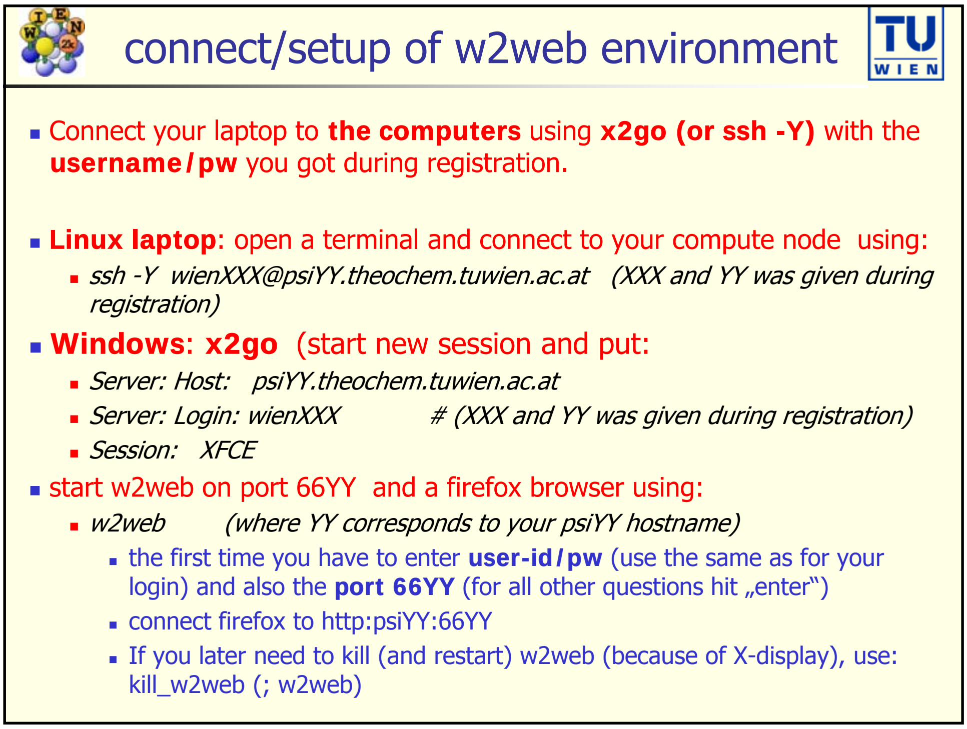
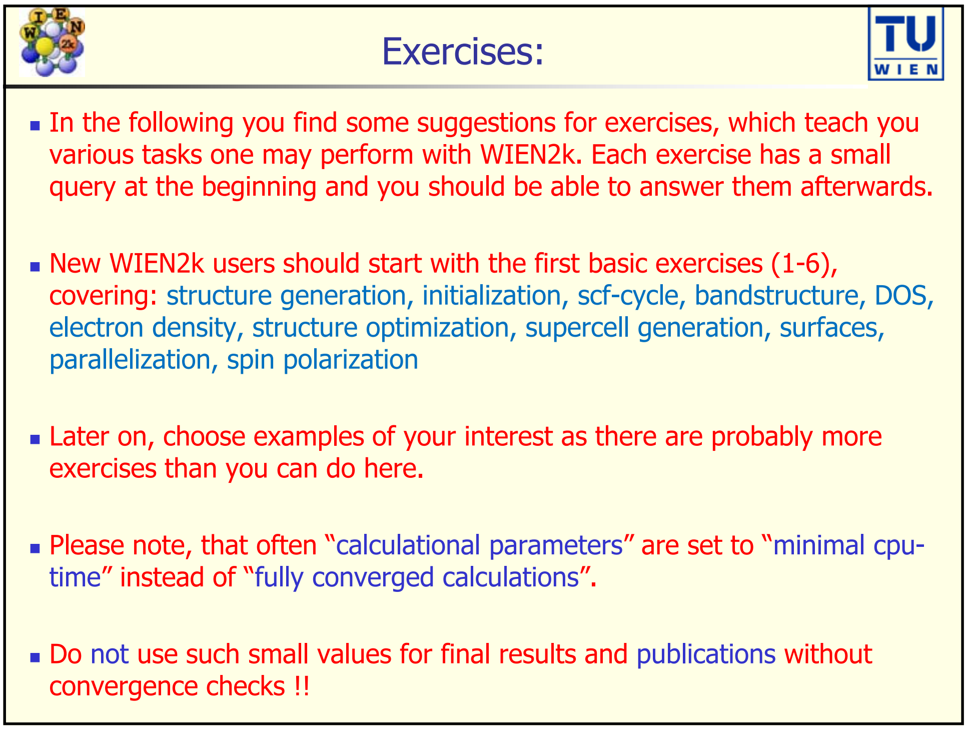


connect/setup of w2web environment

- Connect your laptop to **the computers** using **x2go (or ssh -Y)** with the **username/pw** you got during registration.

- **Linux laptop**: open a terminal and connect to your compute node using:
  - *ssh -Y wienXXX@psiYY.theochem.tuwien.ac.at (XXX and YY was given during registration)*
- **Windows**: **x2go** (start new session and put:
  - *Server: Host: psiYY.theochem.tuwien.ac.at*
  - *Server: Login: wienXXX # (XXX and YY was given during registration)*
  - *Session: XFCE*
- start w2web on port 66YY and a firefox browser using:
  - *w2web (where YY corresponds to your psiYY hostname)*
    - the first time you have to enter **user-id/pw** (use the same as for your login) and also the **port 66YY** (for all other questions hit „enter")
    - connect firefox to http:psiYY:66YY
    - If you later need to kill (and restart) w2web (because of X-display), use: kill_w2web (; w2web)



Exercises:

- In the following you find some suggestions for exercises, which teach you various tasks one may perform with WIEN2k. Each exercise has a small query at the beginning and you should be able to answer them afterwards.

- New WIEN2k users should start with the first basic exercises (1-6), covering: structure generation, initialization, scf-cycle, bandstructure, DOS, electron density, structure optimization, supercell generation, surfaces, parallelization, spin polarization

- Later on, choose examples of your interest as there are probably more exercises than you can do here.

- Please note, that often "calculational parameters" are set to "minimal cpu-time" instead of "fully converged calculations".

- Do not use such small values for final results and publications without convergence checks !!

Content

- 1) Basic tasks with w2web
- 2) Volume optimization
- 3) Optimization of free atomic positions
- 4) commandline interface, supercells, surfaces
- 5) parallelization
- 6) ferromagnetism
- 7) antiferromagnetism
- 8) mBJ and hybrid-DFT
- 9) Xspec (XANES)
- 10) DFT+U
- 11) optical properties
- 12) valence photoelectron spectra
- 13) phonons
- 14) atoms-in-molecules
- 15) Hyperfine interactions (EFG, NMR)
- 16) wannier functions, Berry charges, topology, fold2Bloch

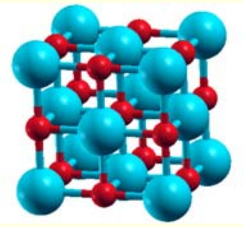
Exercise 1: Getting started:

- Learn basic operations/tasks with the w2web interface of WIEN2k.
- (structure generation, initialization, scf cycle, DOS, electron density with Xcrysdens, bandstructure)

- Questions:
  - *How many iterations did you need for scf ?*
  - *How long takes a single scf-cycle ?*
  - *How many k-points did you actually use in the IBZ ?*
  - *Which states dominate at: -15 eV: -8 to -2.5 eV: -2.5 to EF:*
  - *EF to 2.5 eV: 2.5 to 6 eV:*
  - *At what energy are the semi-core Ti-3s and Ti-3p states ?*
  - *Is Ti positively or negatively charged ? Why ?*
  - *Has Ti a larger eg or t2g occupation ?*

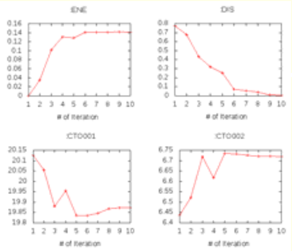
Exercise 1: Getting started:

- i) After w2web has been started, connect with firefox to w2web (psiYY:ZZZZ)
- iv) Try the "quick-start" example for **TiN** (similar to TiC in the UG)
  - *create new session named "TiN", "create" and "select" the suggested directory.*
  - *Generate structure (a=4.235 Ang; reduce RMT by 1%)*
  - *view structure with Xcrysden (switch primitive / conventional cell)*

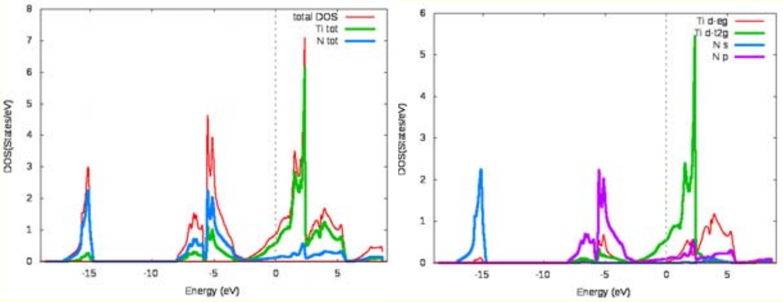


  - *initialize (fast mode; use defaults)*
  - *scf-cycle (run_lapw); use defaults; monitor "STDOUT" (reload in reverse order)*
  - *check "dayfile" (in utils)*

TiN continued



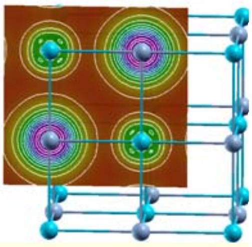
- *utilities: analyse*
  - (:ENE, :DIS, :CTO), graphically
- *utilities: save_lapw (use as save-name: "TiN_exp_pbe_rk7_1000k")*
- ***DOS**: (do the **necessary** steps)*
- *(plot 7 cases: total + Ti-tot + N-tot and Ti-eg + Ti-t2g + N-s + N-p)*



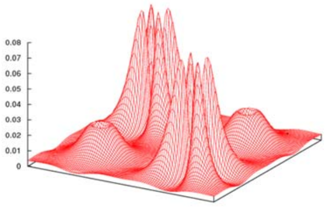
TiN continued …

- *electron density (use xcrysden to select the (100) plane), view it in xcrysden and rhoplot to "understand contour and 3D-plots")*
  - **valence** density (without semicore: check TiN.scf1 to find a EMIN which truncates the Ti-3s,3p states, rerun x lapw2 with this EMIN); compare the density around Ti with TiC (UG)
  - **difference** density (observe "charge transfer" and "t_{2g}-anisotropy" around Ti)
  - densities of the "**N-p**" and "**occupied Ti-d-band**" (get the corresponding E-intervals from DOS-plots (in Ry!) and use these energies in the "x lapw2" step; observe the e_g and t_{2g} asymmetry around Ti and the different N-p "weights", explain the chemical bonding

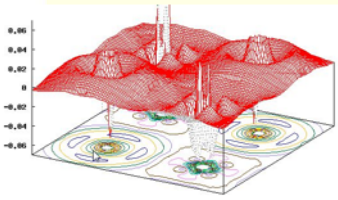
valence ρ



Ti-d band

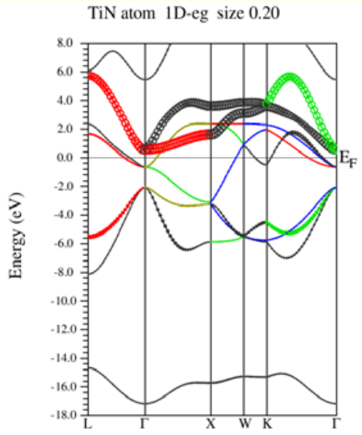


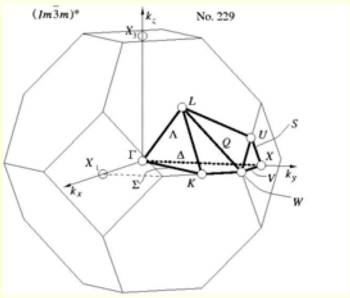
difference density



TiN continued

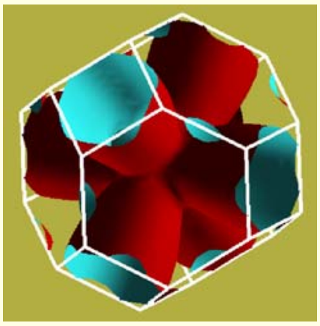
- bandstructure (along L-Gamma-X-W-K-Gamma with "character plotting")
  - *use xcrysden (100 points, save as „TiN.klist_band")*
  - *identify "t2g-" and "eg-" bands (fat band plots) by setting atom/column in TiN.insp*





TiN continued …

- *Fermi surfaces*
  - open a terminal, change into the TiN directory and issue:
  - `xcrysden --wien_fermisurface .`
    - choose a good k-mesh (eg. 10000 points);
    - plot the FS for all bands (**9**, 10,11) which cross E_F and compare to band structure



Exercises 2: lattice parameter of TiC

- Volume optimization in WIEN2k
- learn how to check computational parameters for converged results (reusing prev. calculations), fix a QTL-B problem and set a HDLO

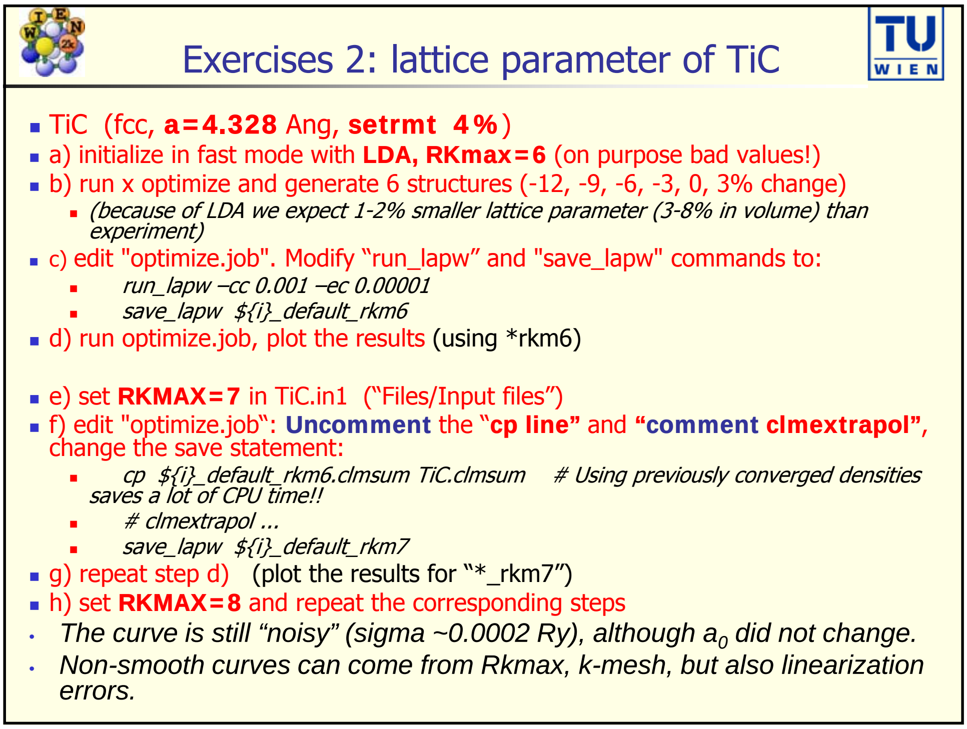
- Questions: (the exp. lattice parameter is 4.328 Ang)
  - *Is TiC "hard" or "soft" ? Why ?*

  - *What are the theoretical lattice parameters/bulk modulus with*

RKmax	6	7	8	7-hdlo	7-hdlo/10000k
a_0/B					

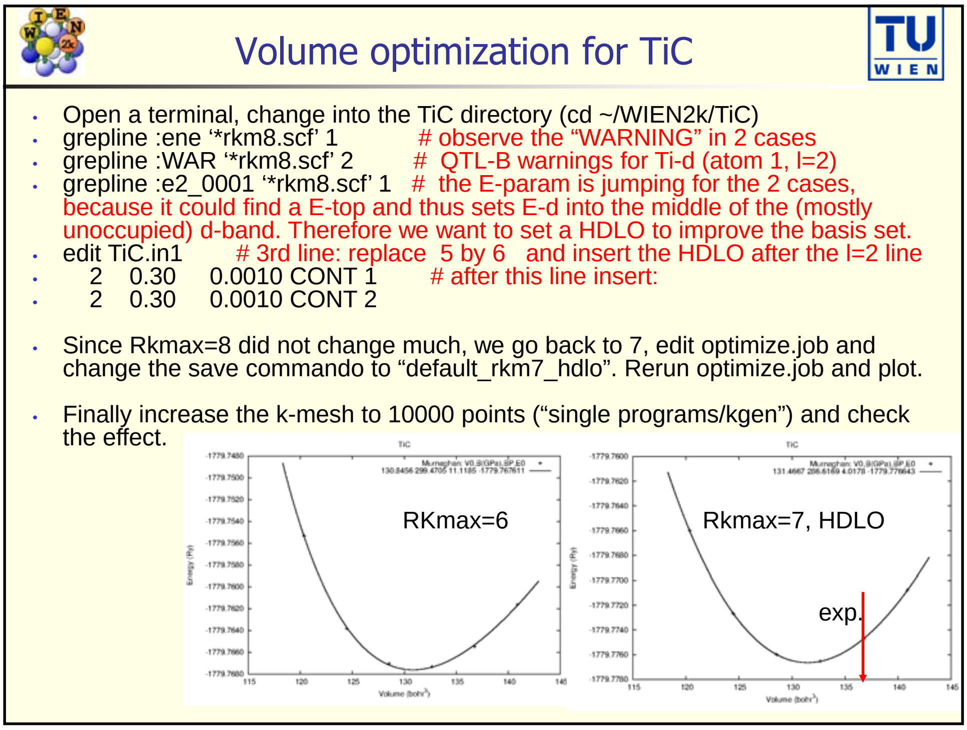
  - *What could we change to get better agreement with experiment ?*

- Depending on your elements, RMTs, bandstructure (metals) and required accuracy fully converged results may require Rkmax ~ 9-10 , 10000 k and 10 volumes with ΔV=1%.
- Different properties may require different parameters.
- Different types of atoms, insulator/metal and system size may need different parameters:
  - *H: RKmax > 2.5; sp-elements: RKmax > 5; d-elements: RKmax > 6; f-elements: RKmax > 7; (see our faq-page)*
  - *1 atom/cell, metal: 1000-10000 k-points or more*
  - *1 atom/cell, insulator: 100-1000 k-points or more*
  - *For N atoms/cell you can reduce the k-mesh by a factor N*



Exercises 2: lattice parameter of TiC

- TiC (fcc, **a=4.328** Ang, **setrmt 4%**)
- a) initialize in fast mode with **LDA, RKmax=6** (on purpose bad values!)
- b) run x optimize and generate 6 structures (-12, -9, -6, -3, 0, 3% change)
  - *(because of LDA we expect 1-2% smaller lattice parameter (3-8% in volume) than experiment)*
- c) edit "optimize.job". Modify "run_lapw" and "save_lapw" commands to:
  - *run_lapw –cc 0.001 –ec 0.00001*
  - *save_lapw ${i}_default_rkm6*
- d) run optimize.job, plot the results (using *rkm6)
- e) set **RKMAX=7** in TiC.in1 ("Files/Input files")
- f) edit "optimize.job": **Uncomment** the "**cp line"** and **"comment clmextrapol"**, change the save statement:
  - *cp ${i}_default_rkm6.clmsum TiC.clmsum # Using previously converged densities saves a lot of CPU time!!*
  - *# clmextrapol ...*
  - *save_lapw ${i}_default_rkm7*
- g) repeat step d) (plot the results for "*_rkm7")
- h) set **RKMAX=8** and repeat the corresponding steps
- *The curve is still "noisy" (sigma ~0.0002 Ry), although a_0 did not change.*
- *Non-smooth curves can come from Rkmax, k-mesh, but also linearization errors.*



Volume optimization for TiC

- Open a terminal, change into the TiC directory (cd ~/WIEN2k/TiC)
- grepline :ene '*rkm8.scf' 1 # observe the "WARNING" in 2 cases
- grepline :WAR '*rkm8.scf' 2 # QTL-B warnings for Ti-d (atom 1, l=2)
- grepline :e2_0001 '*rkm8.scf' 1 # the E-param is jumping for the 2 cases, because it could find a E-top and thus sets E-d into the middle of the (mostly unoccupied) d-band. Therefore we want to set a HDLO to improve the basis set.
- edit TiC.in1 # 3rd line: replace 5 by 6 and insert the HDLO after the l=2 line
- 2 0.30 0.0010 CONT 1 # after this line insert:
- 2 0.30 0.0010 CONT 2
- Since Rkmax=8 did not change much, we go back to 7, edit optimize.job and change the save commando to "default_rkm7_hdlo". Rerun optimize.job and plot.
- Finally increase the k-mesh to 10000 points ("single programs/kgen") and check the effect.

Exercise 3: optimization of positions in $Mg(OH)_2$

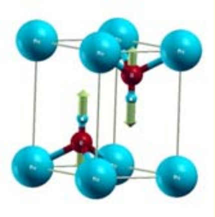
- **learn how to optimize internal atomic positions ("structure optimization").** In principle this should be done in every case, where forces are above 5 mRy/bohr. Often, for atomic positions DFT is more accurate than experiment (while for lattice parameters it is certainly not).
- learn how to use the UG for "arrows" plots
- questions:
  - *why did we use Rkmax=3.0 and only 100 k-points ?*
  - *what are the initial and final (relaxed) forces ?*
  - *how many iterations did you need to optimize positions?*
  - *what are the final z-coordinates for O: and H: atoms?*
  - *what are the O-H and Mg-O distances ?*

Exercise 3: optimization of positions in $Mg(OH)_2$

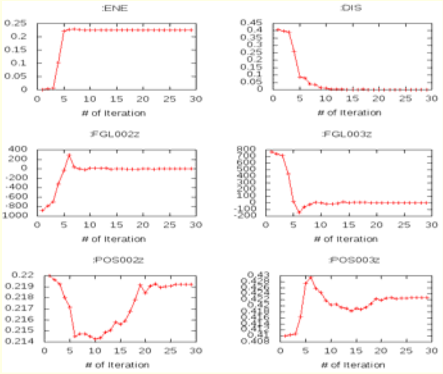
- P-3m1 (164), a=b=3.15 c=4.77 Å γ=120°; Mg(0,0,0) O(1/3,2/3,**0.22**) H(1/3,2/3,**0.41**); RMT: reduce by 7%
- init_lapw -b –numk 100 –rkmax 3
- scf cycle with force convergence (-fc 1)
- edit **case.scf** (and find final :FGL002z, :FGL003z); save case_initial
- minimization using MSR1a (second option from „mini positions") or activate MSR1a button in „scf"): it executes:
  - ***run -min –fc 1 –cc 0.001 –ec 0.0001***
    - -min sets MSR1a in case.inm, (sometimes a crude scf cycle to come closer to „Born-Oppenheimer" surface is necessary (run -fc 20)
- analyze **case.scf** and find out how many scf cycles you needed
    - :ENE :FGL002z :POS002z :FGL003z :POS003z :ITE
- save_lapw case_final
- use the „arrows" utility to display

  initial forces and final relaxations

  (search the UG: „arrows")



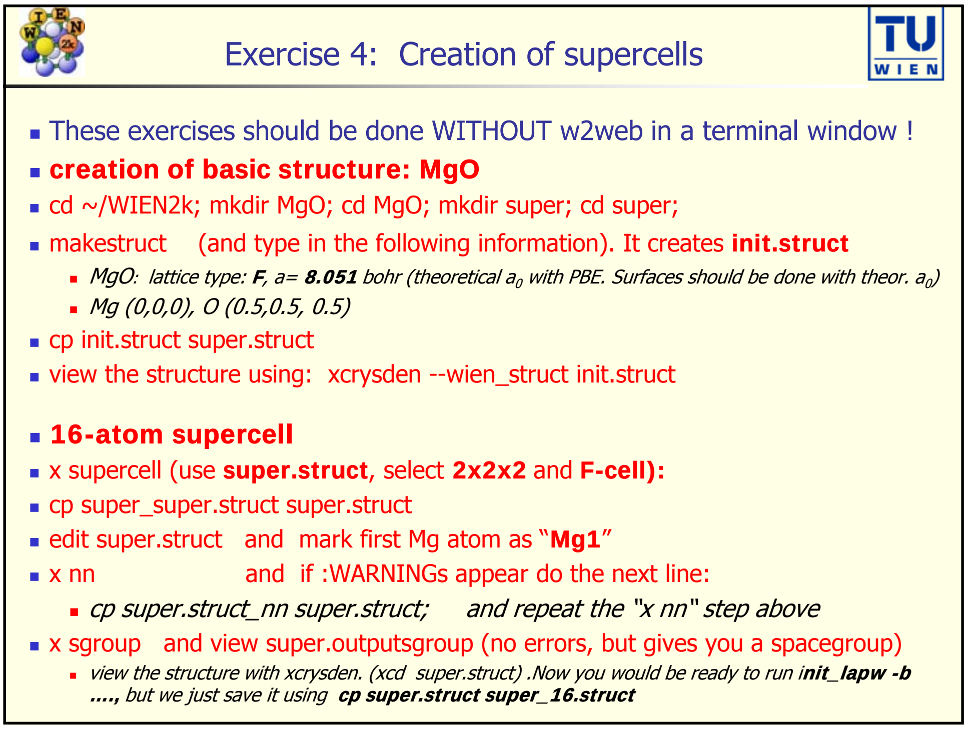
$Mg(OH)_2$ continue



- **check RKMAX convergence:**
  - *increase RKMAX to 3.5 (case.in1)*
  - *run_lapw –fc 1   (and check your forces afterwards)*

Exercise 4: Creation of supercells

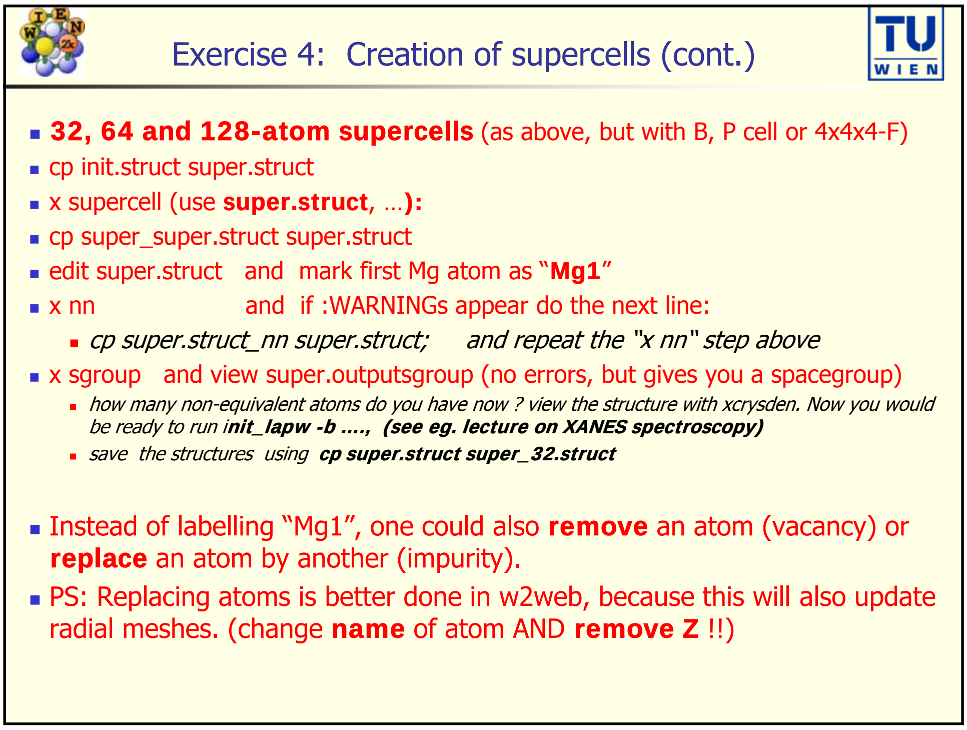
- Learn how to run WIEN2k from the command line
- learn how to create simple (x supercell) and complicated (structure editor with octave) supercells of various sizes
- questions:
  - *MgO 2x2x2 F cell: how many **non-equivalent** atoms do you have in super_super.struct   and the final super.struct ?*
  - *What has nn/sgroup done ?*
  - *MgO (001) surface:*
    - What has sgroup done ??
    - How many **total** (/   ) and **non-equivalent** (/   ) atoms and how many **atoms/layer** (/   ) do you have **before/after** sgroup ?
    - Do you have inversion symmetry ?
    - check the forces of unrelaxed scf-file. Which atoms will relax in which direction (into or out of the surface) ?
    - How much have the surface and sub-surface atoms finally relaxed ?
    - Where would you add an Fe atom ?



Exercise 4: Creation of supercells

- These exercises should be done WITHOUT w2web in a terminal window !
- **creation of basic structure: MgO**
- cd ~/WIEN2k; mkdir MgO; cd MgO; mkdir super; cd super;
- makestruct (and type in the following information). It creates **init.struct**
  - *MgO: lattice type: **F**, a= **8.051** bohr (theoretical a_0 with PBE. Surfaces should be done with theor. a_0)*
  - *Mg (0,0,0), O (0.5,0.5, 0.5)*
- cp init.struct super.struct
- view the structure using: xcrysden --wien_struct init.struct

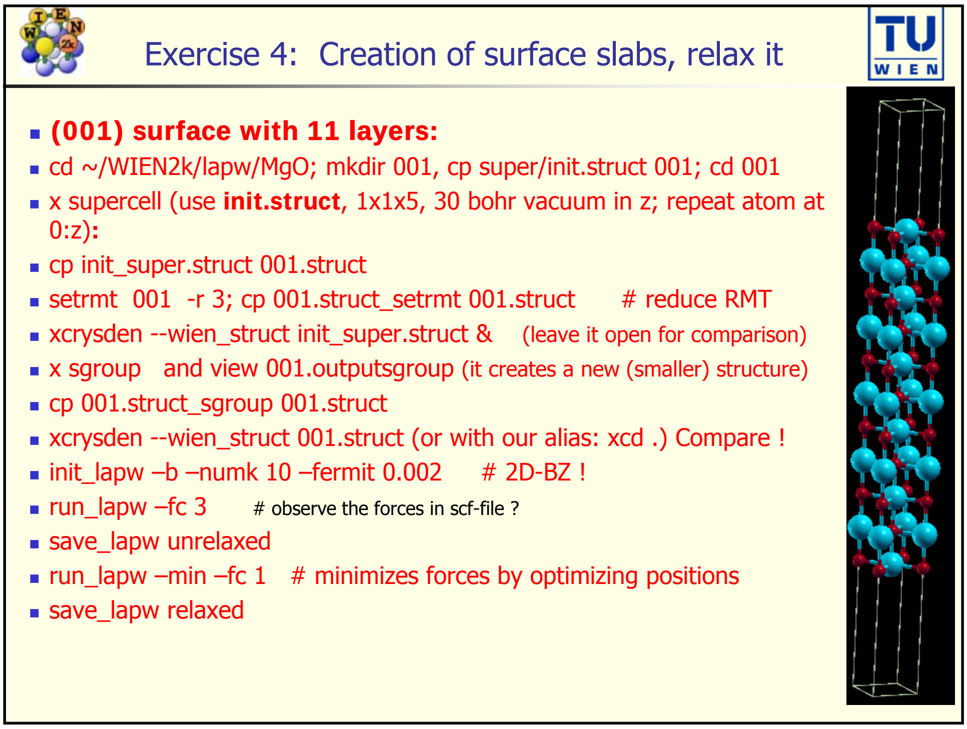
- **16-atom supercell**
- x supercell (use **super.struct**, select **2x2x2** and **F-cell):**
- cp super_super.struct super.struct
- edit super.struct and mark first Mg atom as "**Mg1**"
- x nn and if :WARNINGs appear do the next line:
  - *cp super.struct_nn super.struct; and repeat the "x nn" step above*
- x sgroup and view super.outputsgroup (no errors, but gives you a spacegroup)
  - *view the structure with xcrysden. (xcd super.struct) .Now you would be ready to run i**nit_lapw -b ….,** but we just save it using **cp super.struct super_16.struct***



Exercise 4: Creation of supercells (cont.)

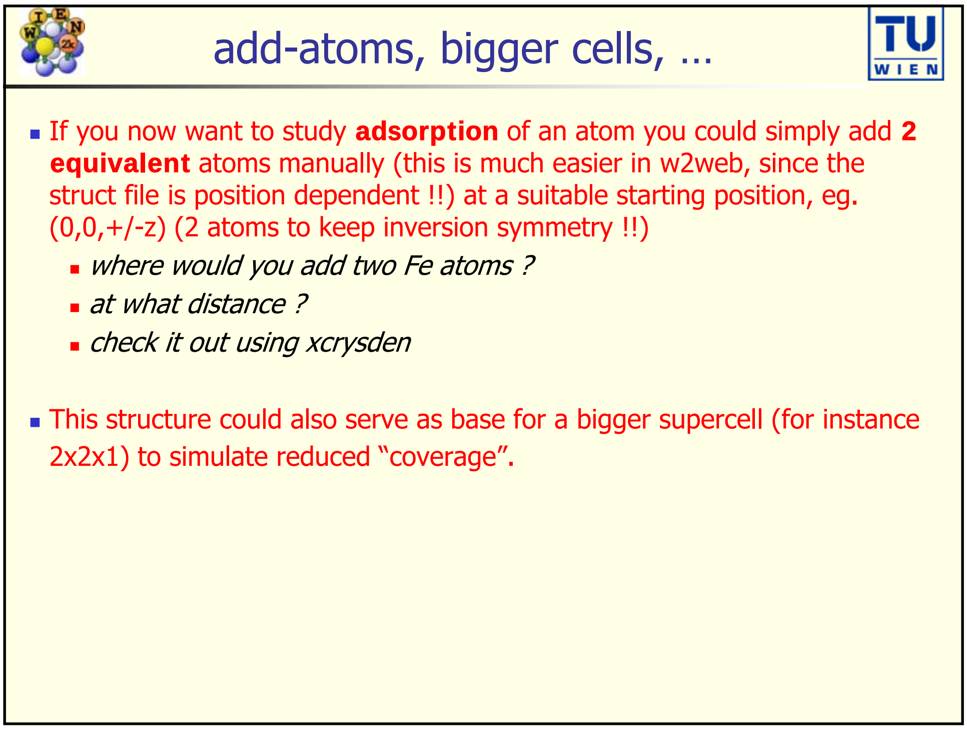
- **32, 64 and 128-atom supercells** (as above, but with B, P cell or 4x4x4-F)
- cp init.struct super.struct
- x supercell (use **super.struct**, …**):**
- cp super_super.struct super.struct
- edit super.struct and mark first Mg atom as "**Mg1**"
- x nn and if :WARNINGs appear do the next line:
  - *cp super.struct_nn super.struct; and repeat the "x nn" step above*
- x sgroup and view super.outputsgroup (no errors, but gives you a spacegroup)
  - *how many non-equivalent atoms do you have now ? view the structure with xcrysden. Now you would be ready to run i**nit_lapw -b …., (see eg. lecture on XANES spectroscopy)***
  - *save the structures using **cp super.struct super_32.struct***

- Instead of labelling "Mg1", one could also **remove** an atom (vacancy) or **replace** an atom by another (impurity).
- PS: Replacing atoms is better done in w2web, because this will also update radial meshes. (change **name** of atom AND **remove Z** !!)



Exercise 4: Creation of surface slabs, relax it

- **(001) surface with 11 layers:**
- cd ~/WIEN2k/lapw/MgO; mkdir 001, cp super/init.struct 001; cd 001
- x supercell (use **init.struct**, 1x1x5, 30 bohr vacuum in z; repeat atom at 0:z)**:**
- cp init_super.struct 001.struct
- setrmt 001 -r 3; cp 001.struct_setrmt 001.struct # reduce RMT
- xcrysden --wien_struct init_super.struct & (leave it open for comparison)
- x sgroup and view 001.outputsgroup (it creates a new (smaller) structure)
- cp 001.struct_sgroup 001.struct
- xcrysden --wien_struct 001.struct (or with our alias: xcd .) Compare !
- init_lapw –b –numk 10 –fermit 0.002 # 2D-BZ !
- run_lapw –fc 3 # observe the forces in scf-file ?
- save_lapw unrelaxed
- run_lapw –min –fc 1 # minimizes forces by optimizing positions
- save_lapw relaxed

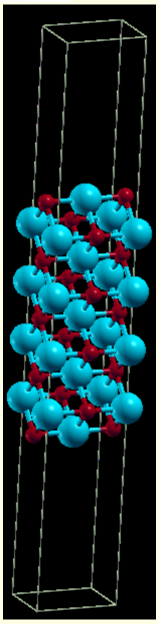


add-atoms, bigger cells, …

- If you now want to study **adsorption** of an atom you could simply add **2 equivalent** atoms manually (this is much easier in w2web, since the struct file is position dependent !!) at a suitable starting position, eg. (0,0,+/-z) (2 atoms to keep inversion symmetry !!)
  - *where would you add two Fe atoms ?*
  - *at what distance ?*
  - *check it out using xcrysden*
- This structure could also serve as base for a bigger supercell (for instance 2x2x1) to simulate reduced "coverage".

Exercise 4: Creation of supercells (structeditor)

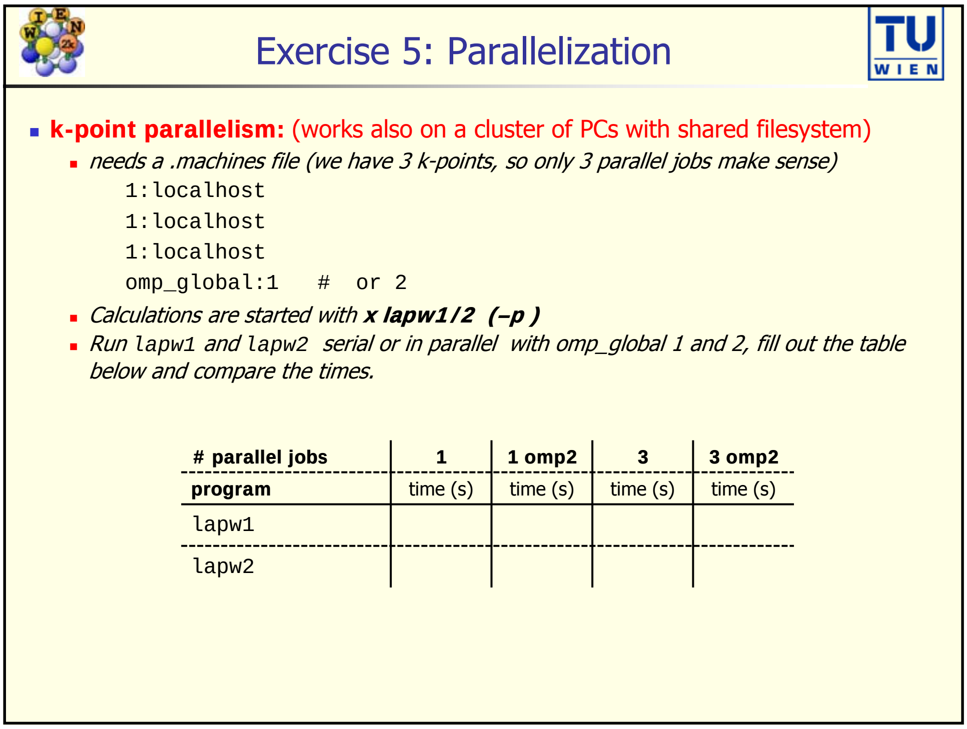
- **(110) surface of MgO with 9 layers:** (using the **structeditor)**
- cd ~/WIEN2k/super; octave (you can use repeat-key arrow-up !)
  - *helpstruct* *# list all possible commands*
  - *a=loadstruct("init.struct");*
  - *ac=makeconventional(a);* *# convert F into P cell*
  - *help makesurface* *# explains the syntax*
  - *sr=makesurface(ac, [1 1 0], 1, 20., 30.);*
  - *showstruct(sr)* *# check out the number of layers and repeat the **sr=makesurface** command with larger thickness until you get 9 layers. How do you get an O-atom at the origin ?*
  - *savestruct(sr,"super.struct")*
  - *quit*
- xcrysden --wien_struct super.struct &
- x sgroup and view super.outputsgroup
- cp super.struct_sgroup super.struct
- xcrysden --wien_struct super.struct
  - *what has sgroup done ?? how many total and non-equivalent atoms and how many **atoms/layer** do you have before/after sgroup ? Do you have inversion symmetry ?*
  - *save the structure using **cp super.struct super_surface-110.struct***



Exercise 5: Parallelization

- These tests should be done in the 001-directory (MgO-001 surface). We use this larger example to test the different ways parallelism is realized in WIEN2k. In order to get longer run times and see the parallelization effect more clearly, **increase Rkmax in 001.in1 to 9.5**
- **OMP parallelism (shared memory only):**
  - *Is used automatically, if the environment variable OMP_NUM_THREADS is set (2 in our setup). You can change it with:*
    - setenv OMP_NUM_THREADS X or
    - edit .machines and insert lines like: omp_global:X (omp_lapw0/1/2:X)
  - *Run* `x lapw0`*,* `x lapw1`*, and* `x lapw2` *with different number of threads and fill out the table below.*
  - *In a second terminal keep a top command running to keep track of the peak CPU usage! (timing may change if machines are overloaded by another user ! Note that psi31-39 have only 6 physical cores ! psi11-psi24 have 8 cores.)*

	# threads = 1		**# threads = 2**		**# threads = 4**		**# threads = 8**	
program	time (s)	% CPU	time (s)	% CPU	time (s)	% CPU	time (s)	% CPU
`lapw0`								
`lapw1`								
`lapw2`								



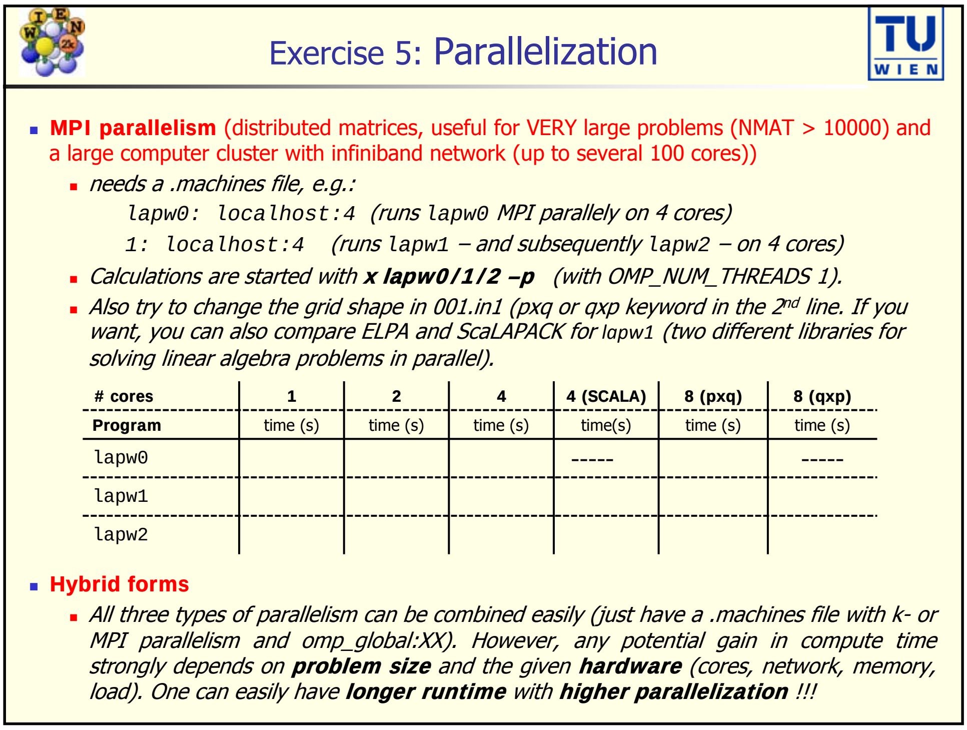
Exercise 5: Parallelization

- **k-point parallelism:** (works also on a cluster of PCs with shared filesystem)
  - *needs a .machines file (we have 3 k-points, so only 3 parallel jobs make sense)*

```
1:localhost
1:localhost
1:localhost
omp_global:1   #  or 2
```

  - *Calculations are started with **x lapw1/2 (-p)***
  - *Run* `lapw1` *and* `lapw2` *serial or in parallel with omp_global 1 and 2, fill out the table below and compare the times.*

# parallel jobs	1	1 omp2	3	3 omp2
program	time (s)	time (s)	time (s)	time (s)
lapw1				
lapw2				



Exercise 5: Parallelization

- **MPI parallelism** (distributed matrices, useful for VERY large problems (NMAT > 10000) and a large computer cluster with infiniband network (up to several 100 cores))
  - *needs a .machines file, e.g.:*

    `lapw0: localhost:4` (runs `lapw0` MPI parallely on 4 cores)

    `1: localhost:4` (runs `lapw1` – and subsequently `lapw2` – on 4 cores)
  - *Calculations are started with **x lapw0/1/2 –p** (with OMP_NUM_THREADS 1).*
  - *Also try to change the grid shape in 001.in1 (pxq or qxp keyword in the 2nd line. If you want, you can also compare ELPA and ScaLAPACK for* `lapw1` *(two different libraries for solving linear algebra problems in parallel).*

# cores	1	2	4	4 (SCALA)	8 (pxq)	8 (qxp)
Program	time (s)	time (s)	time (s)	time(s)	time (s)	time (s)
lapw0				-----		-----
lapw1						
lapw2						

- **Hybrid forms**
  - *All three types of parallelism can be combined easily (just have a .machines file with k- or MPI parallelism and omp_global:XX). However, any potential gain in compute time strongly depends on **problem size** and the given **hardware** (cores, network, memory, load). One can easily have **longer runtime** with **higher parallelization** !!!*

Exercise 6: spin-polarized calculations

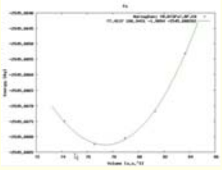
- Learn how to run magnetic calculations in WIEN2k
- Learn how magnetic moments and band width change with volume (pressure)

- questions:
  - *Why do we use Rkmax=8*
  - *What is the theoretical lattice parameter:*
  - *What is the Fe moment in PBE at the experimental and theoretical volume:*
  - *How do the moments and band width change with volume*

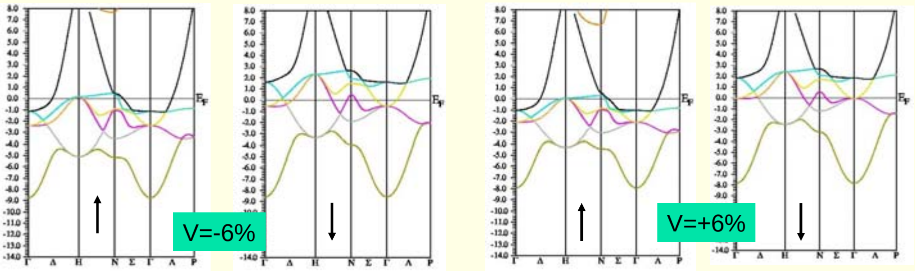
Volume	-6	-3	0	3	6
MMT					
Γ–Γ (eV)					
H-H (eV)					

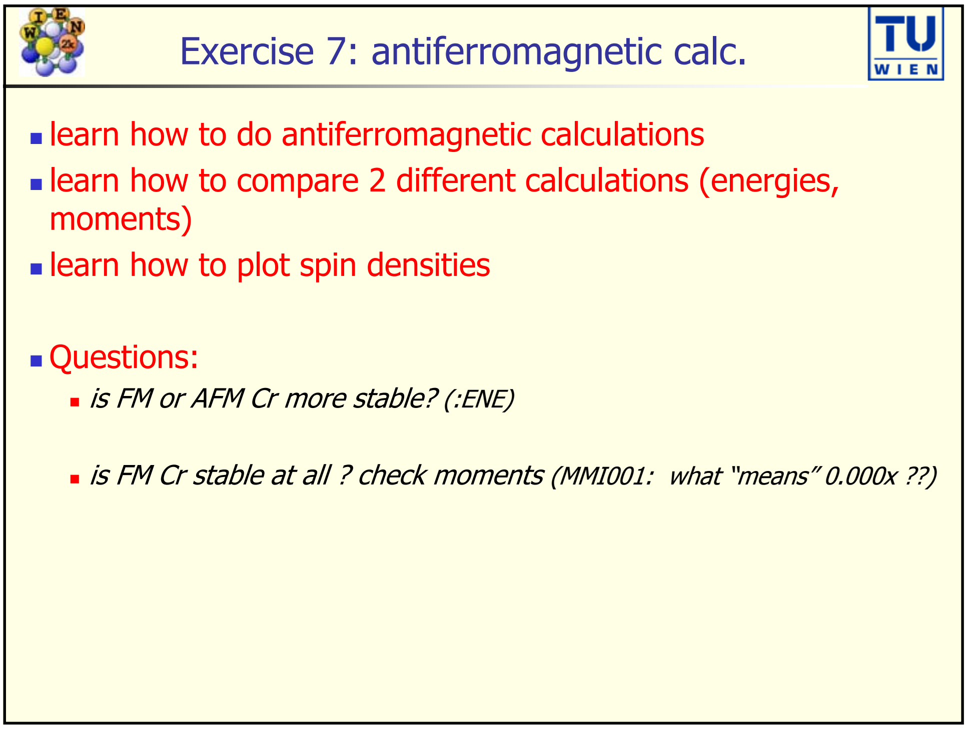
Exercise 6: spin-polarized calculations

- Magnetism: bcc Fe (a_0=2.86 Å)
  - *setrmt: 3%; 5000k; spin-polarization:yes, use RKmax=8*
  - *do a volume optimization (-6, -3, 0, 3, 6 %)* *(activate runsp_lapw instead of run_lapw !)*
    - check equilibrium volume, :MMT as function of volume



  - *compare bandstructure and DOS for large/small volumes (use restore_lapw for desired volume; adjust EF in case.insp)*

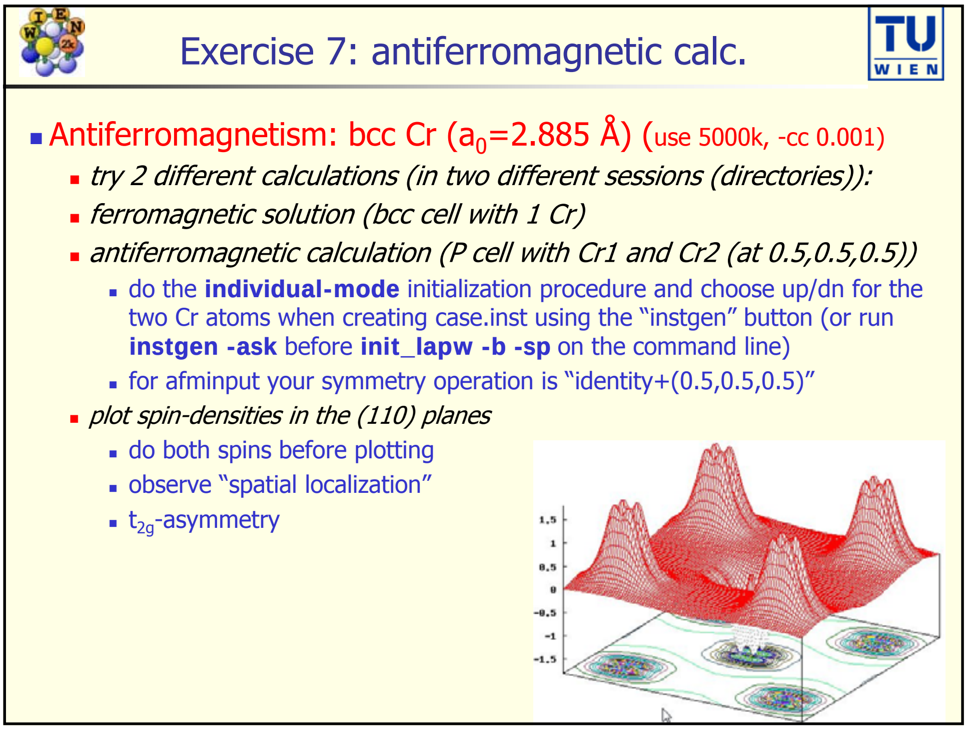




Exercise 7: antiferromagnetic calc.

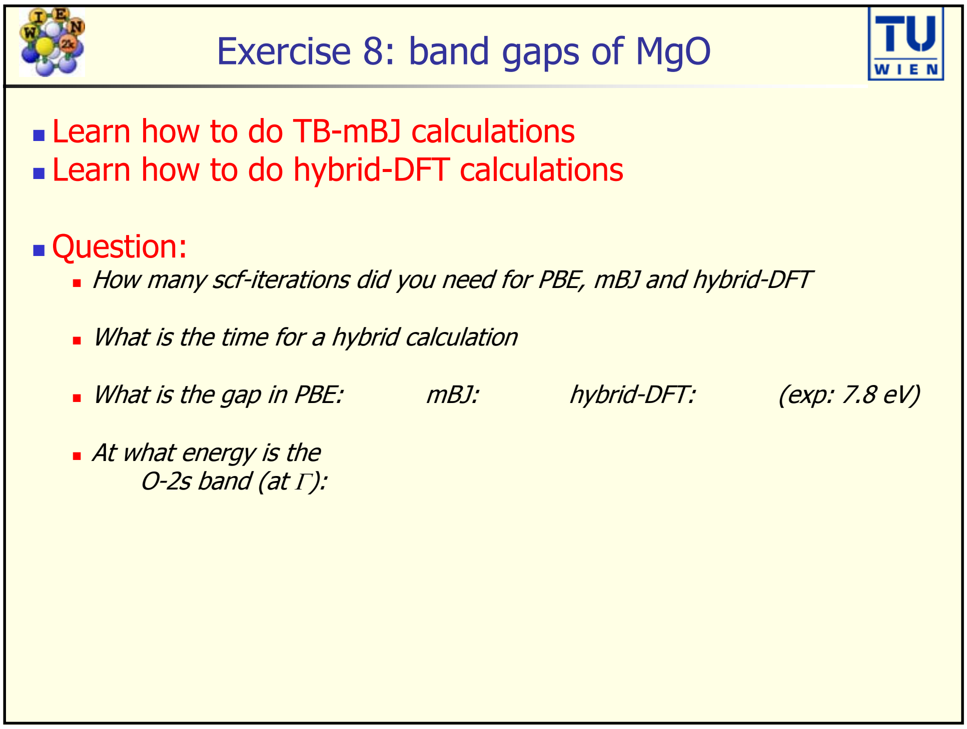
- learn how to do antiferromagnetic calculations
- learn how to compare 2 different calculations (energies, moments)
- learn how to plot spin densities

- Questions:
  - *is FM or AFM Cr more stable? (:ENE)*
  - *is FM Cr stable at all ? check moments (MMI001: what "means" 0.000x ??)*



Exercise 7: antiferromagnetic calc.

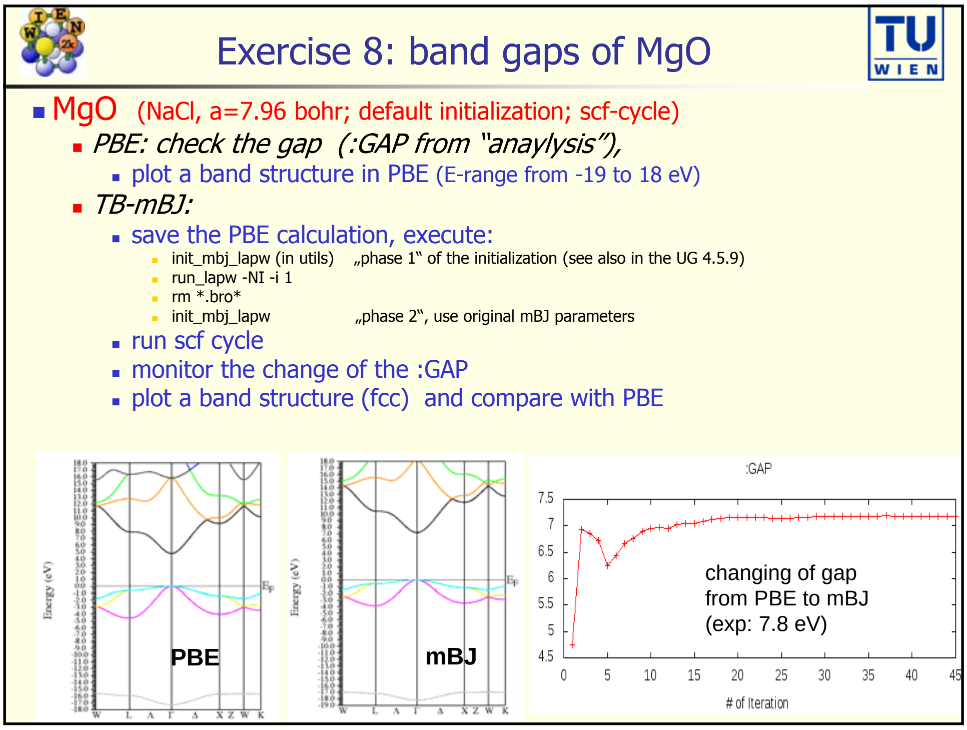
- Antiferromagnetism: bcc Cr (a_0=2.885 Å) (use 5000k, -cc 0.001)
  - *try 2 different calculations (in two different sessions (directories)):*
  - *ferromagnetic solution (bcc cell with 1 Cr)*
  - *antiferromagnetic calculation (P cell with Cr1 and Cr2 (at 0.5,0.5,0.5))*
    - do the **individual-mode** initialization procedure and choose up/dn for the two Cr atoms when creating case.inst using the "instgen" button (or run **instgen -ask** before **init_lapw -b -sp** on the command line)
    - for afminput your symmetry operation is "identity+(0.5,0.5,0.5)"
  - *plot spin-densities in the (110) planes*
    - do both spins before plotting
    - observe "spatial localization"
    - t_{2g}-asymmetry



Exercise 8: band gaps of MgO

- Learn how to do TB-mBJ calculations
- Learn how to do hybrid-DFT calculations

- Question:
  - *How many scf-iterations did you need for PBE, mBJ and hybrid-DFT*
  - *What is the time for a hybrid calculation*
  - *What is the gap in PBE: mBJ: hybrid-DFT: (exp: 7.8 eV)*
  - *At what energy is the O-2s band (at Γ):*



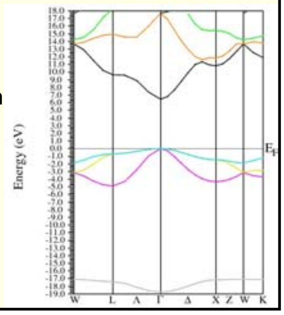
Exercise 8: band gaps of MgO

- MgO (NaCl, a=7.96 bohr; default initialization; scf-cycle)
  - *PBE: check the gap (:GAP from "anaylysis"),*
    - plot a band structure in PBE (E-range from -19 to 18 eV)
  - *TB-mBJ:*
    - save the PBE calculation, execute:
      - init_mbj_lapw (in utils) „phase 1" of the initialization (see also in the UG 4.5.9)
      - run_lapw -NI -i 1
      - rm *.bro*
      - init_mbj_lapw „phase 2", use original mBJ parameters
    - run scf cycle
    - monitor the change of the :GAP
    - plot a band structure (fcc) and compare with PBE

changing of gap from PBE to mBJ (exp: 7.8 eV)

Exercise 8: continued …

- Perform a hybrid-DFT calculation using YS-PBE0
  - create a new case, perform a PBE calculation and save the results.
  - the setup for hybrid-calculations can be made in w2web (Utils/init_hf_lapw), or in a terminal-window using „init_hf_lapw". (More details are given in the UG 4.5.8)
    - Select NBAND=12 (case.inhf)
    - and a 4x4x4 / 4x4x4 k-point mesh (no reduction)
  - scf cycle with **-hf -p** (insert 2 lines with **1:localhost** into **.machines**)
    - we do this in k-parallel since hybrids are expensive, alternatively we could also use a „reduced" hf-k-mesh, see UG
  - monitor the change of the :GAP (only every 2nd value is from HF !), use „scfmonitor :gap" on the command line.
  - plot a band structure:
    - only the k-mesh selection can be done in w2web, then open a terminal and change into the proper directory
    - run_bandplothf_lapw -p
    - cp $WIENROOT/SRC_templates/case.insp case.insp (insert E_F and increase the plotting energy range).
    - x spaghetti -hf -p



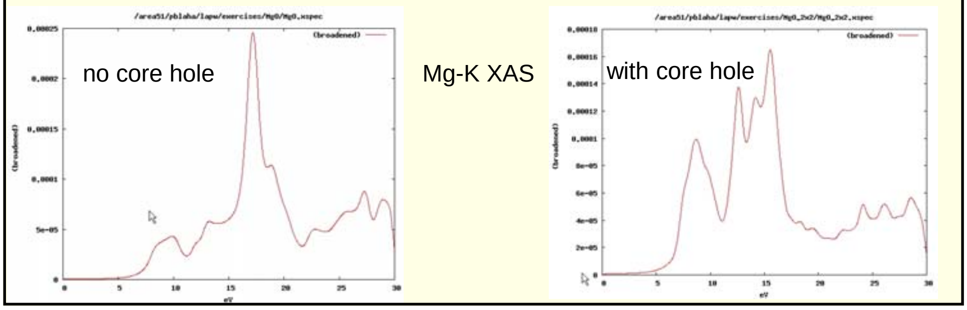
Exercise 9: Mg K-XAS in MgO

- Learn how to calculate XANES spectra (ELNES is very similar)
- Learn how to do a supercell calculation with a core-hole

- Questions:
  - *compare core-hole and no-core-hole calculations with experiment (see lecture notes)*

Exercise 9: Mg K-XAS in MgO

- MgO (NaCl structure, a=7.96 bohr; default initialization with 1000 k-points; scf-cycle)
  - *XSPEC task: larger EMAX in MgO.in1; select in MgO.inxs: Mg-K ABS from 0-30 eV, vary broadening)*
- Supercells: MgO 2x2x2 FCC-supercell for core-hole simulation
  - *create new "session", copy MgO.struct into new directory*
  - *x supercell; (specify proper struct-filename, 2x2x2, F-lattice)*
  - *cp supercell-struct file to correct name "case.struct"; "**label**" 1st atom (Mg → Mg1)*
  - *init_lapw (with 200k, RKmax=6.5)*
  - *edit case.inc (remove a core electron from 1st atom)*
  - *edit case.in2 (add one valence electron)*
  - *run_lapw (for bigger calc. use -it and compare timings for 1st and later iterations!)*
  - *edit case.in2 (remove extra valence electron)*
  - *XSPEC task for Mg-K XAS (see above)*



no core hole

Mg-K XAS

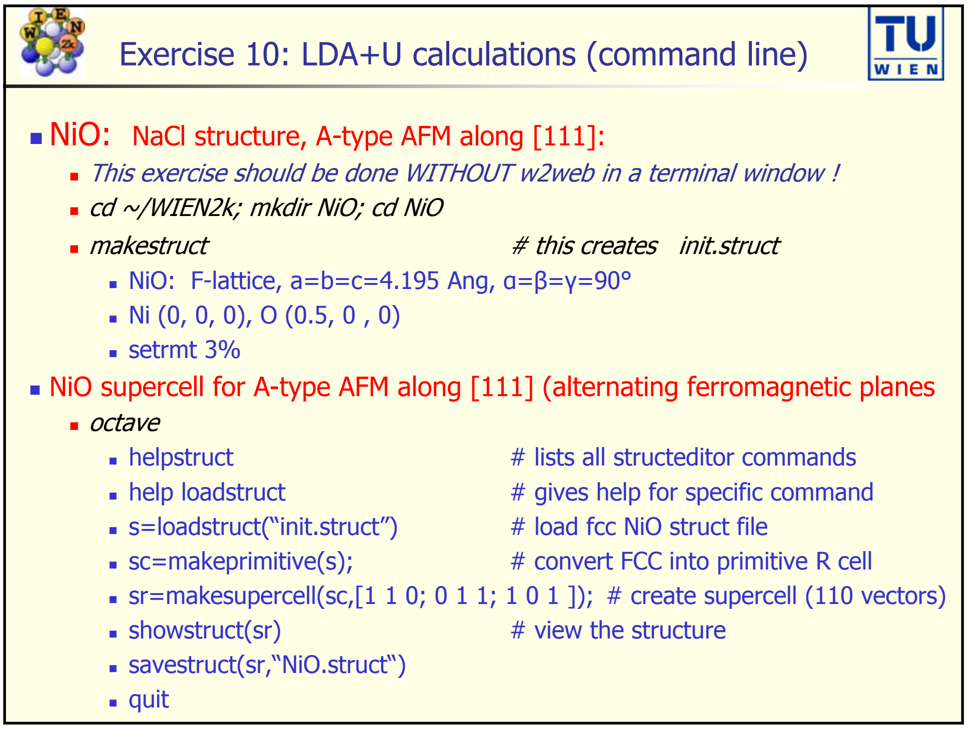
with core hole

Exercise 10: LDA+U calculations (command line)

- Learn how to do DFT+U calculations for correlated electrons
- Learn how to create a more complicated supercell for AFM structure
- Learn how to plot DOS (on the command line)
- Learn how to do a magnetic calculation with spin-orbit coupling
- Learn how to calculate orbital moments

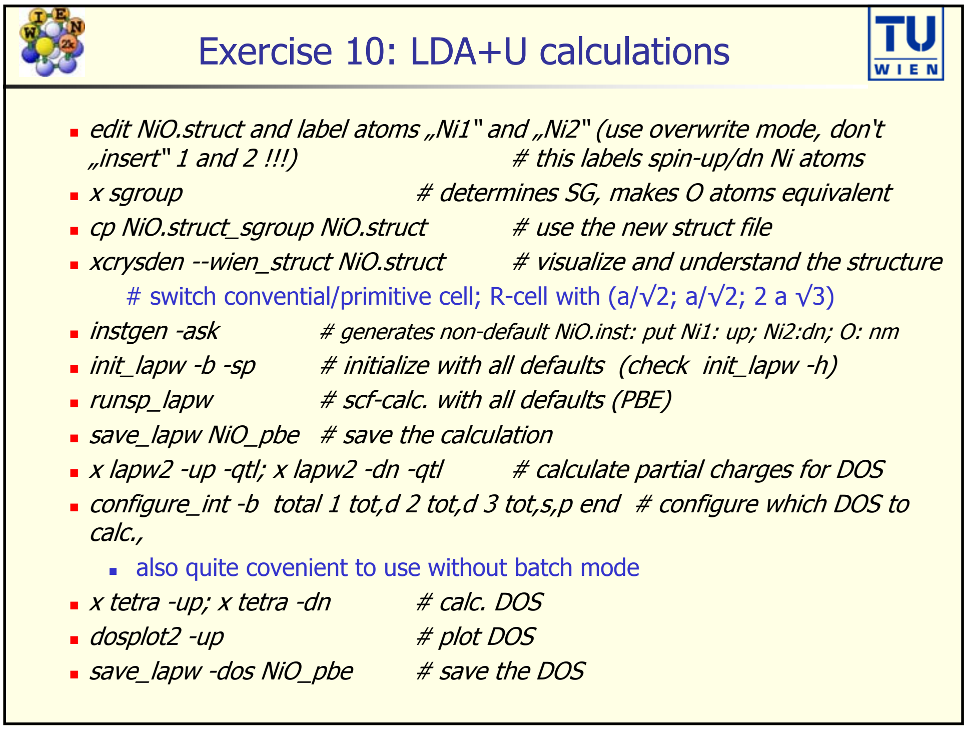
- Questions:
  - *compare DOS (total, Ni1, Ni2, O) for all calculations*
  - *compare gaps (exp: 4eV),spin and orbital moments, occupied Ni-d band position*

scheme	gap (eV)	spin m	orbital m	Ni-3d position
PBE				
PBE+U				
PBE+U+so				
mBJ				



Exercise 10: LDA+U calculations (command line)

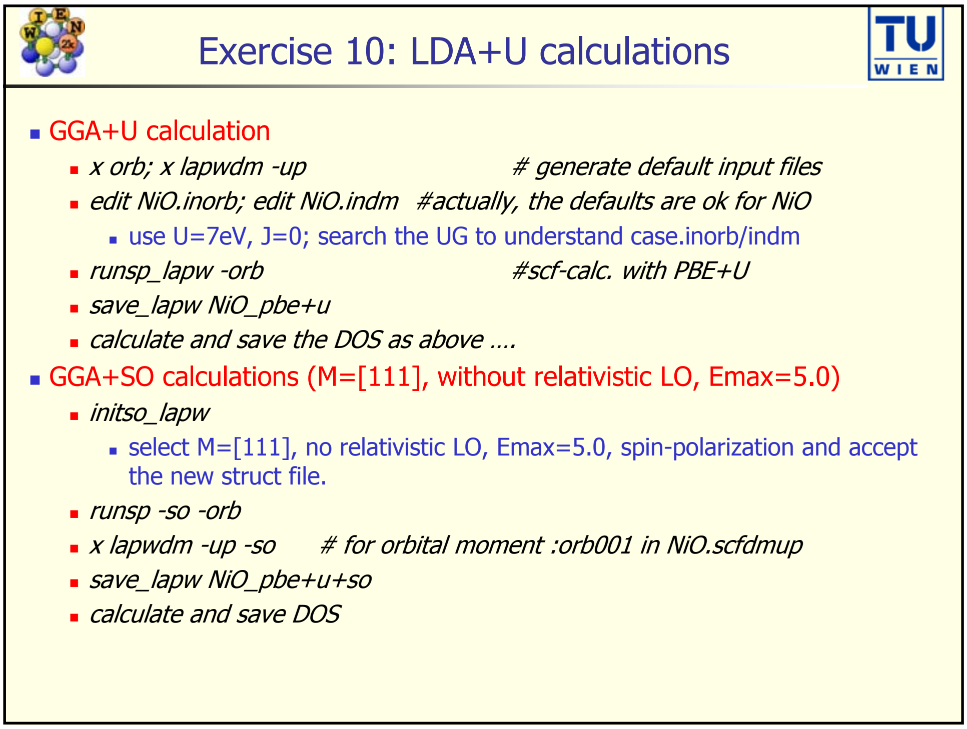
- NiO: NaCl structure, A-type AFM along [111]:
  - *This exercise should be done WITHOUT w2web in a terminal window !*
  - *cd ~/WIEN2k; mkdir NiO; cd NiO*
  - *makestruct # this creates init.struct*
    - NiO: F-lattice, a=b=c=4.195 Ang, α=β=γ=90°
    - Ni (0, 0, 0), O (0.5, 0 , 0)
    - setrmt 3%
- NiO supercell for A-type AFM along [111] (alternating ferromagnetic planes
  - *octave*
    - helpstruct # lists all structeditor commands
    - help loadstruct # gives help for specific command
    - s=loadstruct(“init.struct”) # load fcc NiO struct file
    - sc=makeprimitive(s); # convert FCC into primitive R cell
    - sr=makesupercell(sc,[1 1 0; 0 1 1; 1 0 1]); # create supercell (110 vectors)
    - showstruct(sr) # view the structure
    - savestruct(sr,"NiO.struct")
    - quit



Exercise 10: LDA+U calculations

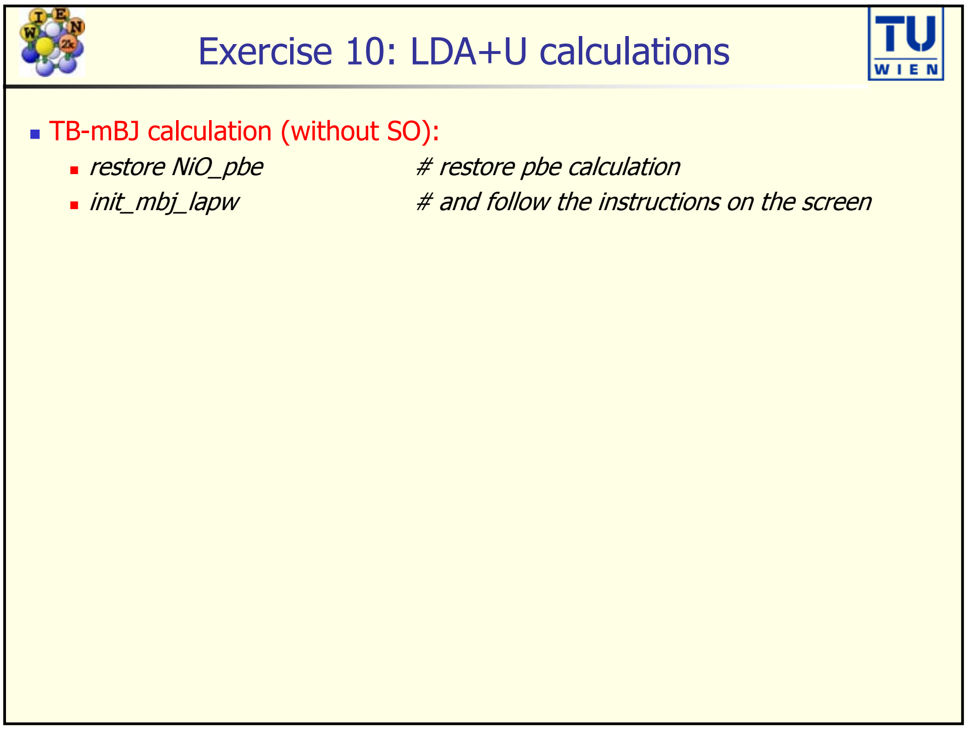
- *edit NiO.struct and label atoms „Ni1" and „Ni2" (use overwrite mode, don't „insert" 1 and 2 !!!) # this labels spin-up/dn Ni atoms*
- *x sgroup # determines SG, makes O atoms equivalent*
- *cp NiO.struct_sgroup NiO.struct # use the new struct file*
- *xcrysden --wien_struct NiO.struct # visualize and understand the structure*

  # switch convential/primitive cell; R-cell with (a/√2; a/√2; 2 a √3)
- *instgen -ask # generates non-default NiO.inst: put Ni1: up; Ni2:dn; O: nm*
- *init_lapw -b -sp # initialize with all defaults (check init_lapw -h)*
- *runsp_lapw # scf-calc. with all defaults (PBE)*
- *save_lapw NiO_pbe # save the calculation*
- *x lapw2 -up -qtl; x lapw2 -dn -qtl # calculate partial charges for DOS*
- *configure_int -b total 1 tot,d 2 tot,d 3 tot,s,p end # configure which DOS to calc.,*
  - also quite covenient to use without batch mode
- *x tetra -up; x tetra -dn # calc. DOS*
- *dosplot2 -up # plot DOS*
- *save_lapw -dos NiO_pbe # save the DOS*



Exercise 10: LDA+U calculations

- GGA+U calculation
  - *x orb; x lapwdm -up* *# generate default input files*
  - *edit NiO.inorb; edit NiO.indm #actually, the defaults are ok for NiO*
    - use U=7eV, J=0; search the UG to understand case.inorb/indm
  - *runsp_lapw -orb* *#scf-calc. with PBE+U*
  - *save_lapw NiO_pbe+u*
  - *calculate and save the DOS as above ….*
- GGA+SO calculations (M=[111], without relativistic LO, Emax=5.0)
  - *initso_lapw*
    - select M=[111], no relativistic LO, Emax=5.0, spin-polarization and accept the new struct file.
  - *runsp -so -orb*
  - *x lapwdm -up -so* *# for orbital moment :orb001 in NiO.scfdmup*
  - *save_lapw NiO_pbe+u+so*
  - *calculate and save DOS*



Exercise 10: LDA+U calculations

- TB-mBJ calculation (without SO):
  - *restore NiO_pbe* *# restore pbe calculation*
  - *init_mbj_lapw* *# and follow the instructions on the screen*

NiO cont…

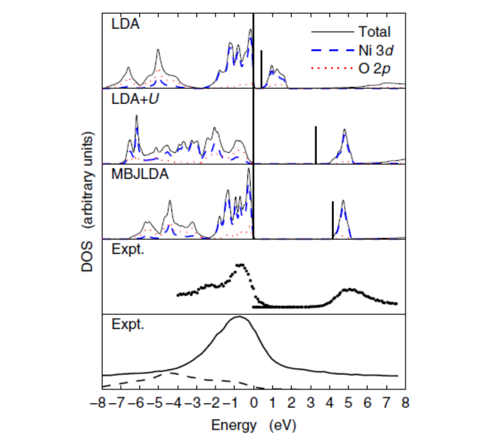
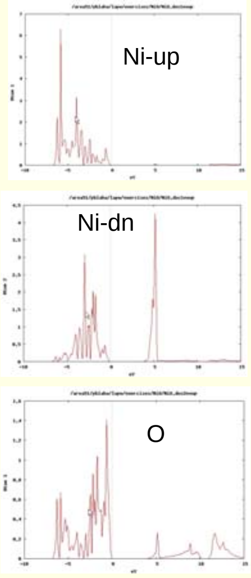


FIG. 2 (color online). DOS of NiO. The vertical bars indicate the end of the fundamental band gap which starts at $E = 0$ eV. The panels labeled “Expt.” show photoelectron [25] (upper panel) and XES [33] [lower panel, Ni (solid line) and O (dashed line) spectra] measurements.

from Tran,Blaha, PRL 102, 226401 (2009)

GGA+U



Exercise 11: optical properties

- Learn how to calculate optical properties (in the single particle approx.)
- Consider k-point convergence and relativistic effects

- Question:
  - *How many k-points do you need to converge optics for Al:*
  - *What is the plasma frequency in Al ?*

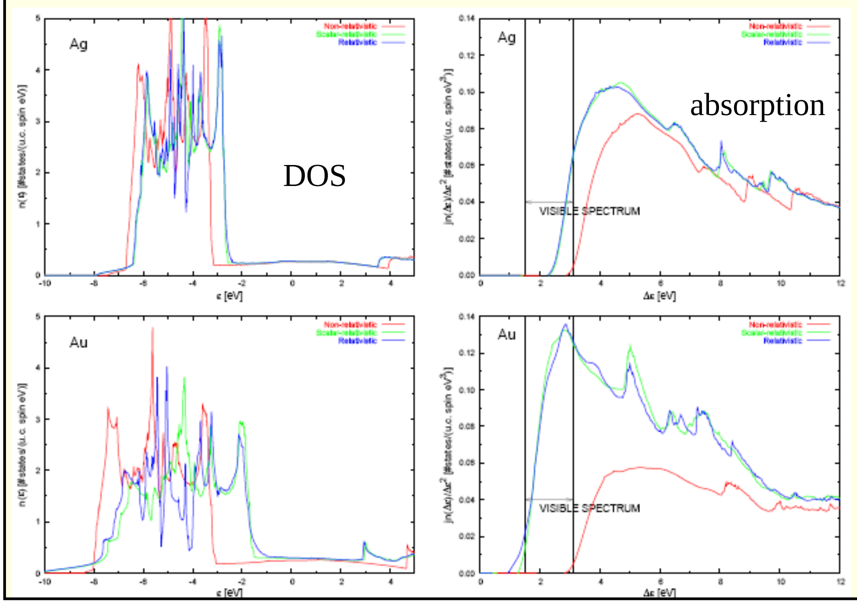
  - *What is the "optical gap" in Ag and Au:*

case	NREL	RELA	RELA+SO
Ag			
Au			

Exercise 11: optical properties

- Optical properties: fcc Al (either in w2web or on the command line)
  - *a_0=4.05 Å*
  - *init_lapw (use 10000 k-points!)*
  - *run_lapw*
  - *calculate optics (as described in the optics lecture, compare with the Al - Fig.)*
    - calculate plasma frequency (case.outputjoint) and dielectric function
    - check your results with respect to k-mesh
      - x kgen (check k-mesh dependency with k-mesh=k-mesh$_{old}$*2)
      - x lapw1 ; x lapw2 –fermi
      - x optic, x joint, x kram # this creates default inputs. Edit them for your needs
      - x optic, x joint, x kram
      - opticplot
- Optical properties: fcc Ag and Au (both have a_0=4.08 Å)
  - NREL calc. (change RELA to NREL in case.struct after init_lapw), do the scf and optics (save nrel; save nrel -optic
  - scalar-relativistic calc. (NREL → RELA), do the scf and optics (save rela)
  - include spin-orbit: init_so; run_lapw -so (case.inso **without RLOs** since optic does not support RLOs; put large Emax in case.in1); optics (save rela-so)

Ag and Au: a relativistic effect



Exercise 12: Valence PES of PbO_2

- Learn valence photoelectron spectroscopy calculations with WIEN2k
- Learn how to renormalize the PDOS and decompose the interstital DOS

- Questions: Compare spectra with lecture notes !
  - *Note the change of cross sections with excitation energy. Fill the following table:*

Energy	Pb-6s	Pb-6p	Pb-5d	O-2s	O-2p
56					
1487					
7799					

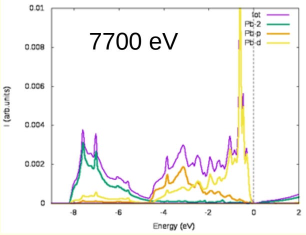
  - *Note the different localization of the atomic orbitals. Put the **fraction of charge** into the Table:*

Pb-6s	Pb-6p	Pb-5d	O-2s	O2p

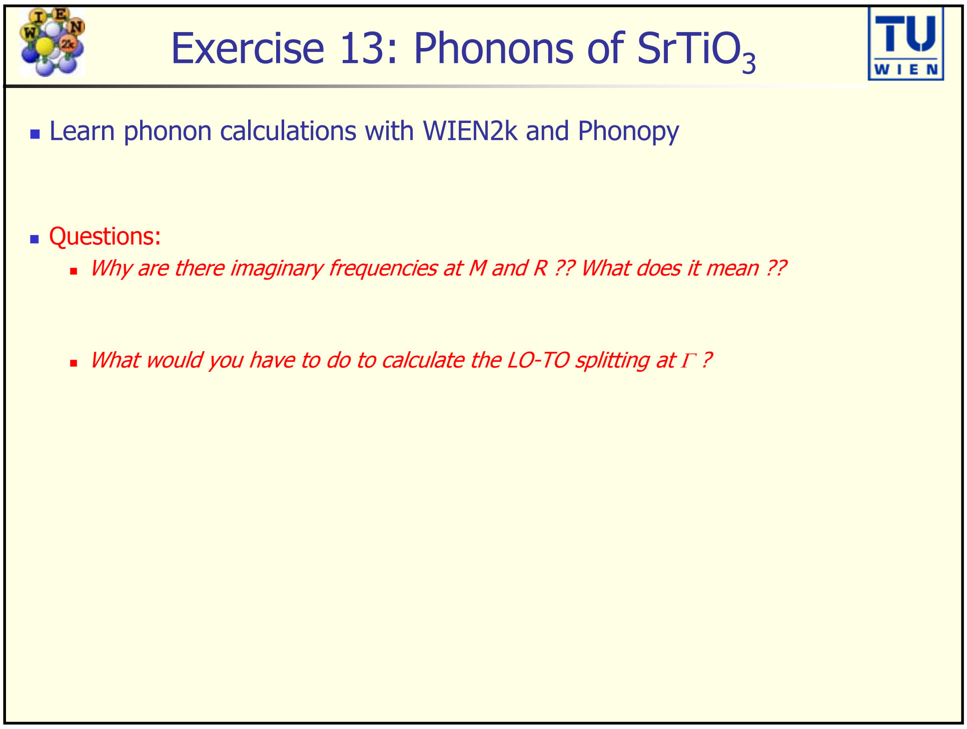
  - *Note the difference in the PDOS and renormalized PDOS*
  - *PS: The cross section of Pb-5d is a bit overestimated, since the Pb-d states close to EF have already 6d character (smaller cross section)*

Exercise 12: Valence PES of PbO_2

- PbO2: SG 136 (P42/mnm), a= 9.37, c=6.42 bohr; Pb(0,0,0), O(.307,.307,0)
- mkdir ~/WIEN2k/PbO2; cd ~/WIEN2k/PbO2
- makestruct
- cp init.struct PbO2.struct
- init -b
- run
- x lapw2 -qtl
- configure_int   with total, Pb-s,p,d and O-s,p DOS
- x tetra
- x pes
  - *Run and then plot it for 3 different excitation energies: 56, 1486 and 7700 eV*
  - *Press "enter" for all other questions*



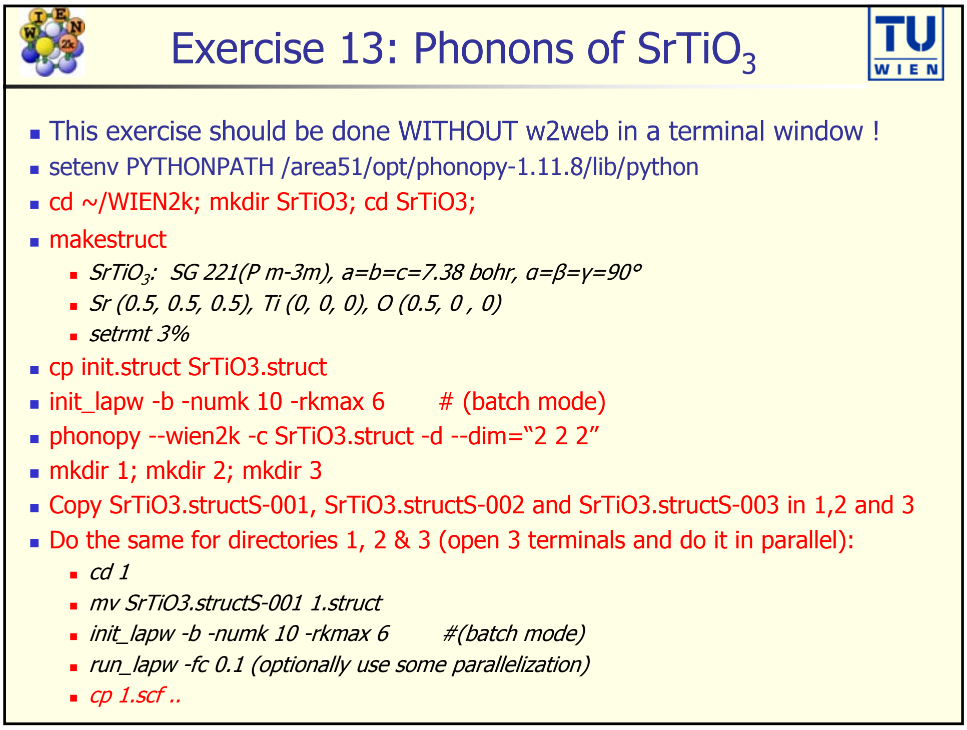
- Plot the PDOS, the renormalized PDOS and the PES (for all 3 energies).
- dosplot2   # dosplot2 -ren   # dosplot2 -pes
- Nicer plots are obtained when the data in case.dos1ev, case.dosrn1ev, case.pes1 are broadened by some Gaussian.



Exercise 13: Phonons of $SrTiO_3$

- Learn phonon calculations with WIEN2k and Phonopy

- Questions:
  - *Why are there imaginary frequencies at M and R ?? What does it mean ??*
  - *What would you have to do to calculate the LO-TO splitting at Γ ?*

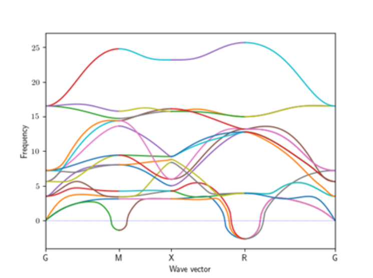


Exercise 13: Phonons of $SrTiO_3$

- This exercise should be done WITHOUT w2web in a terminal window !
- setenv PYTHONPATH /area51/opt/phonopy-1.11.8/lib/python
- cd ~/WIEN2k; mkdir SrTiO3; cd SrTiO3;
- makestruct
  - *$SrTiO_3$: SG 221(P m-3m), a=b=c=7.38 bohr, α=β=γ=90°*
  - *Sr (0.5, 0.5, 0.5), Ti (0, 0, 0), O (0.5, 0 , 0)*
  - *setrmt 3%*
- cp init.struct SrTiO3.struct
- init_lapw -b -numk 10 -rkmax 6 # (batch mode)
- phonopy --wien2k -c SrTiO3.struct -d --dim="2 2 2"
- mkdir 1; mkdir 2; mkdir 3
- Copy SrTiO3.structS-001, SrTiO3.structS-002 and SrTiO3.structS-003 in 1,2 and 3
- Do the same for directories 1, 2 & 3 (open 3 terminals and do it in parallel):
  - *cd 1*
  - *mv SrTiO3.structS-001 1.struct*
  - *init_lapw -b -numk 10 -rkmax 6 #(batch mode)*
  - *run_lapw -fc 0.1 (optionally use some parallelization)*
  - *cp 1.scf ..*

Phonons continued

- phonopy --wien2k -f 1.scf 2.scf 3.scf
- create band.conf with editor, containing the following information:
  - *ATOM_NAME = Sr Ti O*
  - *DIM = 2 2 2*
  - *PRIMITIVE_AXIS = 1.0 0.0 0.0 0.0 1.0 0.0 0.0 0.0 1.0*
  - *BAND = 0 0 0 1/2 1/2 0 0 1/2 0 1/2 1/2 1/2 0 0 0*
  - *BAND_LABELS = G   M X R G*
  - *BAND_CONNECTION = .TRUE.*
- phonopy --wien2k -c SrTiO3.struct band.conf –p
- save with: phonopy --wien2k -c SrTiO3.struct band.conf –p –s #(band.pdf)
- your result should be like:



(Remove window and open a new one,
otherwise the other Python examples will not work)

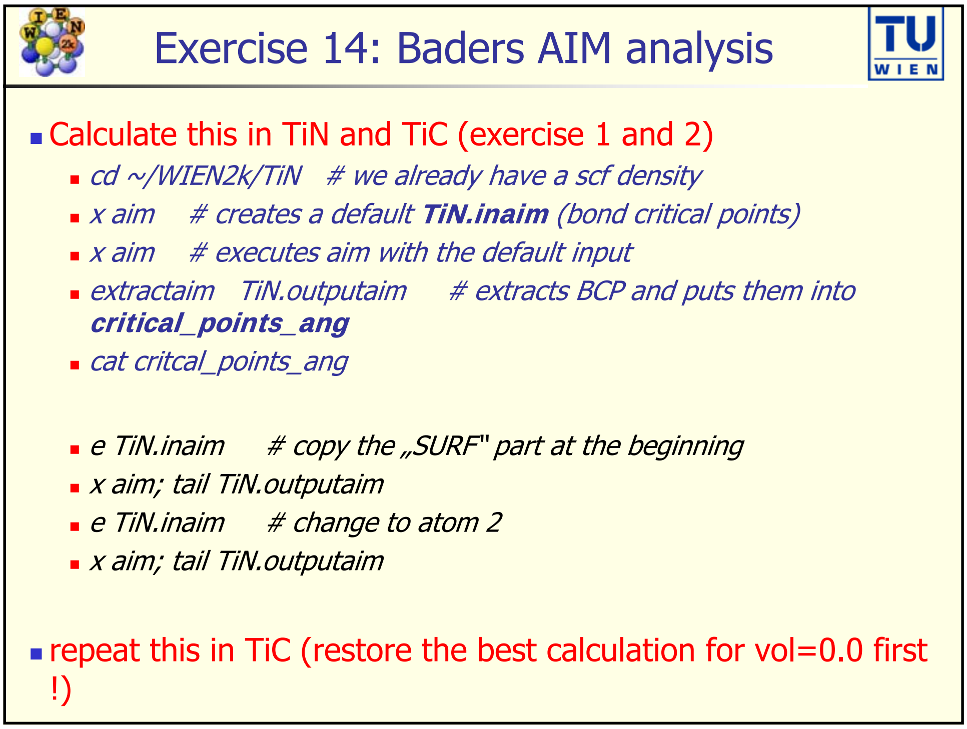
Exercise 14: Baders AIM analysis

- Learn how to calculate “Atoms in Molecules” properties (bond critical points, atomic basins, atomic charges) with the aim and the critic2 (unsupported software) tool

- Questions:
  - *How many different bond-   ring-   and cage -   critical points do we have in TiN and TiC?*

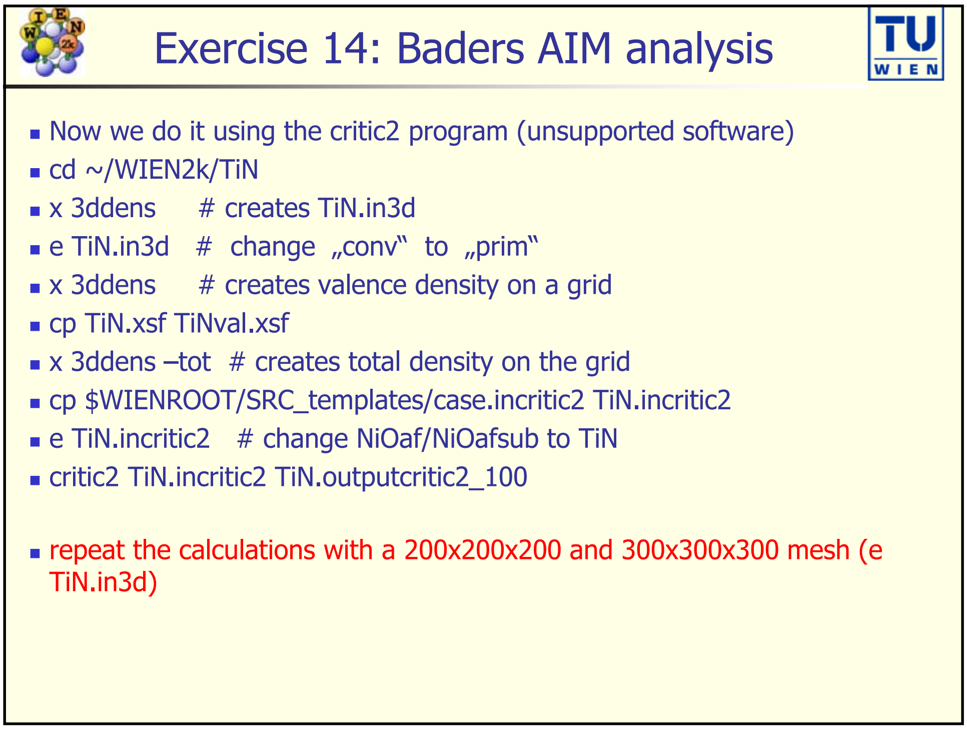
  - *What are the Ti (N,C) Bader charges and atomic volumes in TiN and TiC using aim and critic2 /100/200/300)?*

	Vol - aim	Vol-crit 100/200/300	Q(Ti) aim	Q(Ti)-(crit) 100/200/300	Q(X) aim	Q(X)-crit 100/200/300
Ti(TiN)						
Ti(TiC)						
N(TiN)						
C(TiC)						



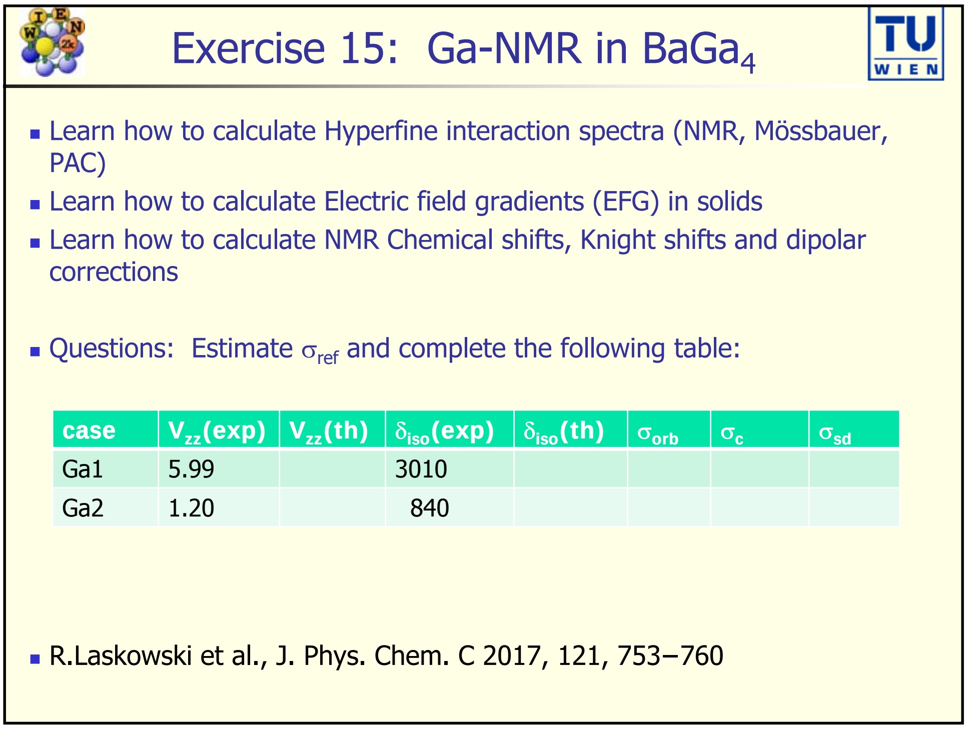
Exercise 14: Baders AIM analysis

- Calculate this in TiN and TiC (exercise 1 and 2)
  - *cd ~/WIEN2k/TiN # we already have a scf density*
  - *x aim # creates a default **TiN.inaim** (bond critical points)*
  - *x aim # executes aim with the default input*
  - *extractaim TiN.outputaim # extracts BCP and puts them into **critical_points_ang***
  - *cat critcal_points_ang*
  - *e TiN.inaim # copy the „SURF‟ part at the beginning*
  - *x aim; tail TiN.outputaim*
  - *e TiN.inaim # change to atom 2*
  - *x aim; tail TiN.outputaim*
- repeat this in TiC (restore the best calculation for vol=0.0 first !)



Exercise 14: Baders AIM analysis

- Now we do it using the critic2 program (unsupported software)
- cd ~/WIEN2k/TiN
- x 3ddens # creates TiN.in3d
- e TiN.in3d # change „conv‟ to „prim‟
- x 3ddens # creates valence density on a grid
- cp TiN.xsf TiNval.xsf
- x 3ddens –tot # creates total density on the grid
- cp $WIENROOT/SRC_templates/case.incritic2 TiN.incritic2
- e TiN.incritic2 # change NiOaf/NiOafsub to TiN
- critic2 TiN.incritic2 TiN.outputcritic2_100
- repeat the calculations with a 200x200x200 and 300x300x300 mesh (e TiN.in3d)

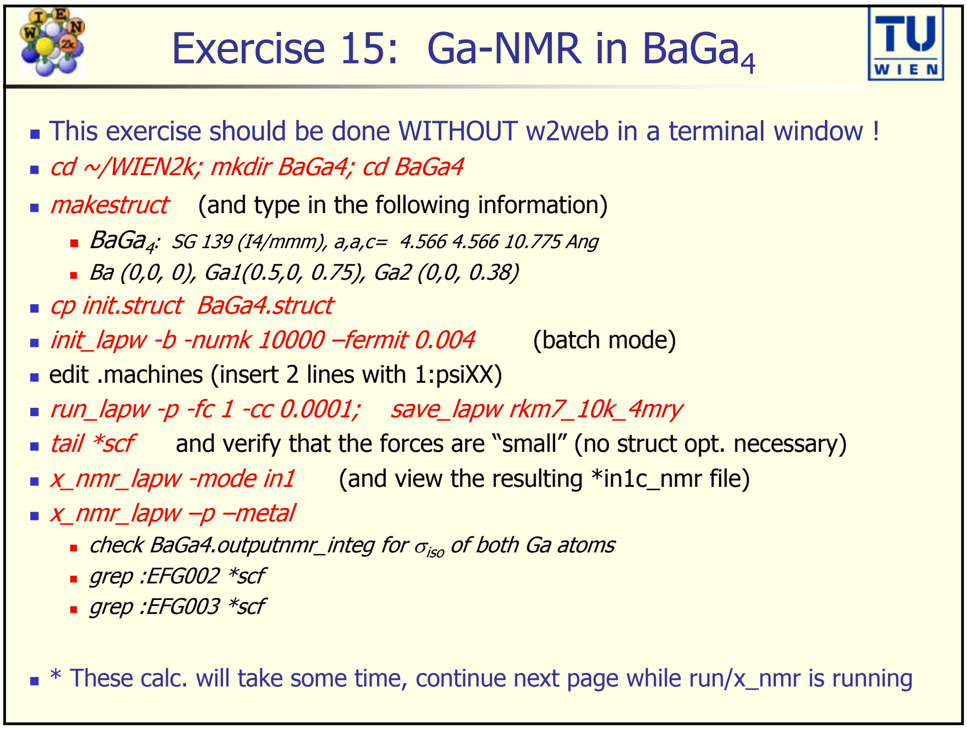


Exercise 15: Ga-NMR in $BaGa_4$

- Learn how to calculate Hyperfine interaction spectra (NMR, Mössbauer, PAC)
- Learn how to calculate Electric field gradients (EFG) in solids
- Learn how to calculate NMR Chemical shifts, Knight shifts and dipolar corrections
- Questions: Estimate σ_{ref} and complete the following table:

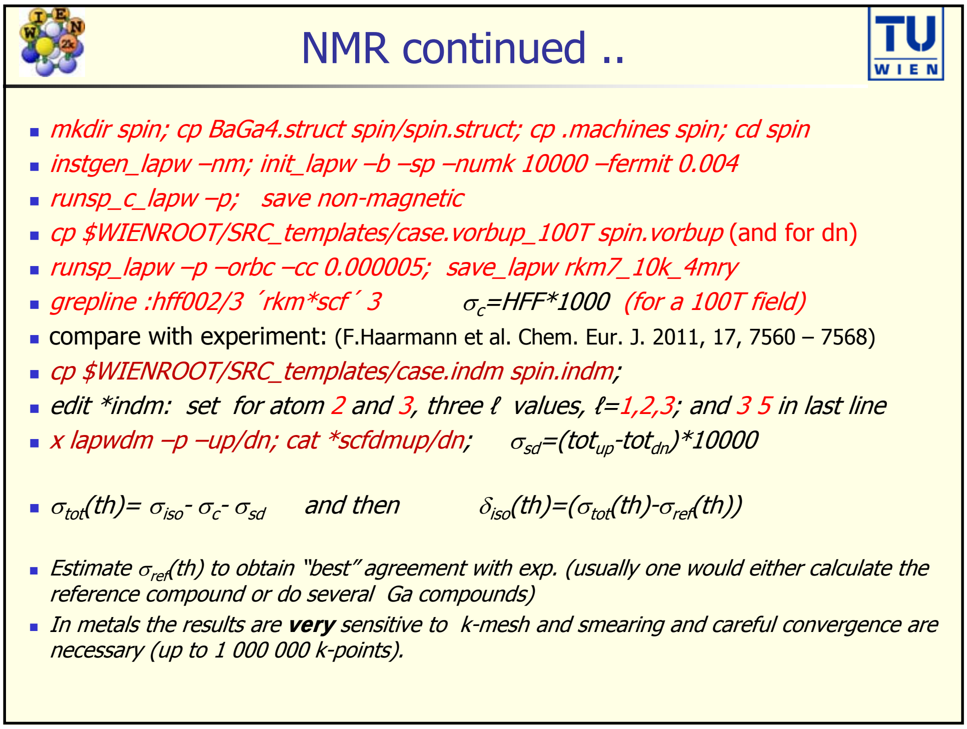
case	V_{zz}(exp)	V_{zz}(th)	δ_{iso}(exp)	δ_{iso}(th)	σ_{orb}	σ_c	σ_{sd}
Ga1	5.99		3010				
Ga2	1.20		840				

- R.Laskowski et al., J. Phys. Chem. C 2017, 121, 753−760



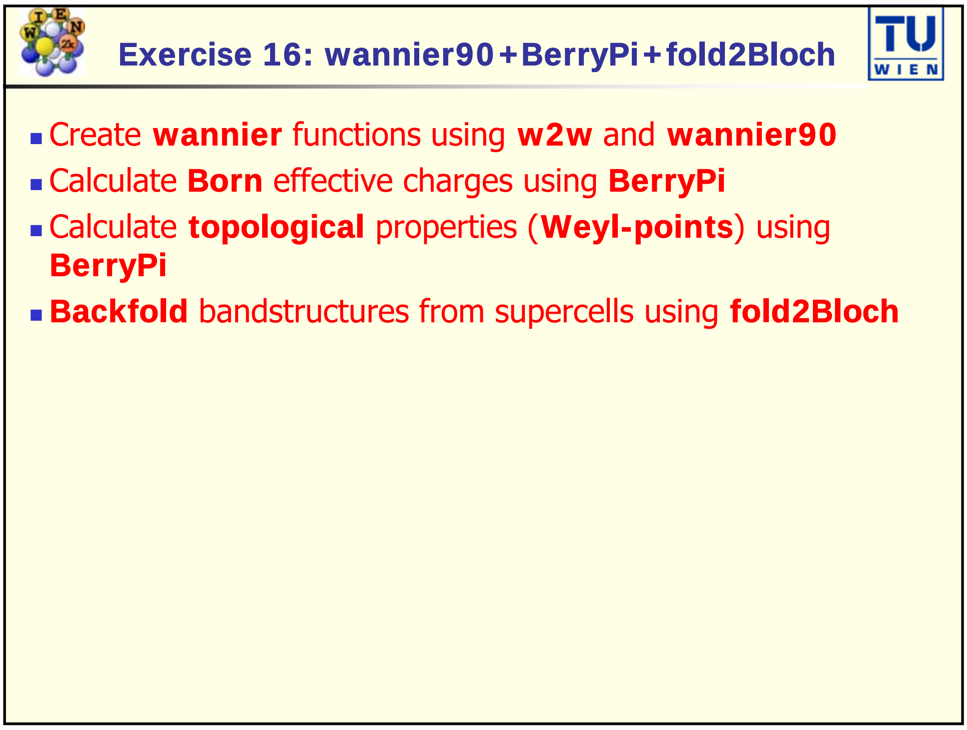
Exercise 15: Ga-NMR in $BaGa_4$

- This exercise should be done WITHOUT w2web in a terminal window !
- *cd ~/WIEN2k; mkdir BaGa4; cd BaGa4*
- *makestruct* (and type in the following information)
  - *$BaGa_4$: SG 139 (I4/mmm), a,a,c= 4.566 4.566 10.775 Ang*
  - *Ba (0,0, 0), Ga1(0.5,0, 0.75), Ga2 (0,0, 0.38)*
- *cp init.struct BaGa4.struct*
- *init_lapw -b -numk 10000 –fermit 0.004* (batch mode)
- edit .machines (insert 2 lines with 1:psiXX)
- *run_lapw -p -fc 1 -cc 0.0001; save_lapw rkm7_10k_4mry*
- *tail *scf* and verify that the forces are "small" (no struct opt. necessary)
- *x_nmr_lapw -mode in1* (and view the resulting *in1c_nmr file)
- *x_nmr_lapw –p –metal*
  - *check BaGa4.outputnmr_integ for σ_{iso} of both Ga atoms*
  - *grep :EFG002 *scf*
  - *grep :EFG003 *scf*
- * These calc. will take some time, continue next page while run/x_nmr is running



NMR continued ..

- *mkdir spin; cp BaGa4.struct spin/spin.struct; cp .machines spin; cd spin*
- *instgen_lapw –nm; init_lapw –b –sp –numk 10000 –fermit 0.004*
- *runsp_c_lapw –p; save non-magnetic*
- *cp $WIENROOT/SRC_templates/case.vorbup_100T spin.vorbup* (and for dn)
- *runsp_lapw –p –orbc –cc 0.000005; save_lapw rkm7_10k_4mry*
- *grepline :hff002/3 ´rkm*scf´ 3* σ_c*=HFF*1000 (for a 100T field)*
- compare with experiment: (F.Haarmann et al. Chem. Eur. J. 2011, 17, 7560 – 7568)
- *cp $WIENROOT/SRC_templates/case.indm spin.indm;*
- *edit *indm: set for atom 2 and 3, three ℓ values, ℓ=1,2,3; and 3 5 in last line*
- *x lapwdm –p –up/dn; cat *scfdmup/dn;* $\sigma_{sd}=(tot_{up}-tot_{dn})*10000$
- $\sigma_{tot}(th)= \sigma_{iso}- \sigma_c- \sigma_{sd}$ *and then* $\delta_{iso}(th)=(\sigma_{tot}(th)-\sigma_{ref}(th))$
- *Estimate* $\sigma_{ref}(th)$ *to obtain "best" agreement with exp. (usually one would either calculate the reference compound or do several Ga compounds)*
- *In metals the results are **very** sensitive to k-mesh and smearing and careful convergence are necessary (up to 1 000 000 k-points).*



Exercise 16: wannier90+BerryPi+fold2Bloch

- Create **wannier** functions using **w2w** and **wannier90**
- Calculate **Born** effective charges using **BerryPi**
- Calculate **topological** properties (**Weyl-points**) using **BerryPi**
- **Backfold** bandstructures from supercells using **fold2Bloch**

GaAs -- MLWF

+

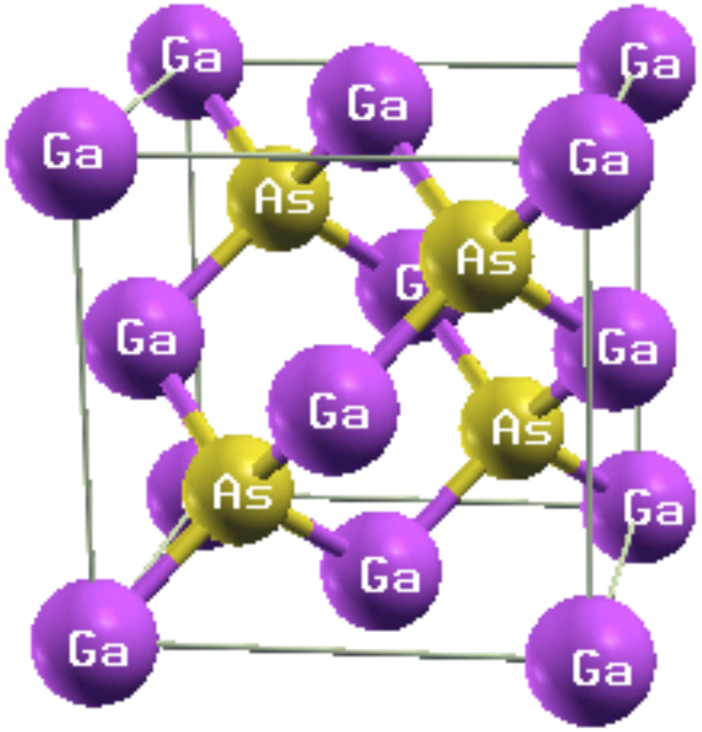
+

WANNIER90

Special thanks to Elias Assmann (TU Graz)
for the generous help in
preparation of this tutorial

YouTube video: https://youtu.be/R4c1YHDh3GE

1. Wien2k SCF



Create a tutorial directory, e.g.

```
$ mkdir GaAs-MLWF
```

Create the structure file using the following parameters:

2 atoms per primitive unit cell (Ga, As)

Lattice "F" = f.c.c.

Lattice parameters $a_0 = b_0 = c_0$ = 10.683 Bohr

Positions: "0 0 0" for Ga and "1/4 1/4 1/4" for As; RMT's - automatic

You can use `xcrysden` to view the structure

```
$ xcrysden --wien_struct GaAs-MLWF.struct
```

Initialize Wien2k calculation (LDA, ~600 k-points ≡ 8x8x8 mesh)

```
$ init_lapw -b -vxc 5 -numk 600
```

Run regular SCF calculation using default convergence criteria

```
$ run_lapw
```

After SCF cycle is completed (~8 iterations). We proceed with the band structure

Prepare the list of k points to be used for the band structure plot (`GaAs-MLWF.klist_band` file) using `xcrysden`

```
xcrysden File > Open Wien2k
   > Select k-path
```

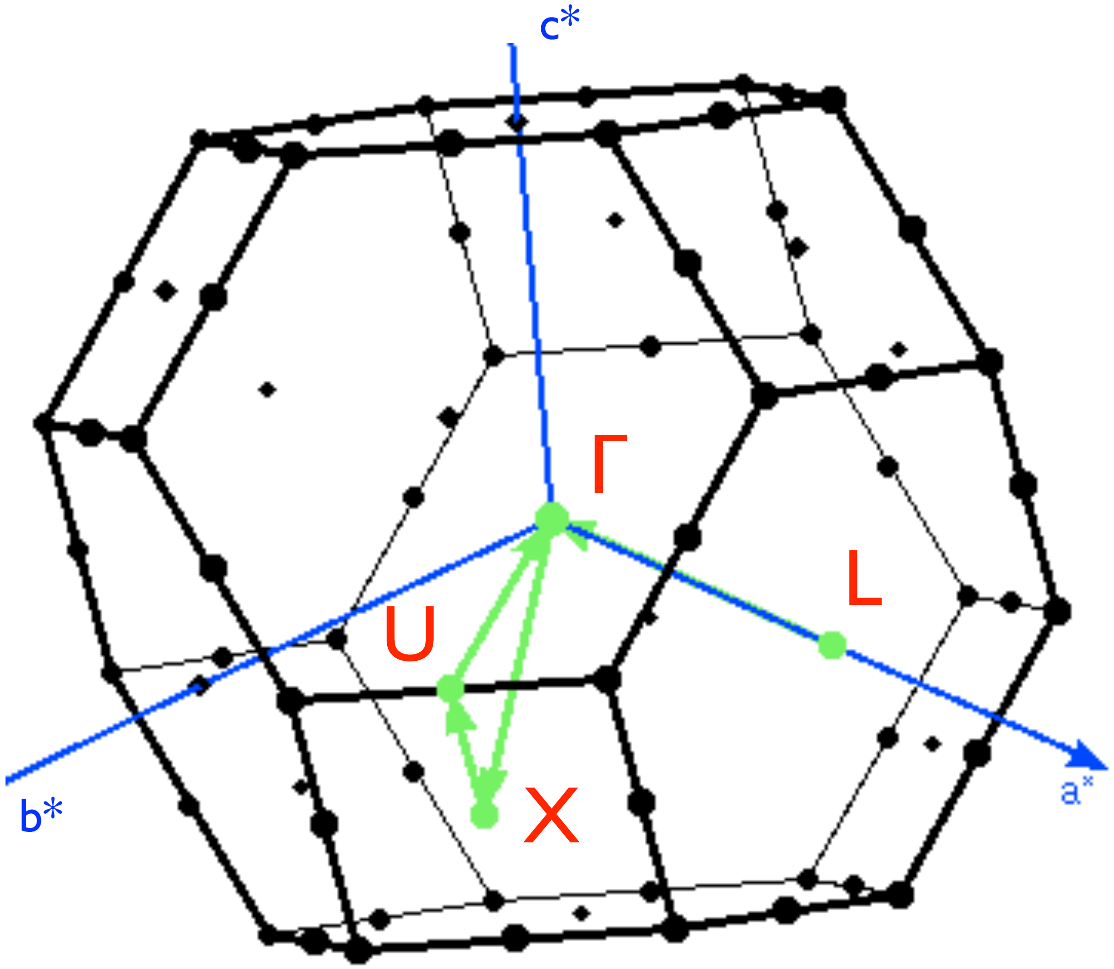
Select points L(1/2 0 0), Γ(0 0 0), X(1/2 1/2 0), (5/8 5/8 1/4), Γ

Set the total of 100 k-points along the path.

Save the list as `GaAs-MLWF.klist_band`

Solve eigenvalue problem on the k-path

```
$ x lapw1 -band
```



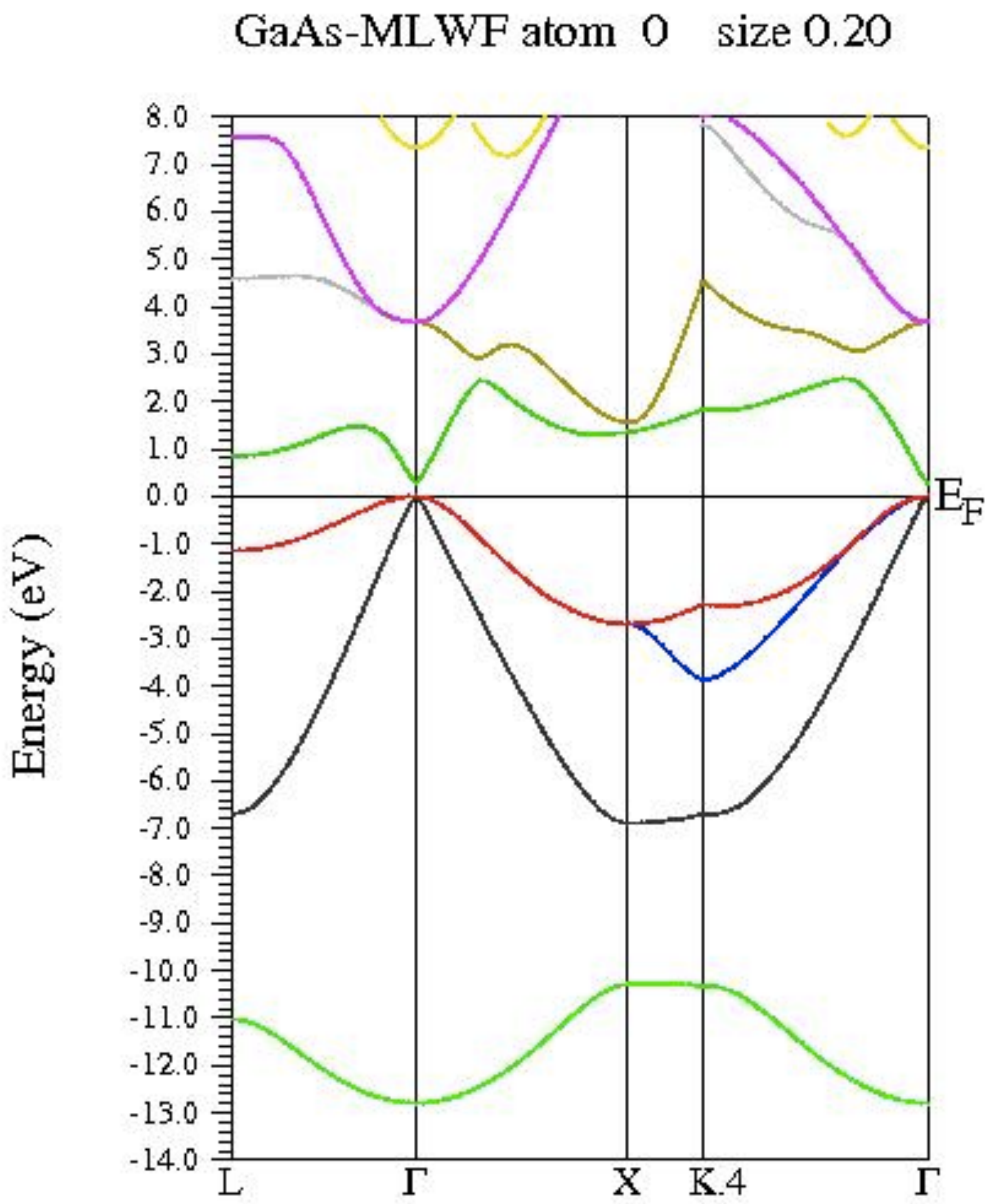
Get the Fermi energy

```
$ grep :FER *scf
```

For the band structure plot we will use the web interface (w2web). Create a new session and navigate to the current work directory.

```
w2web Tasks > Bandstructure
w2web Select
      "Edit GaAs-MLWF.insp",
      insert the Fermi energy,
      save
w2web x spaghetti
w2web plot band structure
```

Your band structure will be similar to the one shown on the right. Our aim is to construct Wannier functions that reproduce this band structure including valence and some conduction bands.



Before we proceed it is useful to determine the band indices for the region of interest

```
$ grep :BAN *scf2
:BAN00004:   4   -2.275995   -2.275442  2.00000000
:BAN00005:   5   -2.275995   -2.275261  2.00000000
:BAN00006:   6   -0.784989   -0.777234  2.00000000
:BAN00007:   7   -0.777234   -0.772519  2.00000000
:BAN00008:   8   -0.777234   -0.772371  2.00000000
:BAN00009:   9   -0.772654   -0.770305  2.00000000
:BAN00010:  10   -0.771540   -0.769389  2.00000000
:BAN00011:  11   -0.625463   -0.437790  2.00000000
:BAN00012:  12   -0.191045    0.314670  2.00000000
:BAN00013:  13    0.029025    0.314670  2.00000000
:BAN00014:  14    0.067108    0.314670  2.00000000
:BAN00015:  15    0.335212    0.647926  0.00000000
:BAN00016:  16    0.429016    0.720389  0.00000000
:BAN00017:  17    0.585553    1.053515  0.00000000
:BAN00018:  18    0.585553    1.053515  0.00000000
:BAN00019:  19    0.853994    1.118413  0.00000000
```

d-orb. of As and Ga (do not participate in bonding)

bonding + antibonding orbitals

⇧ Emin (Ry)

⇧ Emax

⇧ occupancy

2. Construction of Wannier functions

Prepare a separate directory

```
$ prepare_w2wdir GaAs-MLWF GaAs-WANN
$ cd GaAs-WANN
```

Initialize Wien2Wannier

```
$ init_w2w
```

Select 8x8x8 k-mesh (unshifted);

energy range (eV) -13 10 (this is not very critical);

band indices [Nmin Nmax] 11 18 (see the previous page);

for the projection we choose "1:s,p" and "2:s,p" (1 = Ga, 2 = As)

Get the vector file on the full Brillouin zone mesh

```
$  x lapw1
```

Compute matrix elements needed for Wannier90

```
$  x w2w
```

Run Wannier90

```
$  x wannier90
```

Verify the output

spread $\langle \Delta \mathbf{r}^2 \rangle$

⇓

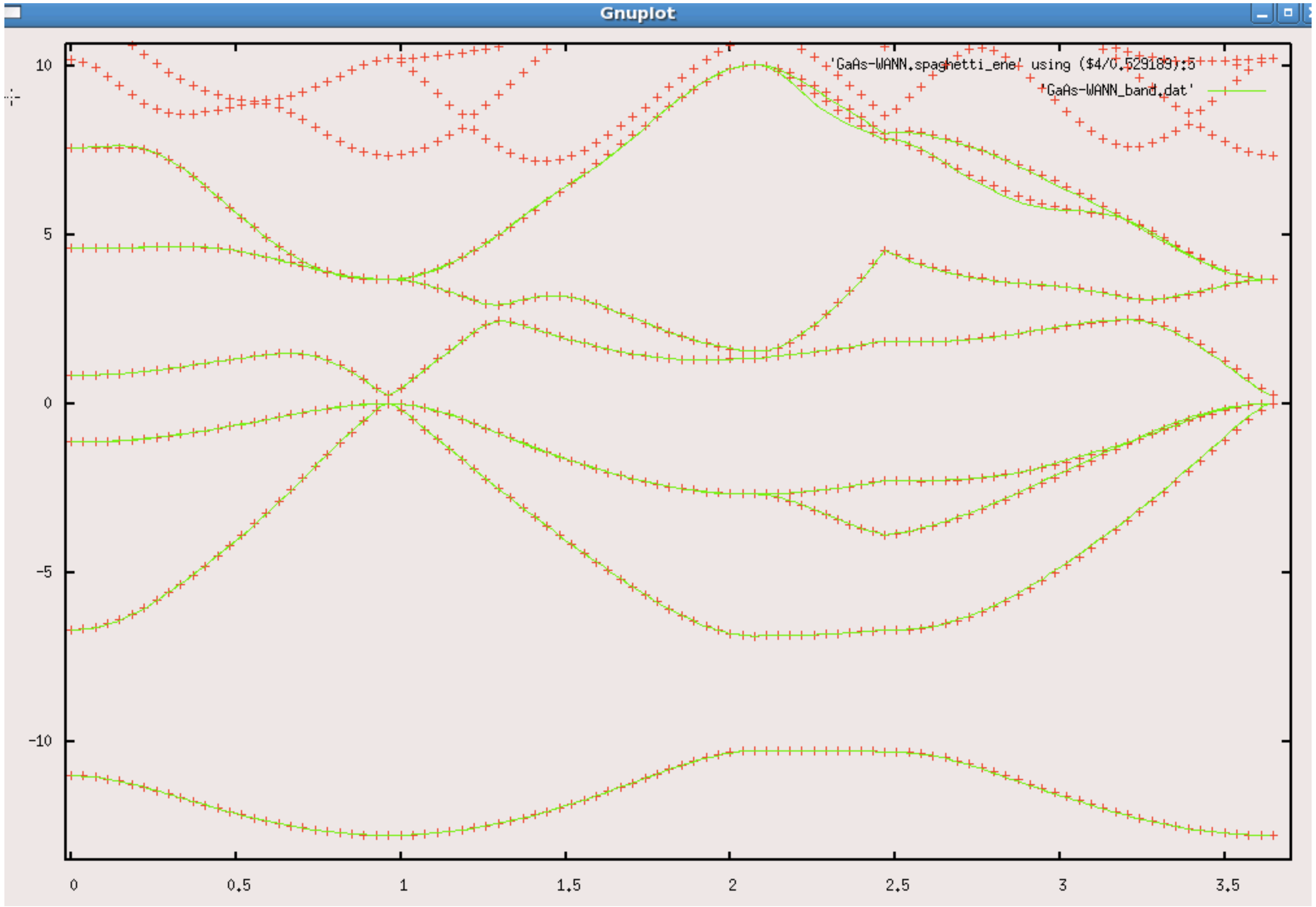
```
$  less GaAs-WANN.wout
...
Final State
  WF centre and spread    1  (  0.000000, -0.000000, -0.000000 )     1.91981243
  WF centre and spread    2  ( -0.000000, -0.000000,  0.000000 )     5.86945318
  WF centre and spread    3  ( -0.000000,  0.000000, -0.000000 )     5.86945318
  WF centre and spread    4  (  0.000000,  0.000000,  0.000000 )     5.86945318
  WF centre and spread    5  (  1.413299, -1.413299, -1.413299 )     1.61179550
  WF centre and spread    6  (  1.413301, -1.413300, -1.413300 )     3.82462632
  WF centre and spread    7  (  1.413300, -1.413301, -1.413300 )     3.82462632
  WF centre and spread    8  (  1.413300, -1.413300, -1.413301 )     3.82462632
...
```

There you can see the position and spread of the WF’s, how they changed in the course of convergence. WF’s 1-4 are all positioned at the origin (atom 1), WF’s 5-8 are centred at the 2nd atom (please check the coordinates)

3. Post-processing

Plot the band structure

```
$  gnuplot
gnuplot> plot 'GaAs-WANN.spaghetti_ene' using
   ($4/0.529189):5, 'GaAs-WANN_band.dat' with lines
```



+ original Wien2k band structure

– Band structure computed from Wannier functions

Plotting WF’s (can take a while). Get the template of an input file

```
$ cp $WIENROOT/SRC_templates/case.inwplot
  GaAs-WANN.inwplot
```

Edit “GaAs-WANN.inwplot”

Select origin “`-1  -1  -1  1`” and axis x, y, z
“ 1 -1 -1 1”
“-1 1 -1 1”
“-1 -1 1 1”
grid point mesh: 30 30 30
“1” for the Wannier function index

Compute the 1st Wannier function on the mesh chosen

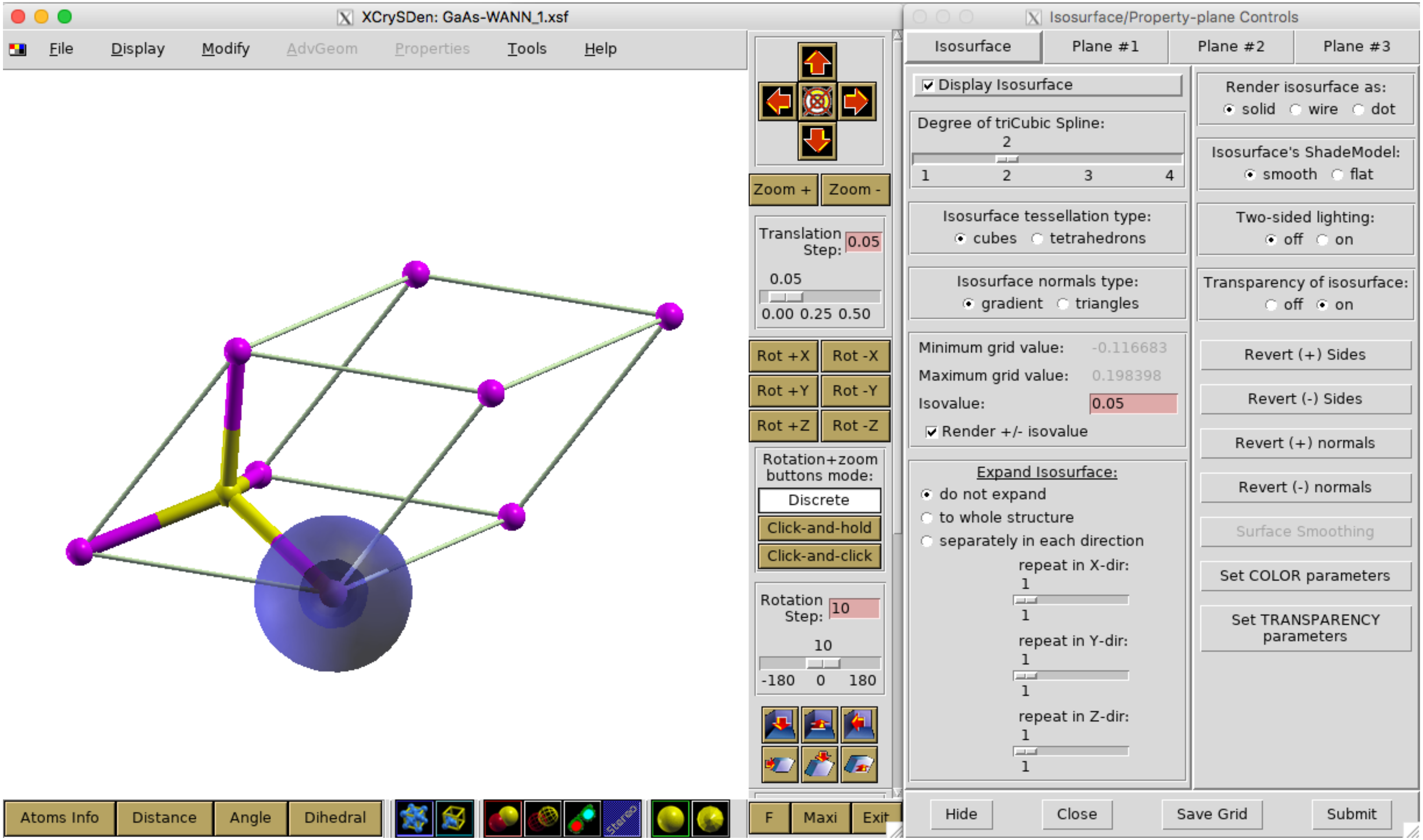
```
$ x wplot -wf 1
```

If you need to plot any other WF’s (2, 3, etc), just edit the option.

Convert the output of `wplot` into `xcrysden` format for plotting.

```
$ wplot2xsf
```

Visualize with `xcrysden` (instructions on the next page)



XCrySDen: GaAs-WANN_1.xsf

Translation Step: 0.05

Rotation+zoom buttons mode: Discrete

Rotation Step: 10

Isosurface/Property-plane Controls

☑ Display Isosurface

Degree of triCubic Spline: 2

Isosurface tessellation type: ● cubes ○ tetrahedrons

Isosurface normals type: ● gradient ○ triangles

Minimum grid value: -0.116683

Maximum grid value: 0.198398

Isovalue: 0.05

☑ Render +/- isovalue

Expand Isosurface:

● do not expand

○ to whole structure

○ separately in each direction

repeat in X-dir: 1

repeat in Y-dir: 1

repeat in Z-dir: 1

Render isosurface as: ● solid ○ wire ○ dot

Isosurface's ShadeModel: ● smooth ○ flat

Two-sided lighting: ● off ○ on

Transparency of isosurface: ○ off ● on

```
$  xcrysden --xsf GaAs-WANN_1.xsf
```

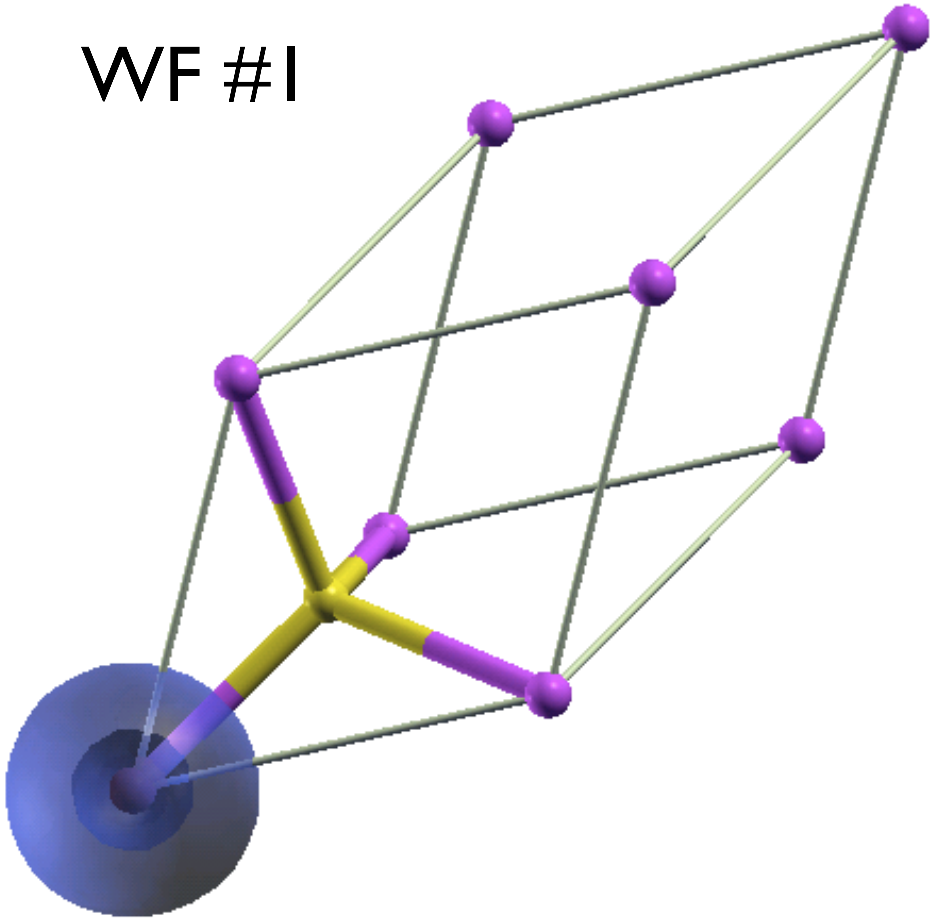
```
xcrysden Tools > Data Grid > OK
```

Set "Isovalue:" and check box "render +/- isovalue" (see screenshot on the previous page)

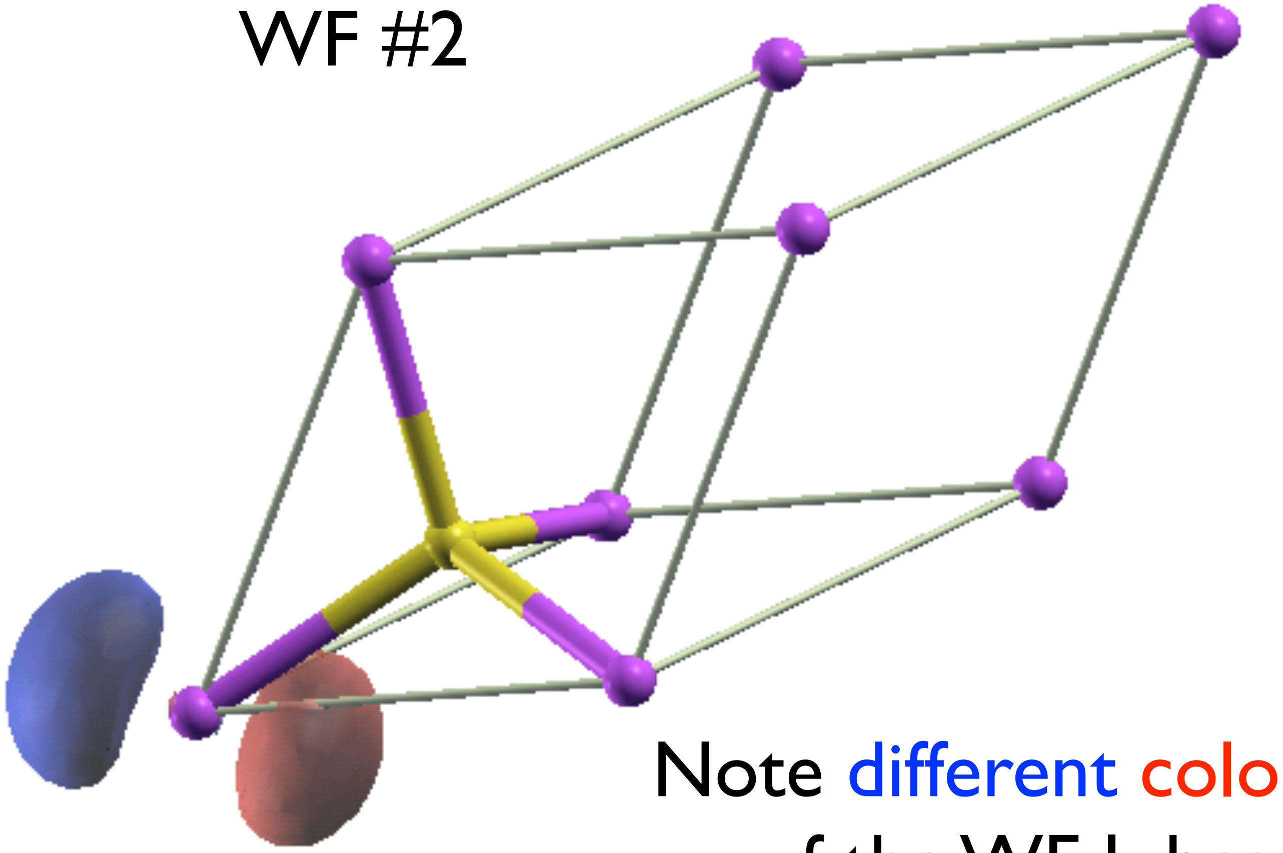
Play with the settings. You will get a spherical (s-like) WF centred at the origin.

The second WF resembles a p-orbital (you can get it by editing "GaAs-WANN.inwplot", re-run "`x wplot`" and "`wplot2xsf`"). The new file should be called **`GaAs-WANN_2.xsf`**

WF #1

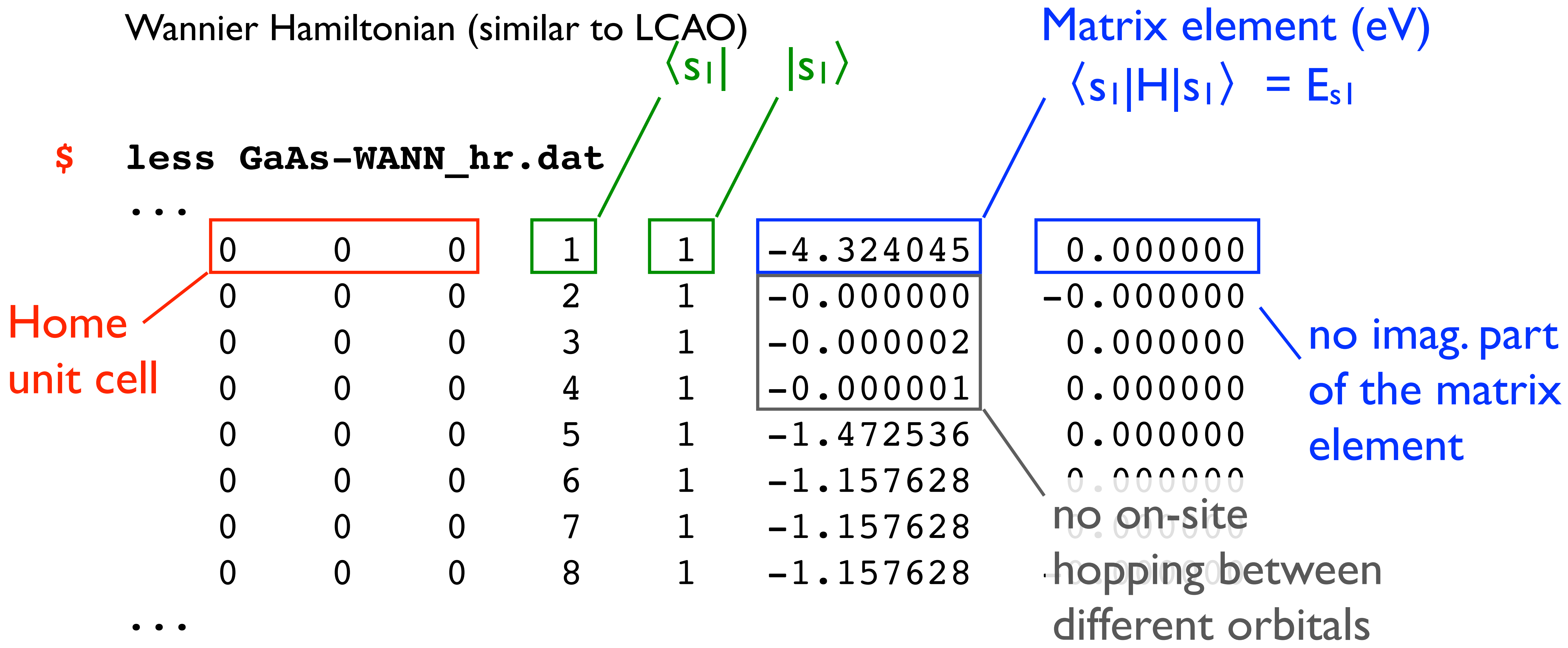


WF #2



Note different colours of the WF lobes

Wannier Hamiltonian (similar to LCAO)



$\langle s_l|$

$|s_l\rangle$

Matrix element (eV)

$\langle s_l|H|s_l\rangle = E_{sl}$

```
$   less GaAs-WANN_hr.dat
 ...
      0      0      0      1      1     -4.324045      0.000000
      0      0      0      2      1     -0.000000     -0.000000
      0      0      0      3      1     -0.000002      0.000000
      0      0      0      4      1     -0.000001      0.000000
      0      0      0      5      1     -1.472536      0.000000
      0      0      0      6      1     -1.157628      0.000000
      0      0      0      7      1     -1.157628      0.000000
      0      0      0      8      1     -1.157628      0.000000
 ...
```

Home unit cell

no imag. part of the matrix element

no on-site hopping between different orbitals

Determine on site energies E_s and E_p for Ga and As and compare them to those suggested by Harrison (note: only their relative differences are important). Ga-p electrons correspond to the "0 0 0 2 2" line, As-s can be found at "0 0 0 4 4", etc. Results for the energy differences from WF are approximately 20-30% greater that Harrison's data. They should not agree exactly as WFs include matrix elements beyond the first nearest neighbour.

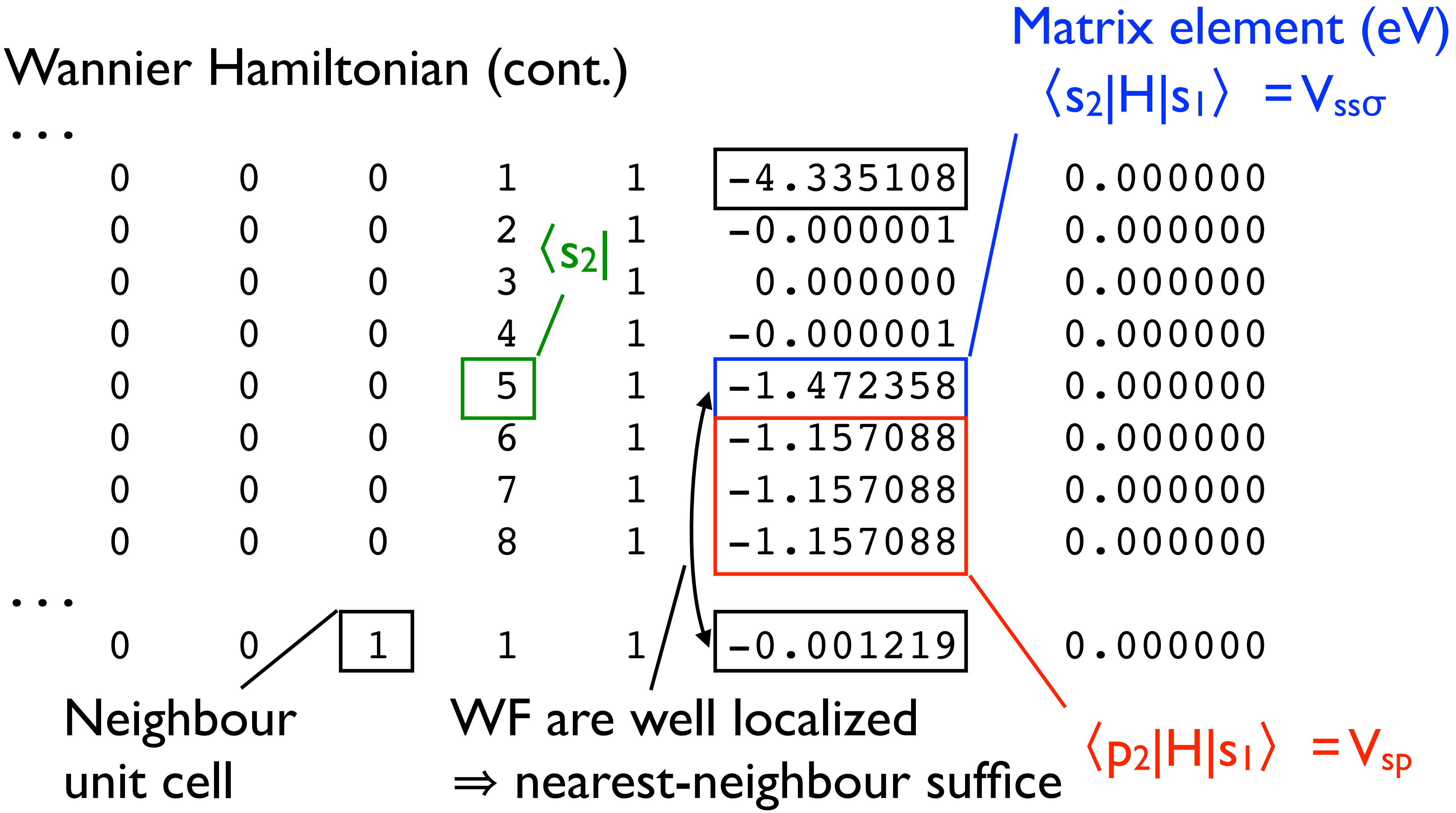
From Harrison's solid state tables:

$E_p(Ga) - E_s(Ga) = 5.9$ eV

$E_p(As) - E_s(As) = 9.9$ eV

$E_p(Ga) - E_p(As) = 3.3$ eV

Wannier Hamiltonian (cont.)



Matrix element (eV)

$\langle s_2|H|s_1\rangle = V_{ss\sigma}$

```
...
    0     0     0     1     1    -4.335108     0.000000
    0     0     0     2     1    -0.000001     0.000000
    0     0     0     3     1     0.000000     0.000000
    0     0     0     4     1    -0.000001     0.000000
    0     0     0     5     1    -1.472358     0.000000
    0     0     0     6     1    -1.157088     0.000000
    0     0     0     7     1    -1.157088     0.000000
    0     0     0     8     1    -1.157088     0.000000
...
    0     0     1     1     1    -0.001219     0.000000
```

$\langle s_2|$

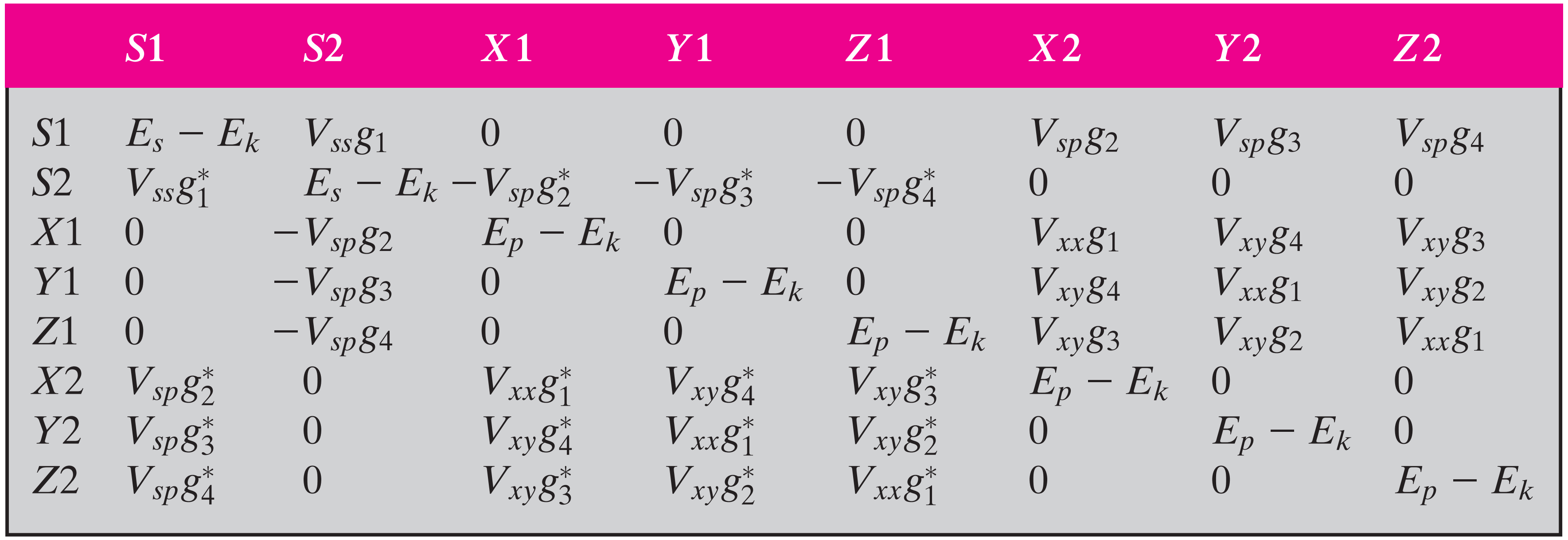
Neighbour unit cell

WF are well localized ⇒ nearest-neighbour suffice

$\langle p_2|H|s_1\rangle = V_{sp}$

Now you have all information required to build your *ab initio* TB sp3 Hamiltonian (Yu & Cardona)

Table 2.25. Matrix for the eight s and p bands in the diamond structure within the tight binding approximation

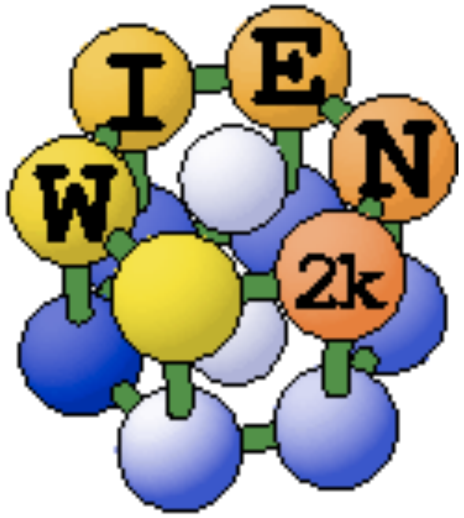


	$S1$	$S2$	$X1$	$Y1$	$Z1$	$X2$	$Y2$	$Z2$
$S1$	$E_s - E_k$	$V_{ss}g_1$	0	0	0	$V_{sp}g_2$	$V_{sp}g_3$	$V_{sp}g_4$
$S2$	$V_{ss}g_1^*$	$E_s - E_k$	$-V_{sp}g_2^*$	$-V_{sp}g_3^*$	$-V_{sp}g_4^*$	0	0	0
$X1$	0	$-V_{sp}g_2$	$E_p - E_k$	0	0	$V_{xx}g_1$	$V_{xy}g_4$	$V_{xy}g_3$
$Y1$	0	$-V_{sp}g_3$	0	$E_p - E_k$	0	$V_{xy}g_4$	$V_{xx}g_1$	$V_{xy}g_2$
$Z1$	0	$-V_{sp}g_4$	0	0	$E_p - E_k$	$V_{xy}g_3$	$V_{xy}g_2$	$V_{xx}g_1$
$X2$	$V_{sp}g_2^*$	0	$V_{xx}g_1^*$	$V_{xy}g_4^*$	$V_{xy}g_3^*$	$E_p - E_k$	0	0
$Y2$	$V_{sp}g_3^*$	0	$V_{xy}g_4^*$	$V_{xx}g_1^*$	$V_{xy}g_2^*$	0	$E_p - E_k$	0
$Z2$	$V_{sp}g_4^*$	0	$V_{xy}g_3^*$	$V_{xy}g_2^*$	$V_{xx}g_1^*$	0	0	$E_p - E_k$

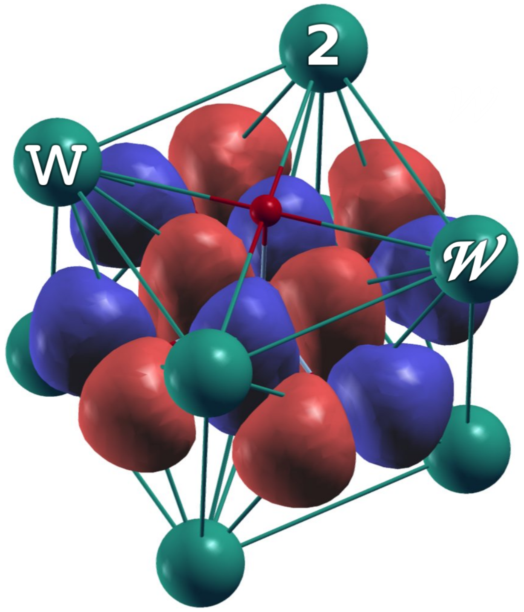
This tutorial was verified using

Wien2k 19.1
Wannier90 2.1.0
python 2.7.14
xcrysden 1.5.60

Born effective charge of GaN



+

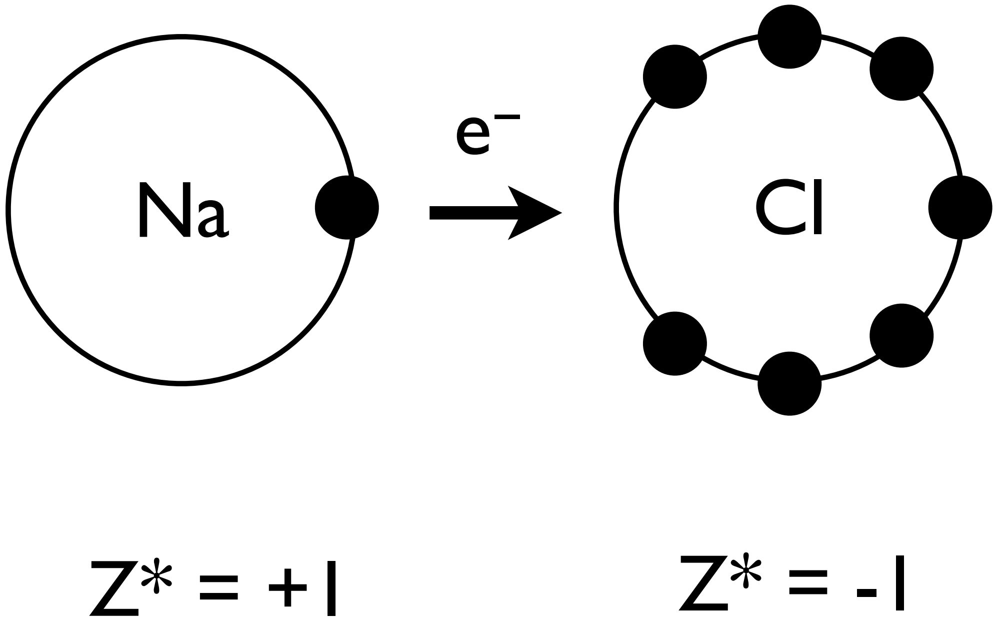


+ BerryPI

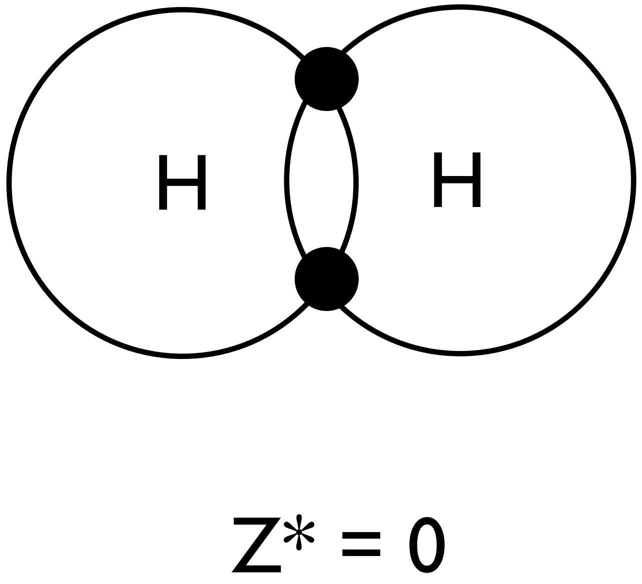
YouTube video: https://youtu.be/hLl9nKf35tA

Background

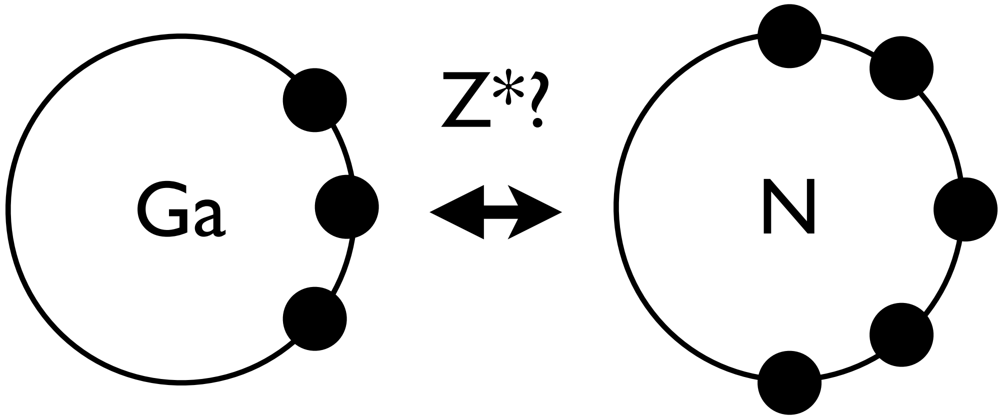
Ionic bond



Covalent bond



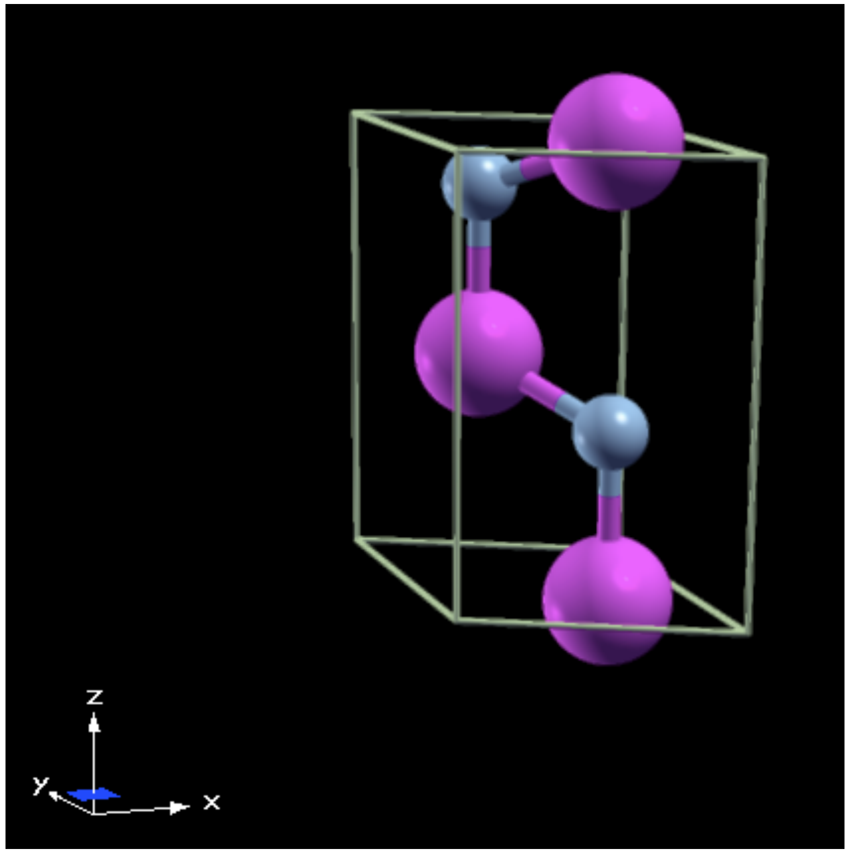
Mixed



Instructions

w2web Construct a structure file (`../GaN-W/GaN-W.struct`)

4-atoms (2-Ga, 2-N) per unit cell
Hexagonal lattice "H", α = β = 90°, γ = 120°
Cell size (Bohr): a = b = 5.963131; c = 9.722374
Coordinates:
Ga (2/3 1/3 0)
Ga (1/3 2/3 1/2)
N (2/3 1/3 0.376393)
N (1/3 2/3 0.876393)



Since we will introduce a displacement later, a small (~3%) RMT reduction is needed.

Initialize SCF calculation with LDA, RKMAX=7, 300 k points. The energy separation between core and valence electrons should be set at -8 Ry to include Ga-3p as valence electrons (otherwise a core leakage warning will be printed).

```
$   init_lapw -b -vxc 5 -ecut -8 -rkmax 7 -numk 300
```

It is worth to verify the structure with XCrysDen

Perform regular SCF calculation

```
$   run_lapw
```

Run Berry phase calculation

```
$  berrypi -k 8:8:4 | tee log
```

Note the ionic and electronic phases along Z-axis (wrapped [-π…+π])

Save the calculation

```
$  save_lapw -d Lambda0
```

Introduce small displacement on N-atoms

w2web Edit Z-coordinate of N atoms by adding 0.001 to the equilibrium value of the fractional coordinate u_z. Since nitrogen has 2 equivalent positions, both need to be updated. Think what is the reason for the need to shift both atoms in this case.

Repeat initialization, SCF and Berry phase calculation steps (note "-a" option used in order to update the log file, not overwrite)

```
$  init_lapw -b -vxc 5 -ecut -8 -rkmax 7 -numk 300

$  run_lapw

$  berrypi -k 8:8:4 | tee -a log
```

option "-a" will append output to the file instead of overriding it

Evaluate the total Berry phase for each of two calculations performed

$$\phi = \phi_{\mathrm{el}} + \phi_{\mathrm{ion}}$$

For the electronic phase use "Berry phase (rad) [-pi ... +pi] spin up+dn" value. For the ionic phase use "Total ionic phase wrap. (rad)" value.

Evaluate the phase change

$$\Delta\phi = \phi(\mathrm{perturbed}) - \phi(\mathrm{unperturbed})$$

It should be about -0.034 rad. Compute the effective charge Z* of Nitrogen in GaN using a Berry phases and the "shortcut" expression

$$Z^*_{ii} = \frac{\Delta\phi_i}{2\pi\Delta u_i}$$

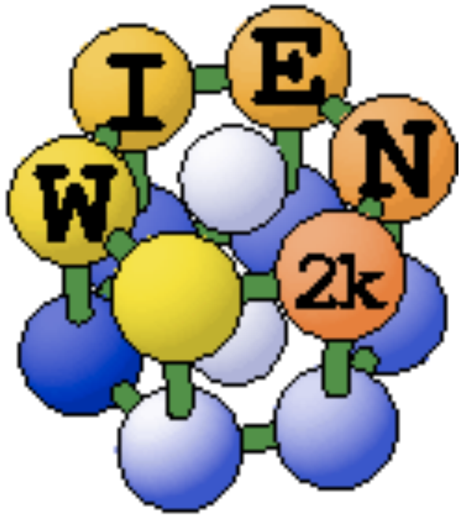
Here Δ*u* is the displacement in fractional coordinates. The equation applies to the case of one atom displaced. In our case, we need to take into account that 2 N-atoms were shifted.

Compare computed Z* with the literature value of -2.73 [Volume 44D of the series Landolt-Börnstein - Group III Condensed Matter pp 420-423, "GaN: effective charge, dielectric constants" by D. Strauch]

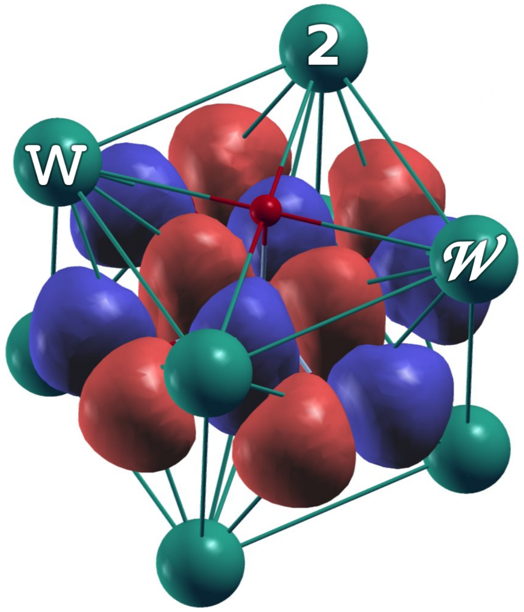
This tutorial was verified using

Wien2k 19.1
BerryPI (build Mar 08, 2019)
python 2.7.14
numpy 1.16.0

Spontaneous polarization $BaTiO_3$



+

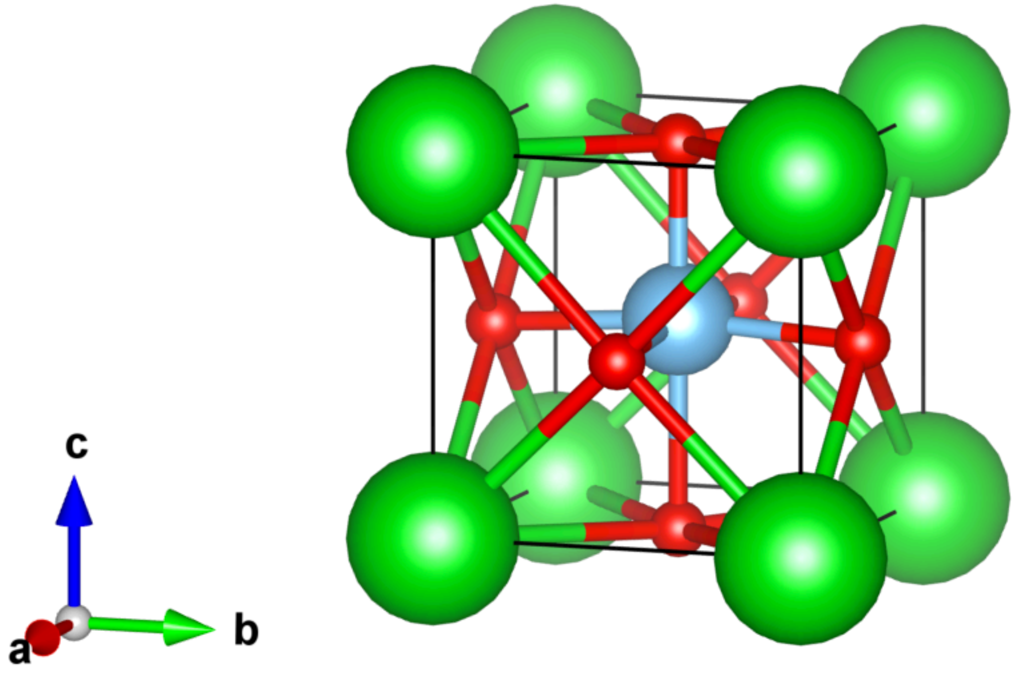


+ BerryPI

Work flow

w2web Create a new session "BaTiO3ncm" with the working directory "…/BaTiO3/BaTiO3ncm"

w2web Construct a structure file (we begin with the lowest-symmetry non-centrosymmetric structure fi



4-atoms (Ba, Ti & 2O) per unit cell
Body-centred tetragonal lattice (spacegroup #99 P4mm)
Cell size (Bohr): a = b = 7.547566; c = 7.626934
Angles: α = γ = β = 90°
Coordinates:
Ba (0 0 0)
Ti (1/2 1/2 0.51517436)
O (1/2 1/2 0.97356131)
O (1/2 0 0.48343742)
[You will see 1 additional equivalent atom after you save the structure O (0 1/2 0.48343742)]

Xcrysden Visualize the structure and compare to the one shown here

w2web Initialize SCF calculation (GGA-PBE) with a small (~3%) RMT reduction, RKMAX=7, and 230 k-points in the whole BZ.

Perform regular SCF calculation with default convergence parameters (~12 iterations). You are welcome to specify additional convergence criteria using –ec and –cc flags. It is the user's responsibility to check the convergence with respect to the quantity of interest (polarization in this case).

```
$   run_lapw
```

Run Berry phase calculation with a 6x6x6 k mesh. Note the "TOTAL POLARIZATION (C/m2)" along Z.

```
$   berrypi -k 6:6:6
```

A typical mistake is to take the calculated polarization as the spontaneous polarization. The spontaneous polarization is measured with respect to a reference structure, which is a centrosymmetric structure in this case.

w2web Create another session "BaTiO3cm" with the working directory ".../BaTiO3/BaTiO3cm"

Copy files to the new directory while you are still in ".../BaTiO3/BaTiO3ncm" folder

```
$   cp * ../BaTiO3cm
```

Change to the newly created folder

```
$   cd ../BaTiO3cm
```

Rename all BaTiO3ncm.* files to BaTiO3cm.* files

```
$   rename_files BaTiO3ncm BaTiO3cm
```

w2web Edit the "BaTiO3cm" structure file without changing the symmetry operations (do not change RMT radii)

Coordinates:

Ba	(0 0 0)
Ti	(1/2 1/2 1/2)
O1	(1/2 1/2 0)
O2	(1/2 0 1/2)
	(0 1/2 1/2)

Restore original k-mesh taking into account the symmetry with 230 k-points (shifted)

```
$   x kgen
```

Initialize the electron density for the new structure. (We do not want to fully initialize the calculation in order to preserve the former symmetry operations.)

```
$   x dstart
```

Perform standard SCF calculation

```
$   run_lapw
```

Run Berry phase calculation with a 6x6x6 k mesh. Note the "TOTAL POLARIZATION (C/m2)" along Z.

```
$   berrypi -k 6:6:6
```

Spontaneous polarization is obtained by taking a difference in polarization between distorted and the reference structures, i.e., P(BaTiO3ncm) and P(BaTiO3cm):

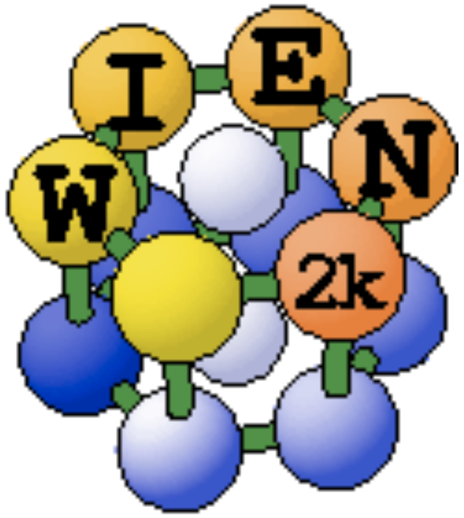
$$P_s = |P_{\text{non-centr.}} - P_{\text{centr.}}|$$

You should get Ps ~ 0.31 C/m^2. The experimental value is Ps = 0.26 C/m^2 for the tetragonal phase of $BaTiO_3$ at T ~ 180 K [H. H. Wieder, Phys. Rev. **99**, 1161 (1955)].

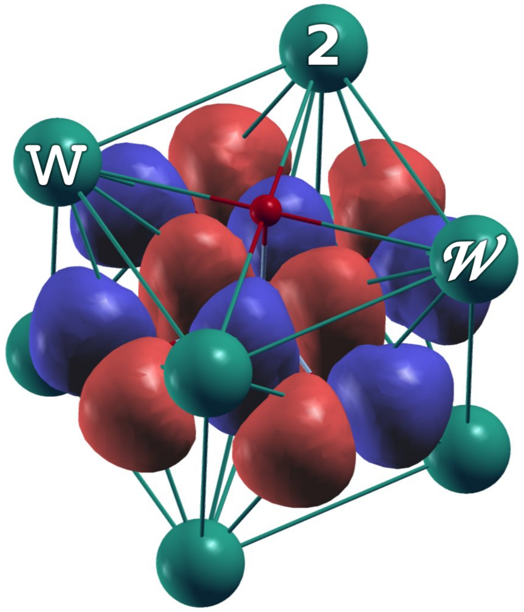
This tutorial was verified using

Wien2k 19.1
BerryPI (build Mar 08, 2019)
python 2.7.14
numpy 1.16.0
xcrysden 1.5.60

Topological properties (Chern number) of Weyl semimetal TaAs



+



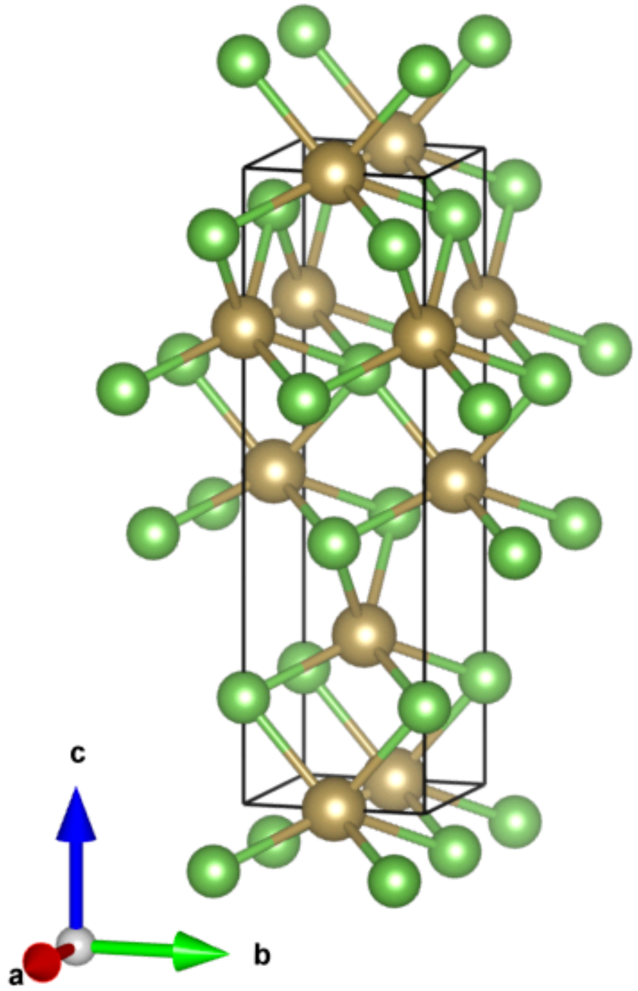
+ BerryPI

Work flow

w2web Construct a structure file

2-atoms (Ta & As) per unit cell
Body-centred tetragonal lattice (spacegroup #109 $I4_1md$)
Cell size (Bohr): a = b = 6.494611; c = 22.004349
Angles: $\alpha = \gamma = \beta = 90°$
Coordinates:
Ta (0 0 3/4)
As (0 0 0.1677)
[You will see 2 additional equivalent atoms after you save the structure
Ta (0 1/2 0) and As (0 1/2 0.4177)]

Initialize SCF calculation (GGA-PBE) with a small (~3%)
RMT reduction, RKMAX=7, and 300 k-points in the whole BZ.



Xcrysden Visualize the structure and compare to the one shown here

Perform regular SCF calculation with energy and change convergence of 0.0001 and 0.001, respectively (~9 iterations)

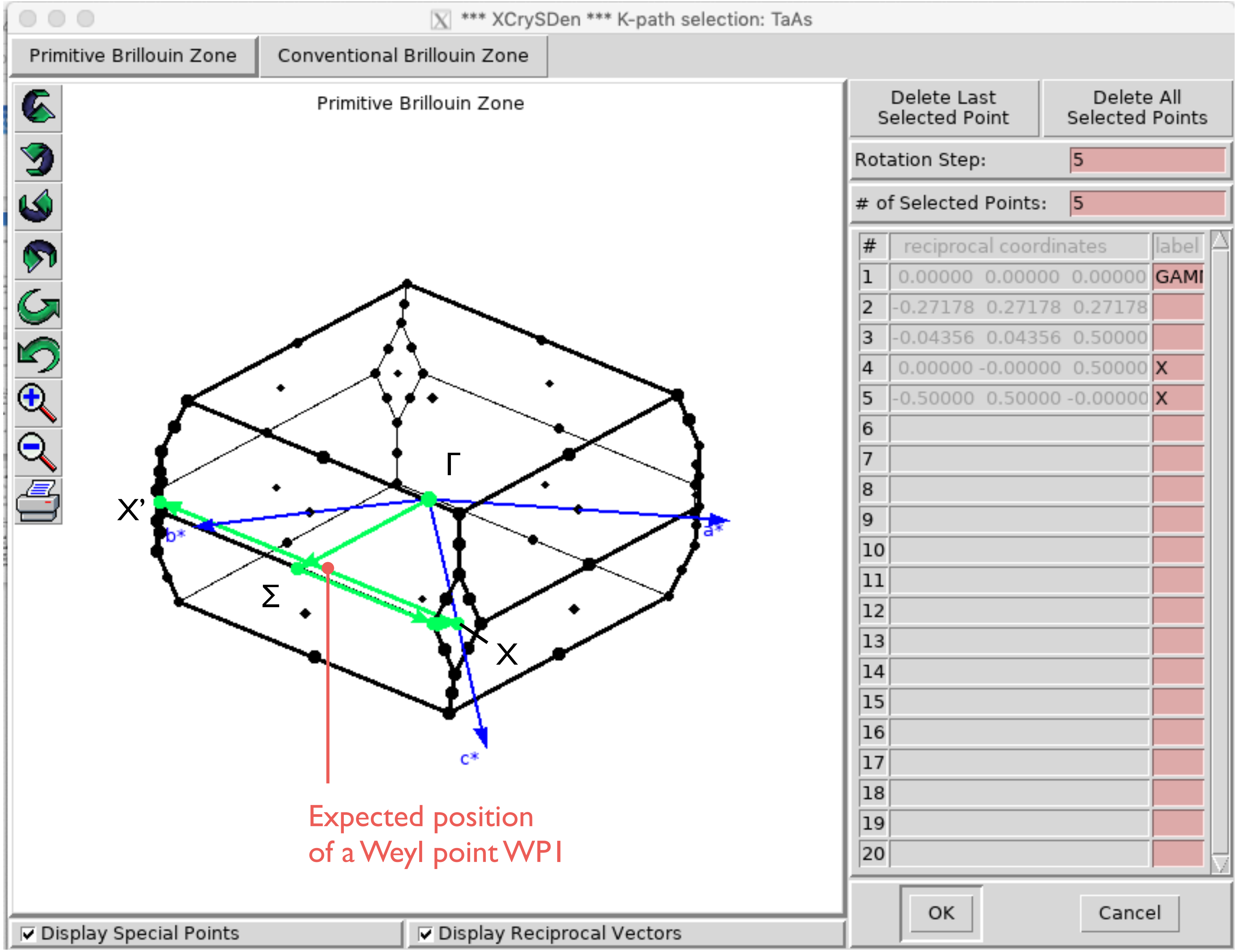
```
$    run_lapw -ec 0.0001 -cc 0.001
```

Save calculation, then initialize spin-orbit calculation (SOC) using all default parameters, and run SCF-SOC (~6 iterations)

```
$    save_lapw -d noSOC
$    init_so_lapw
$    run_lapw -ec 0.0001 -cc 0.001 -so
```

Check the band gap in case.scf file. It should be about 0.087 eV

Xcrysden File > Open WIEN2k > Select k-path for band structure plot that goes through points listed on the screenshot, ask for 600 points along the path and save the k-point list as "case.klist_band". It is always good to check the case.klist_band file and make sure there are no "******" entries resulted from an insufficient format.


*** XCrySDen *** K-path selection: TaAs
Primitive Brillouin Zone
Conventional Brillouin Zone
Primitive Brillouin Zone
Γ
X'
Σ
X
b*
a*
c*
Delete Last Selected Point
Delete All Selected Points
OK
Cancel
☑ Display Special Points
☑ Display Reciprocal Vectors

Rotation Step: 5

# of Selected Points: 5

#	reciprocal coordinates	label
1	0.00000 0.00000 0.00000	GAMI
2	-0.27178 0.27178 0.27178	
3	-0.04356 0.04356 0.50000	
4	0.00000 -0.00000 0.50000	X
5	-0.50000 0.50000 -0.00000	X
6		
7		
8		
9		
10		
11		
12		
13		
14		
15		
16		
17		
18		
19		
20		

Expected position of a Weyl point WP1

Recalculate eigenvalues on the k-path selected

```
$    x lapw1 -band
```

```
$    x lapwso
```

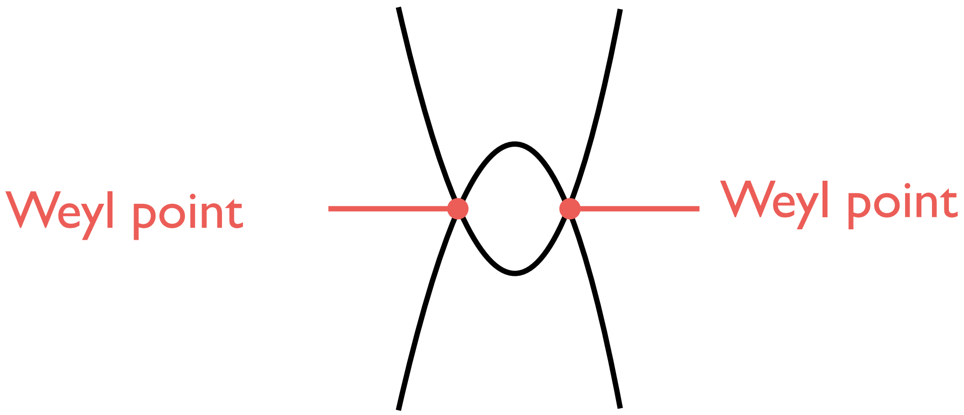
w2web Go to the “Bandstructure” menu and edit the case.insp file:
insert the Fermi energy of 0.801 Ry and set the energy range for plotting at [-1.0, 1.0] eV

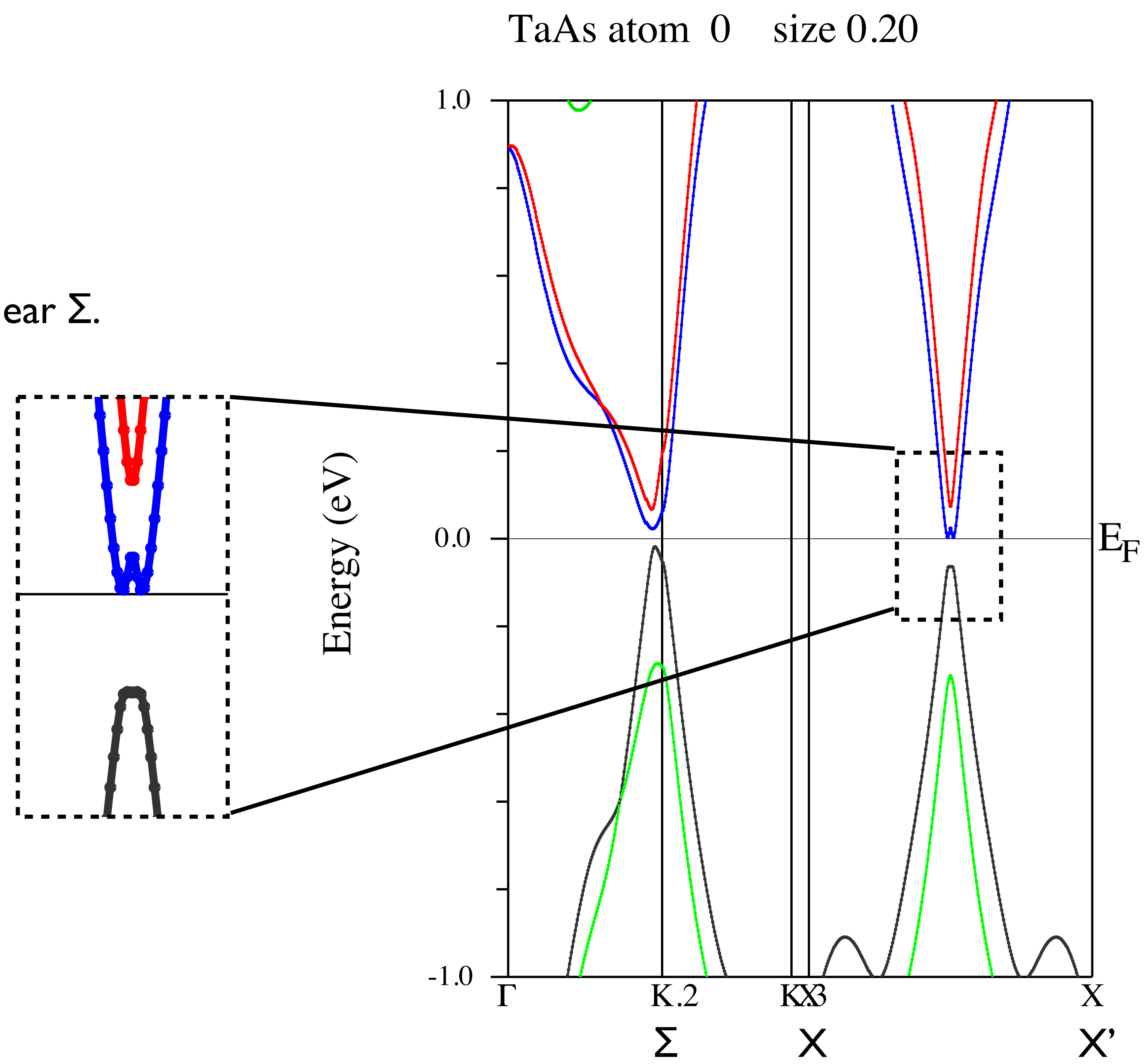
Run “spaghetti” with -so option

```
$    x spaghetti -so
```

w2web Plot the band structure. You will notice the band almost vanish as you approach Weyl points.

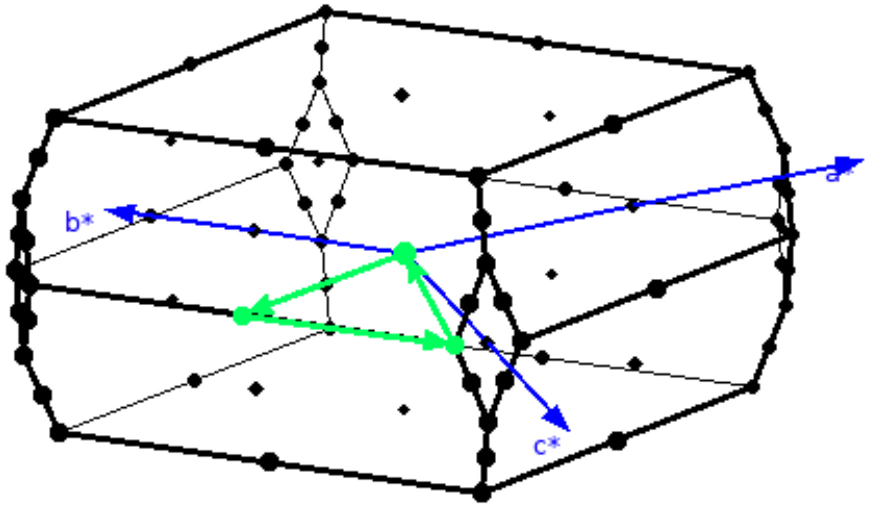
One set of Weyl points (nodes) is located near Σ. It is difficult to design the k-path such that it goes *exactly* through the Weyl point. It is because Weyl points usually do not coincide with high-symmetry points of BZ and do not lie on high-symmetry directions. (See figure at the end of this tutorial.)

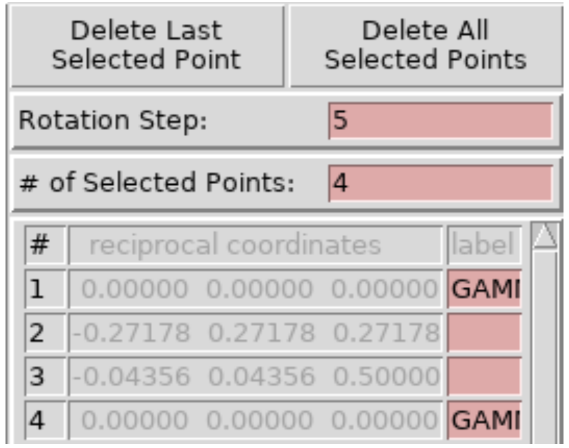




Next we calculate a Berry phase on a closed Wilson loop. It corresponds to the Berry flux through the loop. It is an analogy of the fundamental theorem of the curl, where the magnetic flux through a surface may also be defined as a line integral of the magnetic vector potential taken over the boundary of the surface.

Xcrysden Select k-path that encloses one (!) Weyl point as shown below with 40 points along the path and save as "TaAs.klist"

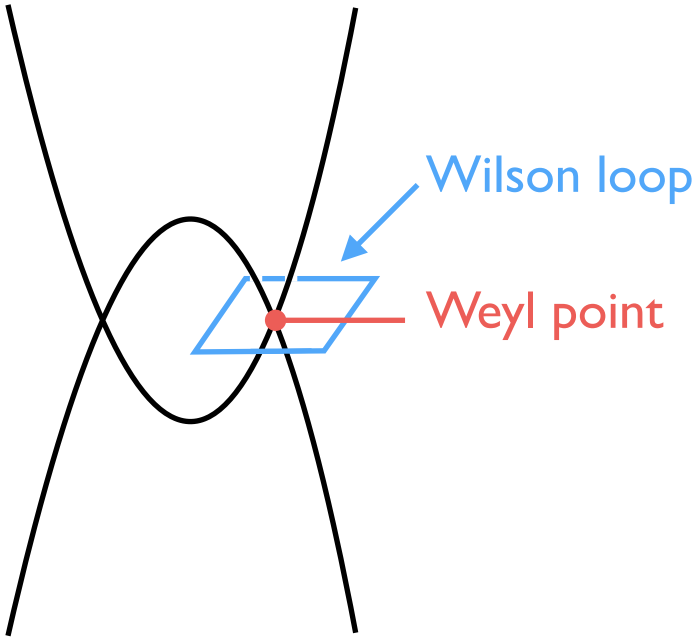




Rotation Step: 5

# of Selected Points: 4

#	reciprocal coordinates	label
1	0.00000 0.00000 0.00000	GAMI
2	-0.27178 0.27178 0.27178	
3	-0.04356 0.04356 0.50000	
4	0.00000 0.00000 0.00000	GAMI



Determine the number of occupied bands in TaAs.scf file (it should be 84)

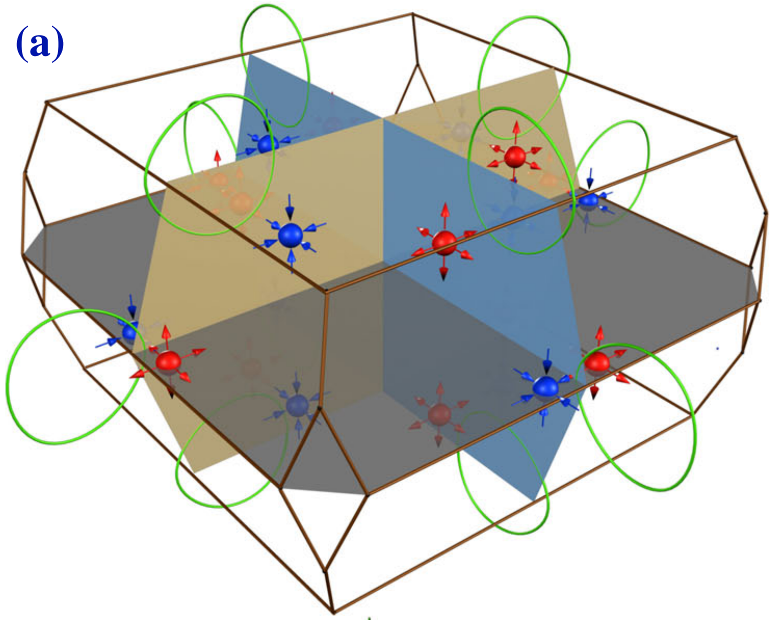
```
$   grep :BAN *scf
```

Run berry phase calculation on the Wilson loop for occupied bands only (-j is for SOC, -w is for Wilson loop, -b sets the range of bands)

```
$   berrypi -j -w -b 1:84
```

The results is: 'Berry phase sum (rad) =', -28.274333908589554 which is exactly -9π and is equivalent to |π| after subtracting an arbitrary number of 2π wrappings. The total Berry flux associated with the particular Weyl point is then |2π|. (The factor of 2 accounts for the fact that only half of the Berry flux goes through the loop.) It also corresponds to the Chern number of 1 (= |2π|/2π) as a topological characteristics. This gives us an indication of a monopole "charge of 1" associated with this Weyl point.

To expand the exercise, you are encouraged to construct a Wilson loop in the *same plane* as before, but not to include any Weyl points inside the loop (see the map of all Weil point in TaAs on the right). The corresponding Berry phase should be 0 (modulo of 2π).



Weng et al.,
Physical Review X 5, 011029 (2015)

This tutorial was verified using

Wien2k 19.1
BerryPI (build Mar 08, 2019)
python 2.7.14
numpy 1.16.0
xcrysden 1.5.60

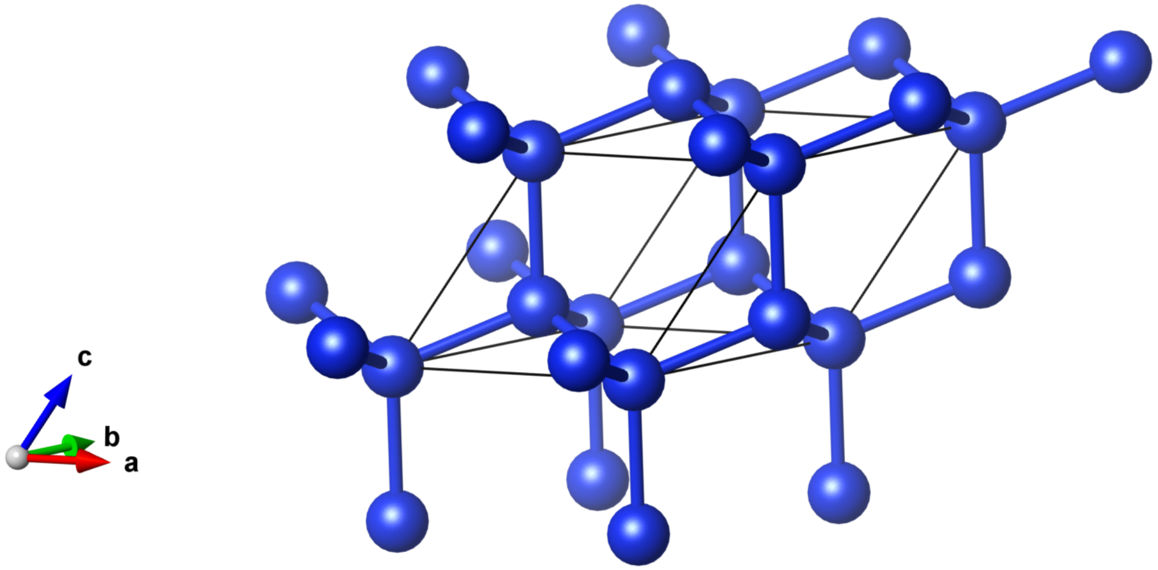
Effective band structure of $Si_{1-x}Ge_x$ alloy

WIEN2k + fold2Bloch

YouTube video:

Instructions

w2web Construct a primitive lattice of Si



2-atoms per unit cell
Primitive lattice “P”, α = β = γ = 60°
Cell size (Ang.): a = b = c = 3.870393 Å
Coordinates:
Si (0 0 0)
Si (1/4 1/4 1/4)

RMTs = 2 Bohrs

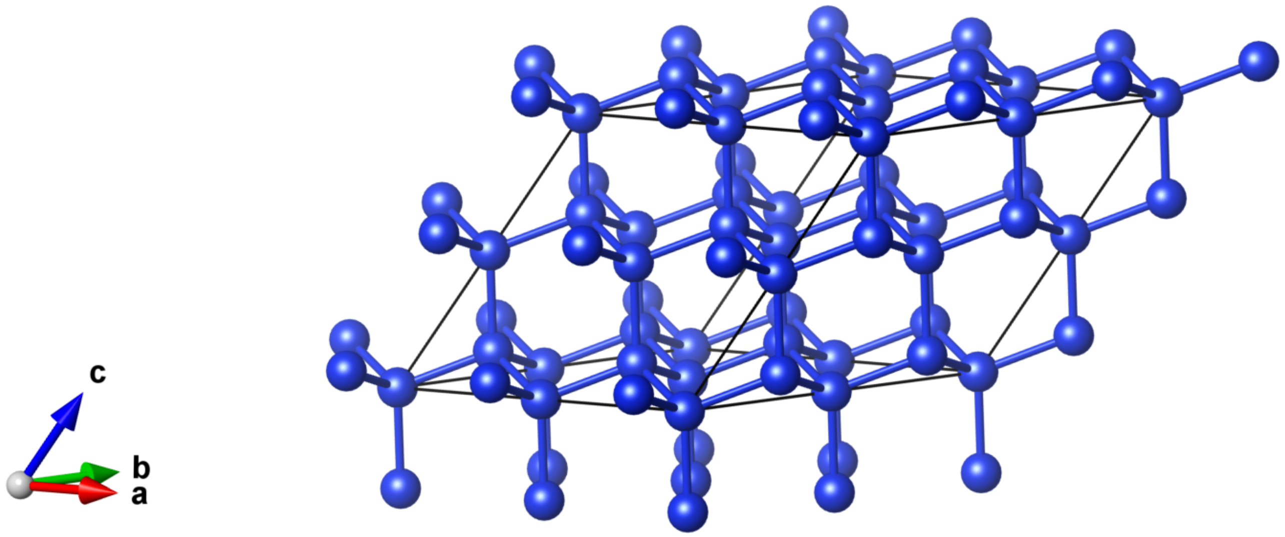
Do **not** initialize calculation.

Create a 2x2x2 supercell: no shifts, P-type lattice, no additional vacuum

```
$  x supercell
```

Rename “[case]_super.struct”
to “[case].struct”

The new supercell should have
16 Si-atoms (2 atoms x 2 x 2 x 2)



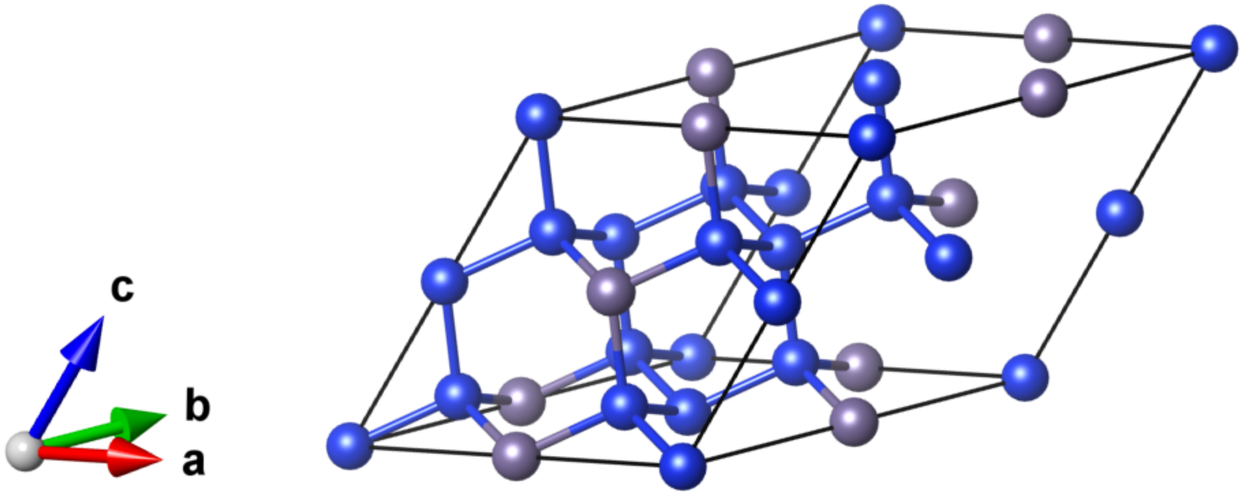
w2web Introduce Ge into the structure by switching 3 Si → 3 Ge atoms
The Ge composition will be 3/16 = 0.1875 (*ca.* 19%)

One possibility is to pick any 3 Si-atoms of your choice. But this choice will be "biased" and does not qualify as a random alloy.

Alternatively, we can generate a randomized sequence of integers between 1 and 16 (total number of Si-atoms) using https://www.random.org/sequences
"**3** **2** **6** 9 13 4 7 16 12 1 5 10 15 14 11 8"
The first 3 numbers are labels of atoms to be switched. Of course every time you will get a different sequence. Thus the arrangement of atoms is not unique.

More elegant solution involves the use of "Special Quasirandom Structures", but it goes beyond this tutorial.

When substituting Si → Ge using StructGen^{TM}, make sure to clear up the Z value and label atoms as Ge 1, Ge 2, Ge 3, Si 1, Si 2, …



Initialize SCF calculation: LDA, RKmax=6, 3x3x3 k-mesh (shifted)
(note: crude parameters are used for tutorial purposes)

```
$   init_lapw -b … or web interface
```

Run SCF calculation (~10 iterations): iterative diagonalization, energy convergence 0.0001, charge convergence 0.001. It can be faster if you parallelize over k-points and/or take advantage of OMP-parallelization (~10 mins on 4 cores)

```
$   export OMP_NUM_THREADS=4

$   run_lapw -it -ec …  (without -p)
```

or

Prepare `.machines` file that contains these two lines (replace XX with your specific host number)

```
1:psiXX
1:psiXX
```

```
$   run_lapw -it -p -ec …
```

Save the calculation in "SCF" directory

```
$   save_lapw … or web interface
```

Copy L-Γ-X k-path template (`LGX.klist_band` not a part of the standard Wien2k distribution). Look inside the file. It is not a standard k path used for band structure plots. You will see the path going from L to Gamma to -L and then X to -X. Usually we would consider this path as redundant for a regular band structure plot. But for unfolding to work, we need the path so span the entire width of the Brillouin zone.

```
$   cp $WIENROOT/LGX.klist_band [case].klist_band
```

Recalculate eigenvalues and wave functions for k-points on the path selected. Do not use k-point parallelization here as it will change the workflow of fold2Bloch analysis because multiple [case].vector files will be generated. (Please talk to an instructor if you would like to explore [-p] this option.)

```
$   x lapw1 -band
```

Make sure the vector files are located in the same directory (sensitive to settings of the $SCRATCH variable)

```
$   ls -l [case].vector
```

Perform unfolding of the vector file using 2:2:2 as a setting for the size of the supercell

```
$   fold2Bloch [case].vector 2:2:2
```

The newly created file [case].f2b contains 5 columns:
Unfolded KX, KY, KZ, Eigenvalue (Ry), Bloch spectral weight

The Fermi energy can typically be found in [case].scf file (:FER label). However, it can be inaccurate, if the k-point mesh for SCF calculation did not include special points (Γ point in this case). The workaround is to find the Fermi energy (typically 0.38-0.39 Ry) by examining sorted eigenvalues (4th column).

```
$   sort -n -k4,4 [case].f2b | less
```

Note values of the spectral weight (5th column) for the valence and conduction band edges. Which of them is more perturbed by the alloy disorder, i.e., preserved less Bloch character?

Copy plotting script to the Wien2k work directory
(`ubs_dots_w2k_octave.m` is not a part of the standard Wien2k distribution)

```
$   cp $WIENROOT/ubs_dots_w2k_octave.m .
```

Edit the plotting script:

KPATH = [1/2 0 0; 0 0 0; 1/2 1/2 0] corresponds to L-Γ-X;

FOLDS = [2 2 2] is our size of the supercell;

KLABEL = {'L'; 'G'; ‘X'} are labels for k-points;

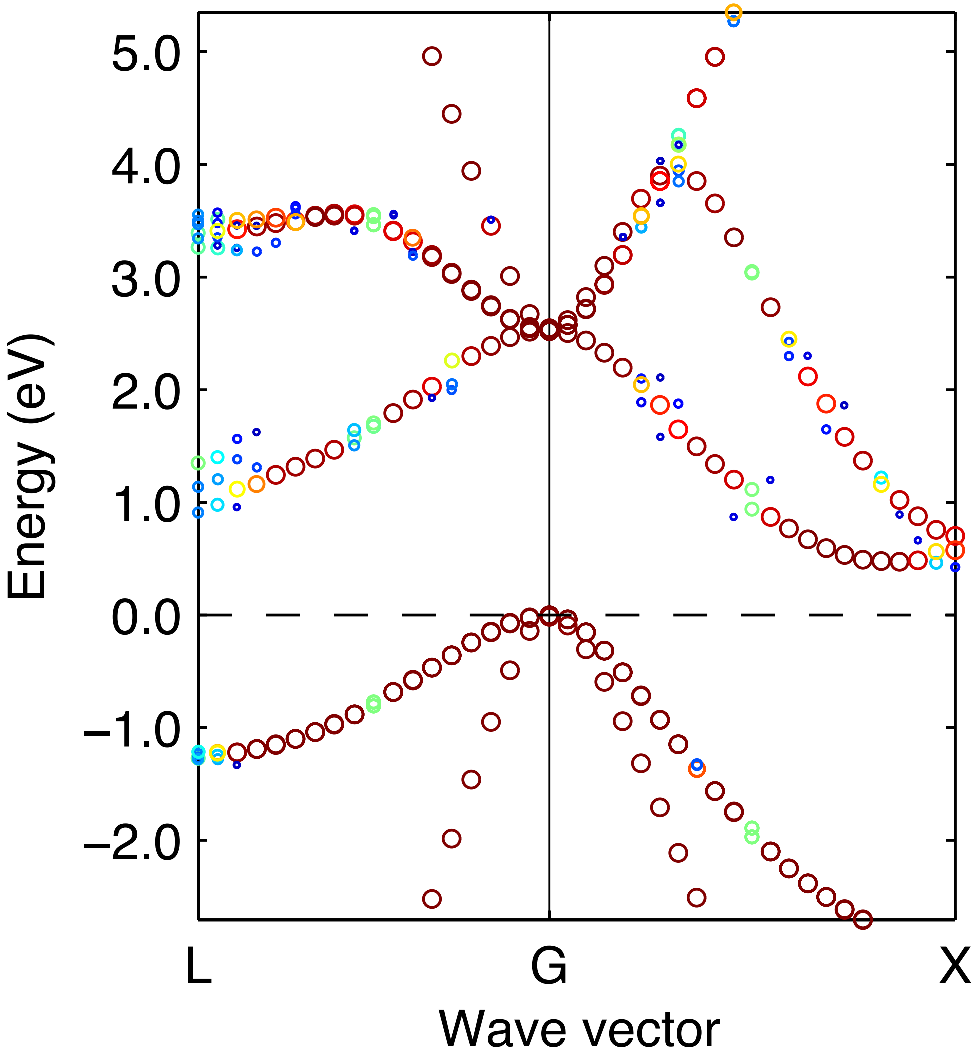
finpt = ‘[case].f2b’ is the input file name;

Ef = 0.385799 is the Fermi energy (Ry);

G = [0.083726 -0.027909 -0.027909;
0.000000 0.078938 -0.039469;
0.000000 0.000000 0.068362]
reciprocal lattice vectors from [case].outputkgen;

Lunch Octave in GUI mode

```
$   octave
octave >> ubs_dots_w2k_octave
```

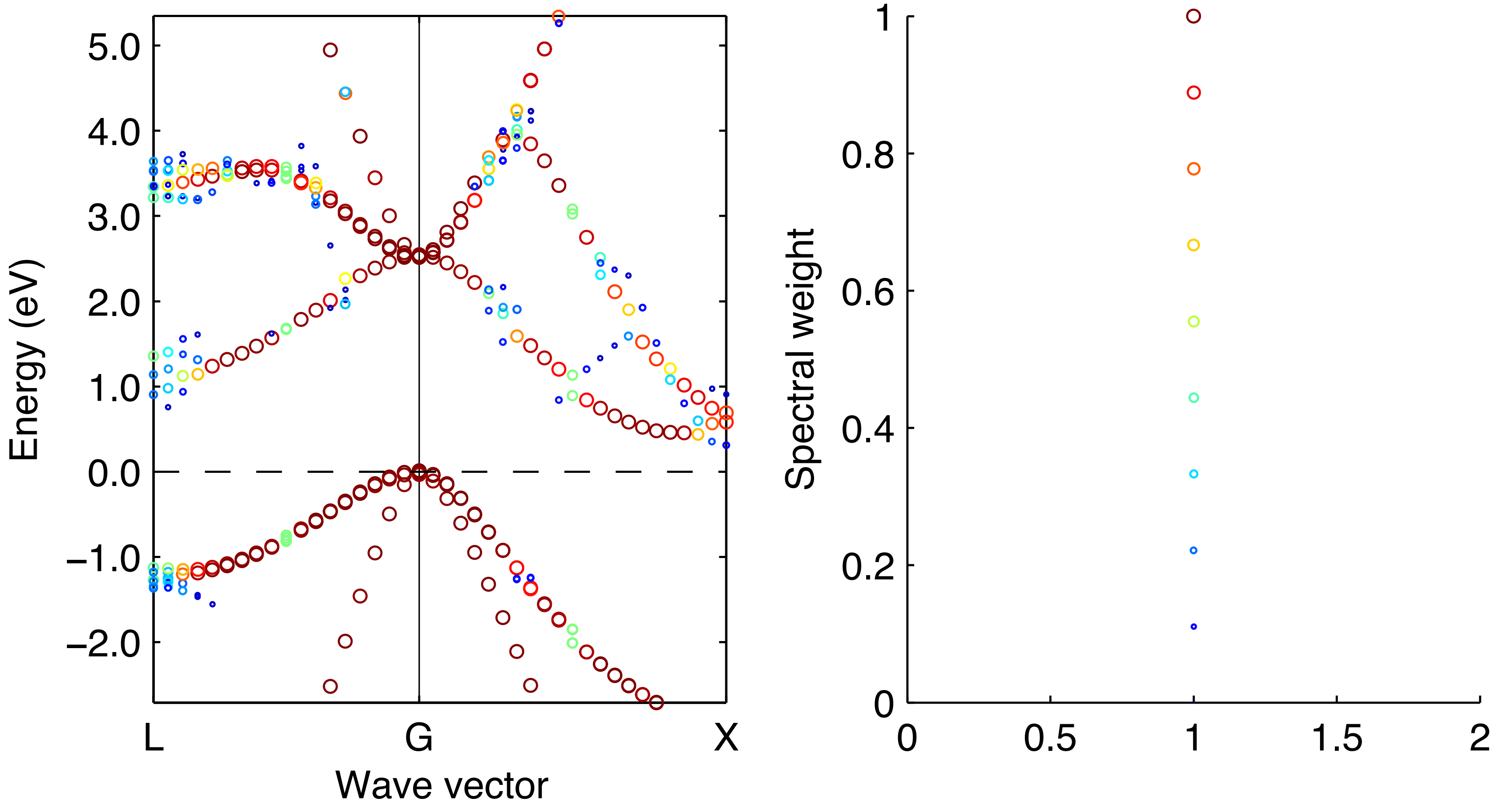


Run plotting script in Octave and view the effective band structure of the alloy.
Note a well-preserved Bloch character at the top of the valence band and bottom of the conduction band in spite of the fact that the structure does not have a zinc-blende symmetry any longer.

When you have time (requires1-2 hrs depending on parallelization and load of machines)…

It should be noted that so far we ignored static atomic displacements that are present in alloys due to the size mismatch between the additive element (Ge) and the host (Si). Forces in our structure are of the order 15 mRy/Bohr that requires optimization of atomic positions.

The suggested procedure would be to build supercell, place alloying element, and perform optimization of atomic positions before undertaking infolding. The effective band structure with static atomic displacements included in calculation is shown below. Can you spot any changes at the band edges?



This tutorial was verified using

Wien2k 19.1
fold2Bloch (build May 29, 2014)
octave 4.2.1