

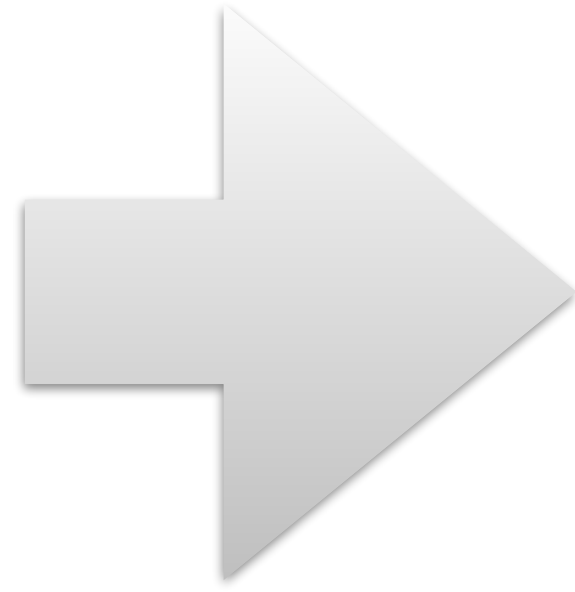
# Introduction to WIEN2k

Weine Olovsson

National Supercomputer Centre (NSC), Linköping University

SNIC training, online mini workshop @NSC 18<sup>th</sup> Nov 2021

# Schedule



[https://www.nsc.liu.se/support/Events/WIEN2k\\_intro\\_Nov2021/](https://www.nsc.liu.se/support/Events/WIEN2k_intro_Nov2021/)

## Tentative Schedule

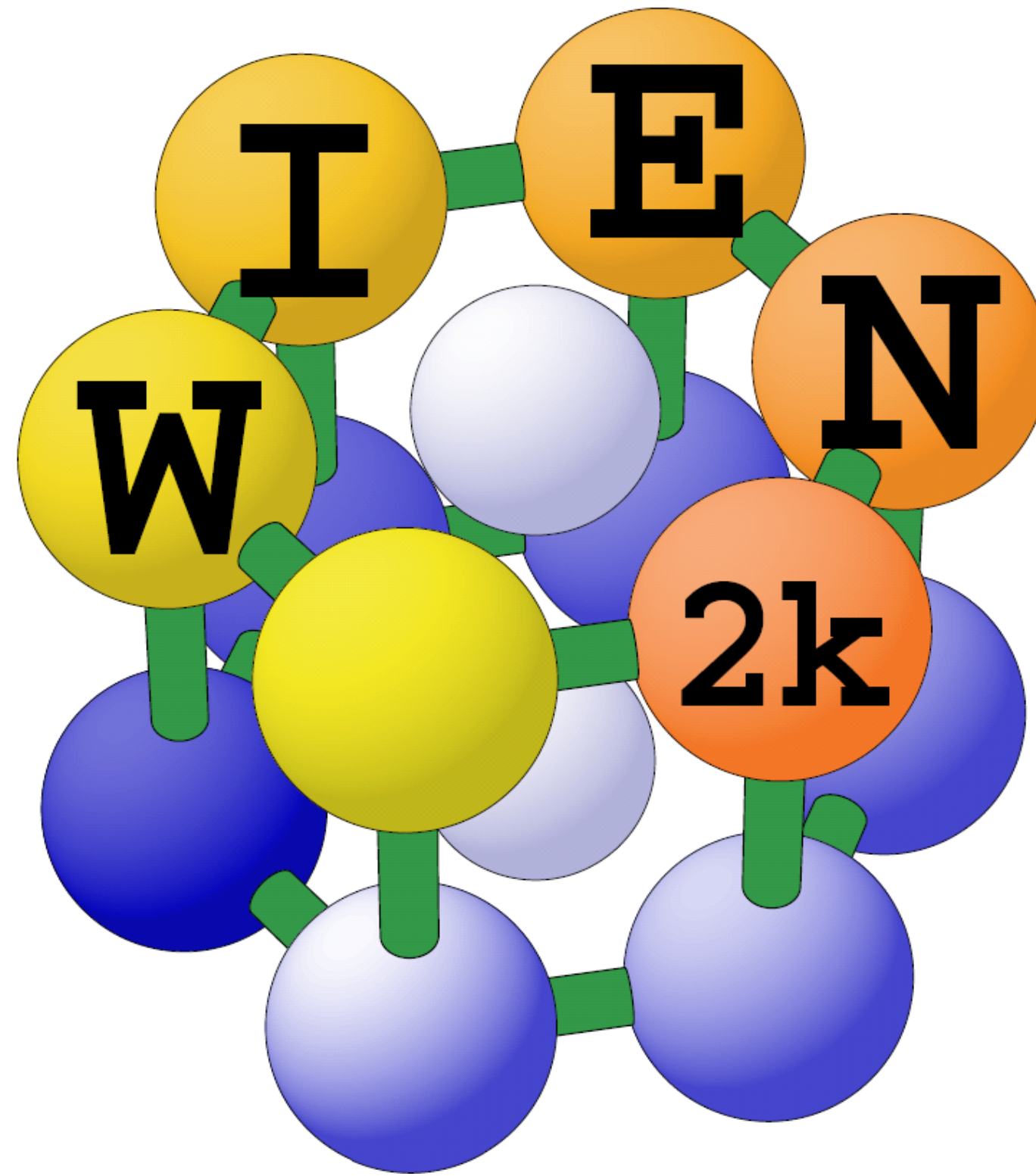
Tentative schedule, times might differ. It will end at latest 17:00, probably earlier. There will also be some breaks.

**13:00 -14:30** Theory and basics

**14:30 -15:30** Practical examples

**15:30 -17:00** Hands-on and questions

Let's take breaks at times!



# WIEN2k

An Augmented Plane Wave Plus Local Orbitals Program  
for Calculating Crystal Properties

<http://wien2k.at/>

# Introduction to WIEN2k

- Basic theory (very brief) - *see resources for refs.*
- Focus on **practical aspects** of running WIEN2k at NSC
- Setting up a job: [init\\_lapw](#) (script) & [w2web](#) (GUI)
- Running efficiently, benchmarks
- Examples
- Check out resources! ... clickable links are [underlined](#)  
- *examples, hints, etc.*



# 1. Theory & Resources

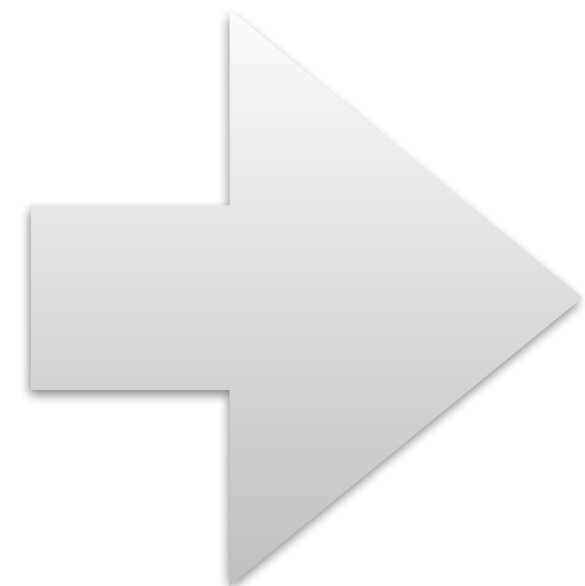
Weine Olovsson

National Supercomputer Centre (NSC), Linköping University

SNIC training, online mini workshop @NSC 18<sup>th</sup> Nov 2021

# Short background

- Software license (group)
- DFT, FP-LAPW-method, (L)APW+lo
- Beyond DFT (GW, BSE, DMFT, ...)
- **widely used** for high accuracy
- Specialized programs (NMR, TELNES, ...)



<http://wien2k.at/>

# APW methods

- Augmented Plane Waves (**APW**), *J.C. Slater*  
[Phys. Rev. 51, 846 \(1937\)](#)
- Linear APW (**LAPW**), *O.K. Andersen*  
[Phys. Rev. B 12, 3060 \(1975\)](#)
- **Local orbitals**, *D.J. Singh*  
[Phys. Rev. B 43, 6388 \(1991\)](#)
- **APW+lo**, *E. Sjöstedt, L. Nordström, D.J. Singh*  
[Sol. Stat. Com. 114, 15 \(2000\)](#)

# FPLAPW & WIEN2k

- All electron method
- Full-potential (no shape approx.)
- Core, semi-core & valence states
- WIEN2k, mixed basis set: APW+lo & LAPW+LO
- P. Blaha *et al.* J. Chem. Phys. **152**, 074101 (2020)

<https://doi.org/10.1063/1.5143061>



# Accuracy: WIEN2k as reference

PAW (VASP) compared with FPLAPW (WIEN2k):

$$\Delta(\text{PAW})_{(\text{VASP})} = 1.9 \text{ meV/atom}$$

H																		He
0.1																		0.0
Li	Be											B	C	N	O	F		Ne
0.2	0.1											0.3	0.3	10.6	8.3	1.5		0.1
Na	Mg											Al	Si	P	S	Cl		Ar
0.0	0.7											0.3	2.0	3.8	3.3	4.0		0.1
K	Ca	Sc	Ti	V	Cr	Mn	Fe	Co	Ni	Cu	Zn	Ga	Ge	As	Se	Br		Kr
0.1	0.2	0.4	0.9	1.3	3.1	1.4	3.4	3.4	2.0	0.4	0.3	0.2	2.4	1.7	1.5	1.5		0.1
Rb	Sr	Y	Zr	Nb	Mo	Tc	Ru	Rh	Pd	Ag	Cd	In	Sn	Sb	Te	I		Xe
0.1	0.1	0.5	2.7	7.3	5.5	8.3	2.3	5.4	4.4	4.1	1.4	0.4	0.2	0.1	0.5	0.9		0.1
Cs	Ba	Lu	Hf	Ta	W	Re	Os	Ir	Pt	Au	Hg	Tl	Pb	Bi	Po	At		Rn
0.3	0.7	4.3	1.2	1.0	3.5	4.3	3.8	1.9	2.5	5.9	0.5	0.4	0.6	0.4	0.4			0.0



Lejaeghere et al. Science 351, 1415 (2016)

	AE													PAW								USPP						NCPP													
	Elk	exciting	FHI-aims/tight	FHI-aims/really_tight	FHI-aims/tier2	FLEUR	FPLO/default	FPLO/T+F	FPLO/T+F+s	RSpt	WIEN2k/default	WIEN2k/enhanced	WIEN2k/acc	GBRV12/ABINIT	GPAW06/GPAW	GPAW09/ABINIT	GPAW09/GPAW	JTH01/ABINIT	JTH02/ABINIT	PSlib031/QE	PSlib100/QE	VASP2007/VASP	VASP2012/VASP	VASPGW2015/VASP	GBRV12/QE	GBRV14/CASTEP	GBRV14/QE	OTFG7/CASTEP	OTFG9/CASTEP	SSSP/QE	Vdb/CASTEP	Vdb2/DACAP0	FHI98pp/ABINIT	HGH/ABINIT	HGH-NLCC/BigDFT	MBK2013/OpenMX	ONCVSP (PD0.1)/ABINIT	ONCVSP (SG15)1/CASTEP	ONCVSP (SG15)1/QE	ONCVSP (SG15)2/CASTEP	
AE	Elk	0.3	0.6	0.6	0.3	0.6	3.9	1.0	1.0	0.9	1.7	1.8	0.3	0.9	3.8	1.3	1.5	1.2	0.6	1.6	0.9	2.1	0.7	0.4	1.1	1.1	1.0	2.5	0.4	0.4	6.4	6.3	13.5	2.2	1.1	2.1	0.7	1.5	1.4	1.4	
	exciting	0.3		0.5	0.5	0.1	0.5	3.9	1.0	0.9	0.8	1.7	1.8	0.2	0.8	3.8	1.3	1.5	1.2	0.6	1.6	0.8	2.1	0.6	0.4	1.0	1.1	1.0	2.5	0.5	0.3	6.4	6.3	13.4	2.2	1.1	2.1	0.7	1.4	1.3	1.4
	FHI-aims/tight	0.6	0.5		0.0	0.5	0.7	3.8	0.9	1.1	0.7	1.8	1.8	0.5	1.0	3.8	1.3	1.6	1.3	0.7	1.7	1.0	2.2	0.8	0.6	1.1	1.2	1.1	2.6	0.7	0.6	6.4	6.3	13.6	2.2	1.2	2.0	0.8	1.5	1.4	1.5
	FHI-aims/really_tight	0.6	0.5	0.0		0.5	0.7	3.8	0.9	1.1	0.8	1.8	1.8	0.5	1.0	3.8	1.3	1.6	1.3	0.7	1.7	1.0	2.2	0.8	0.6	1.1	1.2	1.1	2.6	0.7	0.6	6.5	6.3	13.6	2.2	1.2	2.0	0.8	1.5	1.4	1.5
	FHI-aims/tier2	0.3	0.1	0.5	0.5		0.5	3.9	0.9	0.9	0.8	1.7	1.8	0.2	0.8	3.8	1.3	1.5	1.2	0.6	1.6	0.8	2.0	0.6	0.4	0.9	1.0	0.9	2.5	0.5	0.3	6.4	6.3	13.4	2.2	1.1	2.1	0.7	1.4	1.3	1.4
	FLEUR	0.6	0.5	0.7	0.7	0.5		3.6	0.8	0.8	0.6	1.4	1.5	0.4	0.9	3.5	1.3	1.5	1.0	0.6	1.5	0.8	1.9	0.7	0.6	1.0	1.0	1.0	2.6	0.7	0.5	6.5	6.3	13.2	2.0	1.0	1.9	0.6	1.3	1.3	1.3
	FPLO/default	3.9	3.9	3.8	3.8	3.9	3.6		3.1	3.6	3.3	2.9	2.5	3.9	4.0	3.1	4.1	4.1	3.4	3.6	3.3	3.9	2.8	3.9	4.0	4.0	4.0	4.1	5.8	4.1	3.9	7.9	7.2	13.0	4.9	3.6	3.2	3.7	4.1	4.1	4.1
	FPLO/T+F	1.0	1.0	0.9	0.9	0.9	0.8	3.1		0.8	0.7	1.4	1.4	0.9	1.3	3.4	1.7	1.9	1.0	0.9	1.5	1.3	1.9	1.2	1.0	1.3	1.3	1.3	3.1	1.1	1.0	6.6	6.4	13.7	2.4	1.2	1.8	1.0	1.6	1.6	1.6
	FPLO/T+F+s	1.0	0.9	1.1	1.1	0.9	0.8	3.6	0.8		0.9	1.5	1.5	0.9	1.3	3.5	1.7	1.8	1.2	0.9	1.4	1.3	1.9	1.2	1.0	1.4	1.4	1.4	2.9	1.0	0.9	6.4	6.4	13.0	2.3	1.2	1.8	1.0	1.6	1.6	1.6
	RSpt	0.9	0.8	0.7	0.8	0.8	0.6	3.3	0.7	0.9		1.3	1.3	0.8	1.1	3.4	1.5	1.7	0.9	0.7	1.6	1.1	1.9	1.0	0.8	1.2	1.3	1.3	3.0	1.0	0.8	6.7	6.5	13.2	2.2	1.1	1.8	0.8	1.5	1.5	1.5
	WIEN2k/default	1.7	1.7	1.8	1.8	1.7	1.4	2.9	1.4	1.5	1.3		0.9	1.7	1.9	3.2	2.2	2.3	1.3	1.5	1.8	1.8	1.7	1.8	1.8	1.9	1.9	1.9	3.8	1.8	1.6	7.1	7.0	13.0	2.8	1.7	1.9	1.6	2.1	2.1	2.1
	WIEN2k/enhanced	1.8	1.8	1.8	1.8	1.8	1.5	2.5	1.4	1.5	1.3	0.9		1.8	2.0	2.6	2.1	2.2	1.1	1.5	1.6	1.8	1.4	1.9	2.0	2.0	2.0	2.0	3.8	2.0	1.7	6.9	6.9	12.3	2.8	1.6	1.5	1.7	1.9	1.9	1.9
WIEN2k/acc	0.3	0.2	0.5	0.5	0.2	0.4	3.9	0.9	0.9	0.8	1.7	1.8		0.8	3.8	1.3	1.5	1.2	0.5	1.6	0.8	2.0	0.7	0.3	0.9	1.0	1.0	2.5	0.5	0.3	6.4	6.2	13.4	2.1	1.0	2.0	0.6	1.4	1.3	1.4	
PAW	GBRV12/ABINIT	0.9	0.8	1.0	1.0	0.8	0.9	4.0	1.3	1.3	1.1	1.9	2.0	0.8		4.1	1.5	1.6	1.5	1.1	2.0	1.1	2.3	1.0	0.9	0.7	0.8	0.7	2.8	1.0	0.7	6.4	6.3	15.1	2.5	1.5	2.4	1.1	1.8	1.7	1.8
	GPAW06/GPAW	3.8	3.8	3.8	3.8	3.8	3.5	3.1	3.4	3.5	3.4	3.2	2.6	3.8	4.1		3.6	3.5	3.2	3.5	3.0	3.8	2.8	3.7	3.8	4.0	3.8	4.0	5.6	3.9	3.6	7.4	7.6	12.3	4.5	3.0	3.0	3.6	3.7	3.8	3.7
	GPAW09/ABINIT	1.3	1.3	1.3	1.3	1.3	1.3	4.1	1.7	1.7	1.5	2.2	2.1	1.3	1.5	3.6		0.6	1.5	1.4	2.0	1.5	2.4	1.4	1.3	1.6	1.6	1.6	2.5	1.4	1.3	6.5	6.1	13.6	2.3	1.7	2.3	1.2	1.7	1.7	1.7
	GPAW09/GPAW	1.5	1.5	1.6	1.6	1.5	1.5	4.1	1.9	1.8	1.7	2.3	2.2	1.5	1.6	3.5	0.6		1.6	1.5	2.1	1.6	2.5	1.6	1.5	1.7	1.7	1.7	2.7	1.5	1.4	6.5	6.1	13.6	2.5	1.8	2.3	1.5	1.8	1.8	1.8
	JTH01/ABINIT	1.2	1.2	1.3	1.3	1.2	1.0	3.4	1.0	1.2	0.9	1.3	1.1	1.2	1.5	3.2	1.5	1.6		0.9	1.5	1.4	1.9	1.4	1.3	1.5	1.5	1.5	3.0	1.4	1.1	6.5	6.5	13.0	2.2	1.3	1.5	1.2	1.4	1.4	1.4
	JTH02/ABINIT	0.6	0.6	0.7	0.7	0.6	0.6	3.6	0.9	0.9	0.7	1.5	1.5	0.5	1.1	3.5	1.4	1.5	0.9		1.4	0.9	1.9	0.7	0.7	1.2	1.2	1.2	2.6	0.7	0.6	6.3	6.2	13.4	2.2	1.2	1.9	0.7	1.4	1.4	1.4
	PSlib031/QE	1.6	1.6	1.7	1.7	1.6	1.5	3.3	1.5	1.4	1.6	1.8	1.6	1.6	2.0	3.0	2.0	2.1	1.5	1.4		1.6	1.5	1.6	1.6	2.0	1.9	2.0	3.1	1.6	1.5	6.1	5.8	12.8	2.4	1.6	1.7	1.5	2.1	2.2	2.1
	PSlib100/QE	0.9	0.8	1.0	1.0	0.8	0.8	3.9	1.3	1.3	1.1	1.8	1.8	0.8	1.1	3.8	1.5	1.6	1.4	0.9	1.6		1.7	1.0	0.8	1.1	1.2	1.2	2.2	0.9	0.7	6.1	5.9	13.5	2.1	1.4	1.9	0.9	1.6	1.6	1.6
	VASP2007/VASP	2.1	2.1	2.2	2.2	2.0	1.9	2.8	1.9	1.9	1.9	1.7	1.4	2.0	2.3	2.8	2.4	2.5	1.9	1.9	1.5	1.7		1.8	2.1	2.1	2.2	2.1	3.5	2.1	1.9	6.5	6.1	12.4	3.0	2.2	1.7	1.9	2.5	2.4	2.5
	VASP2012/VASP	0.7	0.6	0.8	0.8	0.6	0.7	3.9	1.2	1.2	1.0	1.8	1.9	0.7	1.0	3.7	1.4	1.6	1.4	0.7	1.6	1.0	1.8		0.7	1.1	1.2	1.1	2.5	0.8	0.6	6.5	6.3	13.4	2.2	1.2	2.1	0.9	1.6	1.5	1.6
VASPGW2015/VASP	0.4	0.4	0.6	0.6	0.4	0.6	4.0	1.0	1.0	0.8	1.8	2.0	0.3	0.9	3.8	1.3	1.5	1.3	0.7	1.6	0.8	2.1	0.7		1.1	1.1	1.1	2.6	0.5	0.4	6.6	6.2	13.7	2.2	1.1	2.2	0.7	1.5	1.4	1.5	
USPP	GBRV12/QE	1.1	1.0	1.1	1.1	0.9	1.0	4.0	1.3	1.4	1.2	1.9	2.0	0.9	0.7	4.0	1.6	1.7	1.5	1.2	2.0	1.1	2.1	1.1	1.1		0.4	0.1	2.6	1.0	0.8	6.3	6.4	15.3	2.3	1.4	2.1	1.2	1.6	1.5	1.6
	GBRV14/CASTEP	1.1	1.1	1.2	1.2	1.0	1.0	4.0	1.3	1.4	1.3	1.9	2.0	1.0	0.8	3.8	1.6	1.7	1.5	1.2	1.9	1.2	2.2	1.2	1.1	0.4		0.3	2.6	0.9	0.9	6.2	6.3	15.0	2.4	1.6	2.1	1.1	1.5	1.5	1.5
	GBRV14/QE	1.0	1.0	1.1	1.1	0.9	1.0	4.1	1.3	1.4	1.3	1.9	2.0	1.0	0.7	4.0	1.6	1.7	1.5	1.2	2.0	1.2	2.1	1.1	1.1	0.1	0.3		2.6	1.0	0.8	6.3	6.3	15.2	2.3	1.4	2.1	1.2	1.6	1.5	1.5
	OTFG7/CASTEP	2.5	2.5	2.6	2.6	2.5	2.6	5.8	3.1	2.9	3.0	3.8	3.8	2.5	2.8	5.6	2.5	2.7	3.0	2.6	3.1	2.2	3.5	2.5	2.6	2.6	2.6	2.6		2.2	2.4	4.8	5.7	14.5	2.7	2.9	3.4	2.4	2.6	2.6	2.6
	OTFG9/CASTEP	0.4	0.5	0.7	0.7	0.5	0.7	4.1	1.1	1.0	1.0	1.8	2.0	0.5	1.0	3.9	1.4	1.5	1.4	0.7	1.6	0.9	2.1	0.8	0.5	1.0	0.9	1.0	2.2		0.6	6.3	6.2	13.6	2.2	1.1	2.1	0.8	1.5	1.4	1.5
	SSSP/QE	0.4	0.3	0.6	0.6	0.3	0.5	3.9	1.0	0.9	0.8	1.6	1.7	0.3	0.7	3.6	1.3	1.4	1.1	0.6	1.5	0.7	1.9	0.6	0.4	0.8	0.9	0.8	2.4	0.6		6.4	6.2	13.6	2.1	1.0	2.0	0.7	1.4	1.2	1.3
	Vdb/CASTEP	6.4	6.4	6.4	6.5	6.4	6.5	7.9	6.6	6.4	6.7	7.1	6.9	6.4	6.4	7.4	6.5	6.5	6.5	6.3	6.1	6.1	6.5	6.5	6.6	6.3	6.2	6.3	4.8	6.3	6.4		9.6	16.3	6.6	6.1	6.6	6.4	5.7	5.8	5.7
	Vdb2/DACAP0	6.3	6.3	6.3	6.3	6.3	6.3	7.2	6.4	6.4	6.5	7.0	6.9	6.2	6.3	7.6	6.1	6.1	6.5	6.2	5.8	5.9	6.1	6.3	6.2	6.4	6.3	6.3	5.7	6.2	6.2	9.6		17.9	6.2	5.9	6.4	6.1	6.5	6.5	6.5
NCPP	FHI98pp/ABINIT	13.5	13.4	13.6	13.6	13.4	13.2	13.0	13.7	13.0	13.2	13.0	12.3	13.4	15.1	12.3	13.6	13.6	13.0	13.4	12.8	13.5	12.4	13.4	13.7	15.3	15.0	15.2	14.5	13.6	13.6	16.3	17.9		14.3	8.5	13.0	13.3	13.3	13.6	13.4
	HGH/ABINIT	2.2	2.2	2.2	2.2	2.2	2.0	4.9	2.4	2.3	2.2	2.8	2.8	2.1	2.5	4.5	2.3	2.5	2.2	2.2	2.4	2.1	3.0	2.2	2.2	2.3	2.4	2.3	2.7	2.2	2.1	6.6	6.2	14.3		0.9	2.6	2.0	2.0	2.0	2.0
	HGH-NLCC/BigDFT	1.1	1.1	1.2	1.2	1.1	1.0	3.6	1.2	1.2</																															

# Other FPLAPW software

- Free to use, available at NSC  
\$ module avail name
- Different specialities, check websites

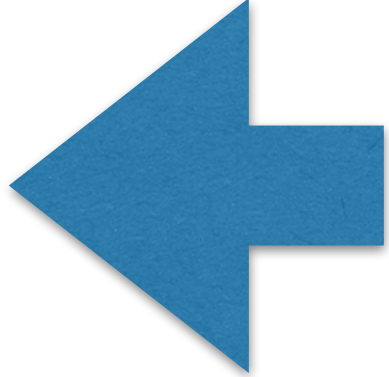


<http://exciting.wikidot.com/>



<https://elk.sourceforge.io/>

# WIEN2k @SNIC HPC centers

- **Tetralith** / Sigma, NSC, LiU  **this mini workshop**
- Kebnekaise, HPC2N, UmU
- Dardel, PDC, KTH *to be installed*
- ...

- **On many systems:** `$ module avail wien2k`  
`$ module spider wien2k`

# WIEN2k @NSC

- Only load WIEN2k module (not compilers etc.)
- Use latest version

```
[weiol@tetralith2 ~]$ module avail wien2k
```

```
----- /software/sse/modules -----  
WIEN2k/recommendation          (D)   WIEN2k/19.1-nsc1-intel-2018b-eb  
WIEN2k/18.2-nsc1-intel-2018a-eb  WIEN2k/21.1-nsc1-intel-2018b-eb  
WIEN2k/18.2-nsc2-intel-2018a-eb
```

Where:

D: Default Module

Use "module spider" to find all possible modules.

Use "module keyword key1 key2 ..." to search for all possible modules matching any of the "keys".

```
[weiol@tetralith2 ~]$ module add WIEN2k/21.1-nsc1-intel-2018b-eb
```

# Resources

- The WIEN2k user guide - Start here! Check carefully!
- Presentations (.pdf, video) - Very useful!
- The email list (search it!) - Similar problem was solved?
- Check around the website (useful links, FAQ, ...)
- P. Blaha et al., WIEN2k paper (2020)

**Questions/trouble running at NSC?** [support@nsc.liu.se](mailto:support@nsc.liu.se)

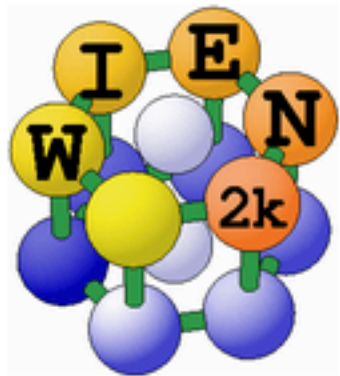


Wien 2k

wien2k.at

Adding a new dimension to DFT calculations of solids ...

# WIEN2k



**P. Blaha, K. Schwarz, G. K. H. Madsen, D. Kvasnicka, J. Luitz, R. Laskowski, F. Tran and L. D. Marks**

**Institute of Materials Chemistry, TU Vienna**

The program package WIEN2k allows to perform electronic structure calculations of solids using density functional theory (DFT). It is based on the full-potential (linearized) augmented plane-wave ((L)APW) + local orbitals (lo) method, one among the most accurate schemes for band structure calculations. WIEN2k is an all-electron scheme including relativistic effects and has many **features**. It has been licensed by more than 3400 user groups and has about 16000 citations on Google scholar (Blaha WIEN2k). A recent extensive overview on WIEN2k is given in:

**WIEN2k: An APW+lo program for calculating the properties of solids.**

P. Blaha, K.Schwarz, F. Tran, R. Laskowski, G.K.H. Madsen and L.D. Marks, J. Chem. Phys. 152, 074101 (2020)

**The current version is**

**WIEN2k\_21.1**

and was released on 14. April 2021. Version 21.1 is a major update, with several new features and many bugfixes. **Upgrading** from WIEN2k\_19 (or earlier) is highly recommended.

In the last years it became a tradition to have at least one **"WIEN-workshop"** every year, where new and experienced users can learn more about the code, get intensive hands-on training, interchange ideas and share experiences.

**CANCELLED due to the CORONA CRISIS**

This year the

**27. WIEN2k workshop**

will be held together with a

**"Blaha - Symposium" (40 years of WIEN2k code)**

WEN2k

(L)APW+lo

FEATURES

HARD+SOFT

ORDER INFO

PAPERS

REG USERS

WORKSHOPS

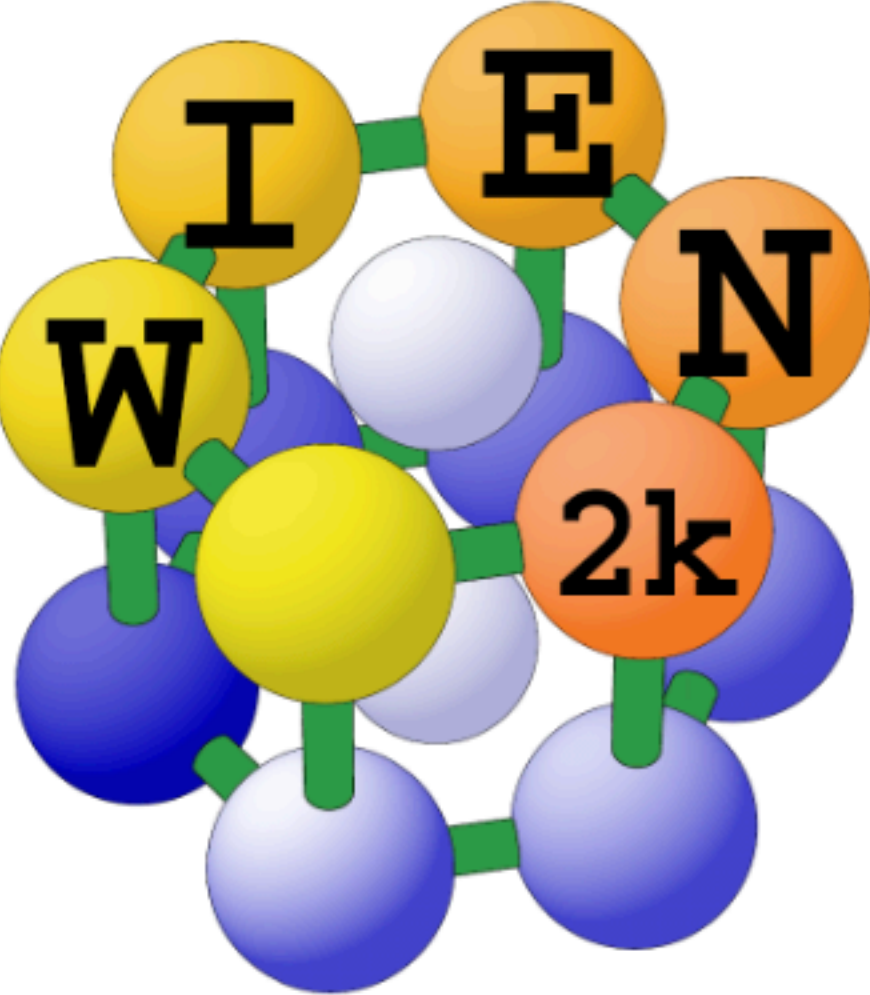
SUPPORTED BY  
NOMAD  
FOR EASY  
UPLOADING, STORING,  
RECOVERING, AND SHARING

usersguide.pdf

wien2k.at/reg\_user/textbooks/usersguide.pdf

(1 of 295) Automatic Zoom

- I Introduction to the WIEN2k package
- II Detailed description of the files and programs of the WIEN2k package
- III Installation of the WIEN2k package and Dimensioning of programs
- IV Appendix



# WIEN2k

An Augmented Plane Wave Plus Local Orbitals Program  
for Calculating Crystal Properties

**User's Guide, WIEN2k\_21.1 (Release 04/30/2021)**

[http://wien2k.at/reg\\_user/textbooks/usersguide.pdf](http://wien2k.at/reg_user/textbooks/usersguide.pdf)



# WIEN2k: An APW+lo program for calculating the properties of solids

Cite as: J. Chem. Phys. 152, 074101 (2020); doi: 10.1063/1.5143061

Submitted: 19 December 2019 • Accepted: 24 January 2020 •

Published Online: 19 February 2020



View Online



Export Citation



CrossMark

Peter Blaha,<sup>1,a)</sup>  Karlheinz Schwarz,<sup>1</sup>  Fabien Tran,<sup>1</sup>  Robert Laskowski,<sup>2</sup>  Georg K. H. Madsen,<sup>1</sup>   
and Laurence D. Marks<sup>3</sup> 

## AFFILIATIONS

<sup>1</sup>Institute of Materials Chemistry, Vienna University of Technology, Getreidemarkt 9/165-TC, A-1060 Vienna, Austria

<sup>2</sup>Institute of High Performance Computing, A\*STAR, 1 Fusionopolis Way, #16-16, Connexis 138632, Singapore

<sup>3</sup>Department of Materials Science and Engineering, Northwestern University, Evanston, Illinois 60208, USA

**Note:** This paper is part of the JCP Special Topic on Electronic Structure Software.

<sup>a)</sup>Author to whom correspondence should be addressed: [pblaha@theochem.tuwien.ac.at](mailto:pblaha@theochem.tuwien.ac.at)

## ABSTRACT

The WIEN2k program is based on the augmented plane wave plus local orbitals (APW+lo) method to solve the Kohn–Sham equations of density functional theory. The APW+lo method, which considers all electrons (core and valence) self-consistently in a full-potential treatment, is implemented very efficiently in WIEN2k, since various types of parallelization are available and many optimized numerical libraries can be used. Many properties can be calculated, ranging from the basic ones, such as the electronic band structure or

transport, ... )

- discussions and exchange of experience between users and developers
- Poster session
  - (You must send a camera-ready pdf-file (max 1 page) to [Peter Blaha](#).)

**[Here you can find the final program \(pdf\).](#)**

**Here you can download the talks and exercises in pdf format**

- [K. Schwarz: Introduction to DFT and LAPW](#)
- [P. Blaha: Getting started](#)
- [K. Schwarz: Magnetism](#)
- [F. Tran: Exchange-correlation functionals](#)
- [J. Tomczak: Dynamical mean field theory \(DMFT\)](#)
- [R. Laskowski: Relativistic effects and magnetism](#)
- [O. Rubel: Wannier functions, Berry Phases and Backfolding](#)
- [P. Blaha: XPS, Optics and XANES](#)
- [P. Blaha: XANES](#)
- [R. Laskowski: BSE](#)
- [P. Blaha: Hyperfine Interactions](#)
- [R. Laskowski: NMR](#)
- [G. K. H. Madsen: Transport \(BoltzTraP2\)](#)
- [M. Arrigoni: Point defects + Spinney](#)
- [P. Blaha: Installation and parallelization](#)
- [Exercises](#)

---

**General Information:**

**Dates: 13-17 August 2019**  
Begin of workshop: Tuesday morning (13. August at 9 a.m.)  
End of workshop: Saturday afternoon (17. August, 1 p.m. with open end exercises)

**Social event: Conference dinner at a Viennese "Heuriger": Friday evening, Aug 16, 2019**

**Conference site:**  
The workshop will take place at:  
TU Vienna, Campus Gusshausstrasse  
Gusshausstrasse 25, A-1040 Vienna, Austria  
Lecture room: Kontaktraum, 6<sup>th</sup> floor  
The TU Vienna is conveniently located close to the City Center and can be reached easily by public transportation (U2, U4, U1, Station Karlsplatz; or U1, Station Paulanergasse).

**Hands-on sessions:**  
Please bring your own **laptop + power cable + adapter** for European plug. Plugs and wlan connection will be provided.  
When you have a MS-Windows laptop, please install the [x2go client](#) on it.

<http://wien2k.at/events/ws2019/>



WIEN 2k

wien2k.at/onlineworkshop/

WIEN2k

(L)APW+lo

FEATURES

HARD+SOFT

ORDER INFO

PAPERS

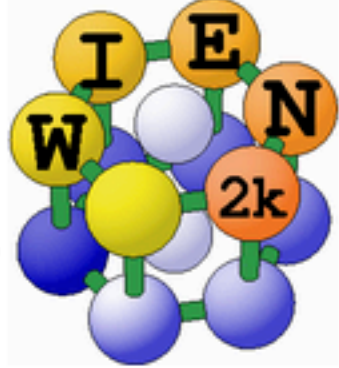
REG USERS

WORKSHOPS

SUPPORTED BY  
NOMAD  
FOR EASY  
UPLOADING, STORING,  
RECOVERING, AND SHARING

Adding a new dimension to DFT calculations of solids ...


## Lecture notes from the WIEN2k-workshop 2016 at McMasters University, Hamilton (Canada):



On this page you find the pdf presentations as well as the videos of this workshop. It is strongly recommended that newcomers to WIEN2k or solid state modelling checkout these files. But even for more experienced users might find useful tricks or procedures for certain tasks.


Pictures of the [participants](#), the [Niagara Falls](#) and the [lecturers](#)

Some impressions how the workshop went along:



workshop

- K.Schwarz: [Density functional theory \(DFT\) and the concepts of the augmented-plane-wave plus local orbitals \(APW+lo\) method](#)



DFT and

<http://wien2k.at/onlineworkshop/>



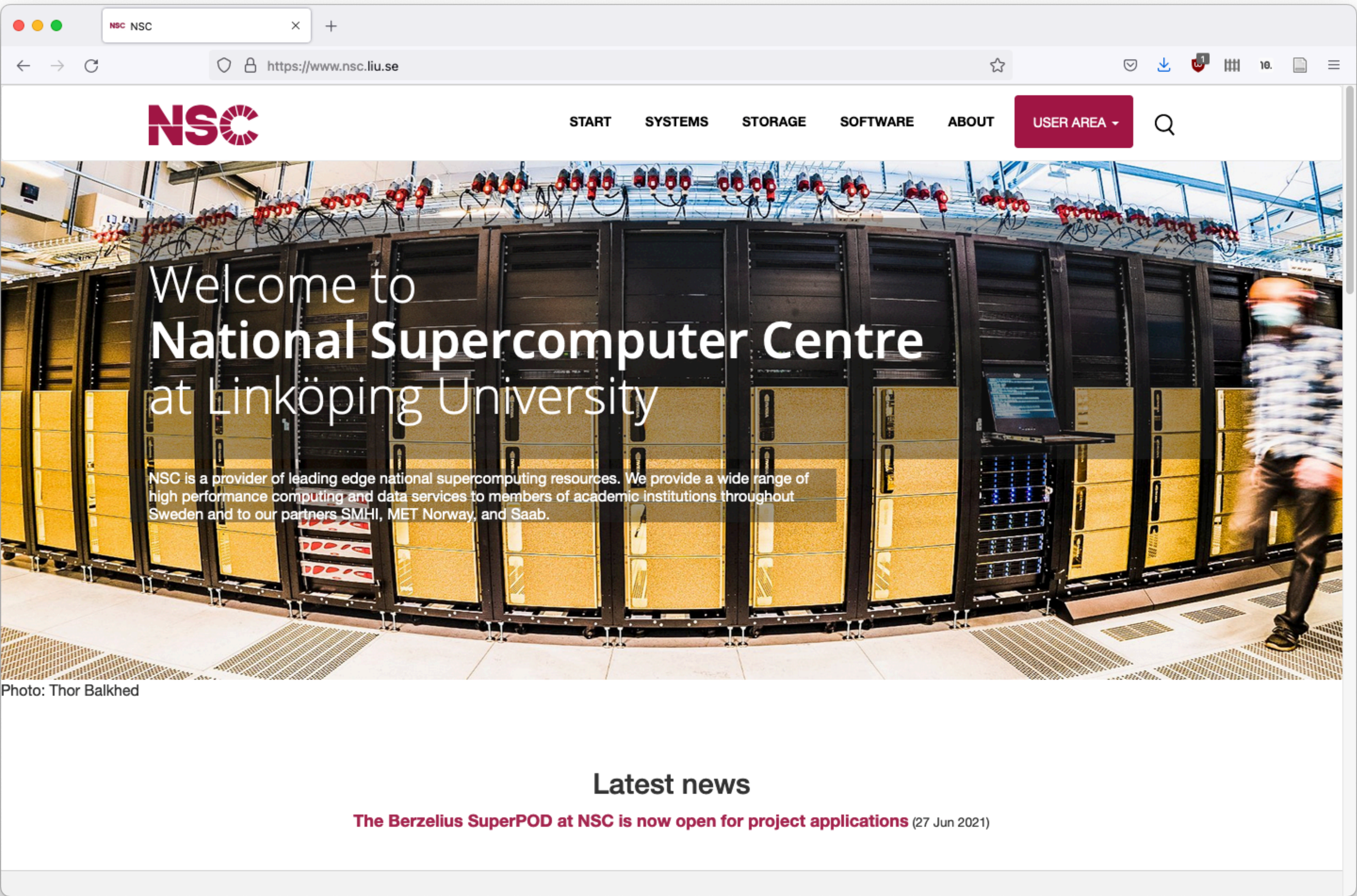


Photo: Thor Balkhed

### Latest news

**The Berzelius SuperPOD at NSC is now open for project applications** (27 Jun 2021)

<https://www.nsc.liu.se>



The screenshot shows a web browser window with the URL <https://www.nsc.liu.se/software/installed/tetralith/wien2k/>. The page features the NSC logo and a navigation menu with options: START, SYSTEMS, STORAGE, SOFTWARE, ABOUT, and a highlighted USER AREA dropdown. A search icon is also present.

The main content area displays a grid of software packages. The 'WIEN2K' package is highlighted with a red background. Below the grid, the text 'Gaussian and GaussView' is centered.

A breadcrumb trail reads: NSC / Software / Installed software / Tetralith & Sigma Software / WIEN2K.

## WIEN2K Installations on Tetralith & Sigma

The WIEN2k installations are generally maintained by Weine Olovsson ([weiol@nsc.liu.se](mailto:weiol@nsc.liu.se)).

"The program package **WIEN2k** allows to perform electronic structure calculations of solids using density functional theory (DFT). It is based on the full-potential (linearized) augmented plane-wave ((L)APW) + local orbitals (lo) method, one among the most accurate schemes for band structure calculations. WIEN2k is an all-electron scheme including relativistic effects and has many features."

To use the NSC installations one needs to be covered by a valid license, for more information [see this link](#).

**Note:** for a single k-point there's at the moment no recipe how to run over several nodes.

### Initial setup

There are some settings which are needed by WIEN2k. This can be done by running `userconfig` after loading the WIEN2k module, which will put the settings into `.bashrc`. These settings could also be put into a file which you source

<https://www.nsc.liu.se/software/installed/tetralith/wien2k/>

**Example batch script**

A batch script for running WIEN2k may look like below, here running MPI over a single node (32 cores). Also note that **a few cores might be enough**, depending on the job. See examples further below on how the jobscript can be tailored and also consider the benchmarks at the end of this page. The examples show how to set up the creation of appropriate **.machines** files used by WIEN2k for parallel calculations.

```
#!/bin/bash
#SBATCH -A SNIC-xxx-yyy
#SBATCH -J jobname
#SBATCH -N 1
#SBATCH --exclusive
#SBATCH -t 12:00:00

module add WIEN2k/21.1-nsc1-intel-2018b-eb
# export SCRATCH=$SNIC_TMP

# set .machines for parallel job
# lapw0 running on one node
echo -n "lapw0: " > .machines
echo -n $(hostlist -e $SLURM_JOB_NODELIST | tail -1) >> .machines
echo "$i:8" >> .machines

# run one mpi job on each node (splitting k-mesh over nodes)
for i in $(hostlist -e $SLURM_JOB_NODELIST)
do
  echo "1:$i:32 " >> .machines
done
echo granularity:1 >> .machines
echo extrafine:1 >> .machines

run_lapw -p

# if you need the vector files for continuing calculations, e.g.
# for DOS, spectrum etc. copy them to your current directory:
# cp $SNIC_TMP/* .
```

(Note that you should edit the jobname and the account number before submitting.)

**Available modules**

WIEN2k/21.1-nsc1-intel-2018b-eb

<https://www.nsc.liu.se/software/installed/tetralith/wien2k/>



NSC WIEN2K

← → ↻ <https://www.nsc.liu.se/software/installed/tetralith/wien2k/> ☆ 10. ☰

## Benchmark and scaling test on Tetralith

Here, I will demonstrate some results for running WIEN2k on the Tetralith cluster. The calculations below are using WIEN2k 18.2 with ELPA corresponding to the [WIEN2k/18.2-nsc1-intel-2018a-eb](#) module (though use the latest recommended one). They are for regular self-consistent DFT PBE calculations, setting RKMAX=7.

Disordered Ga<sub>2</sub>O<sub>3</sub> with 120 inequivalent atoms, 1 k-point (normalized 1 job on 4 cores, 50401 s).

```
grep :RKM *.scf
:RKM : MATRIX SIZE 26231L0s:1032 RKM= 7.00 WEIGHT= 1.00 PGR:
```

Disordered NaCl with 64 inequivalent atoms, 1 k-point (normalized 1 job on 1 core, 2013 s).

```
grep :RKM *.scf
:RKM : MATRIX SIZE 5048L0s: 416 RKM= 7.00 WEIGHT= 1.00 PGR:
```

FeOCl with 12 inequivalent atoms, 416 irreducible k-points (normalized 1 job k-point parallel on 8 cores, 12563 s).

```
grep :RKM *.scf
:RKM : MATRIX SIZE 3049L0s: 184 RKM= 7.00 WEIGHT= 4.00 PGR:
```

The first two cases are for larger systems where 1 k-point might be suitable for the self-consistent calculations, while the third case is for a smaller system with more k-points.

Jobs / Time	System / Configuration
~1.0	Disordered Ga <sub>2</sub> O <sub>3</sub> (120 atoms, 1 k-point)
~1.5	Disordered NaCl (64 atoms, 1 k-point)
~2.0	FeOCl (12 atoms, 416 k-points)

# 2. Setting up job

Weine Olovsson

National Supercomputer Centre (NSC), Linköping University

SNIC training, online mini workshop @NSC 18<sup>th</sup> Nov 2021




# Typical workflow

- Prepare **structure file**: `cif2struct`, `supercell`
- **Initialize** calculation: `init_lapw` & `w2web`
- Prepare job script & **submit**
- **Convergence**, any problems?
- Post-processing, other steps
  - DOS, band structure, ...

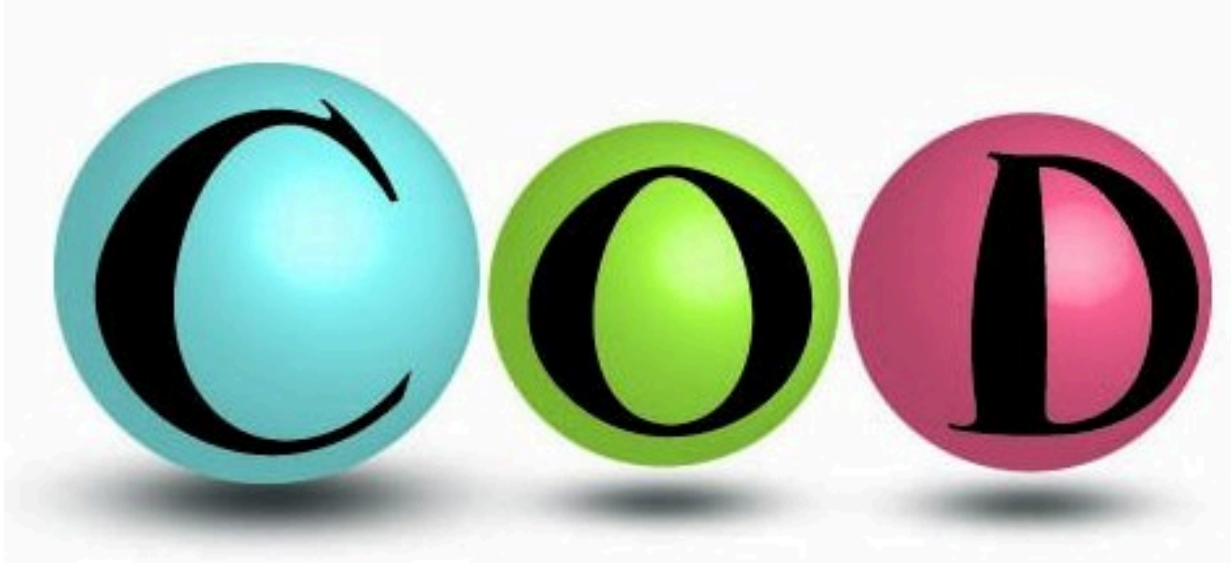


Crystallography Open Database X

https://www.crystallography.net/cod/



# Crystallography Open Database

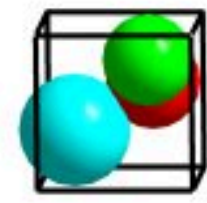


**Open-access collection of crystal structures of organic, inorganic, metal-organics compounds and minerals, excluding biopolymers.**

Including data and [software](#) from [CrystalEye](#), developed by Nick Day at the [department of Chemistry](#), the University of Cambridge under supervision of [Peter Murray-Rust](#).


All data on this site have been placed in the [public domain](#) by the contributors.

Currently there are **481012** entries in the COD.  
Latest deposited structure: [1565225](#) on **2021-11-15** at **10:38:58 UTC**



[CIFs Donators](#)

---



**Advisory Board**

Daniel Chateigner, Xiaolong Chen, Marco Ciriotti,  
Robert T. Downs, Saulius Gražulis, Werner Kaminsky, Armel Le Bail, Luca Lutterotti,  
Yoshitaka Matsushita, Andrius Merkys, Peter Moeck, Peter Murray-Rust, Miguel Quirós Olozábal,  
Hareesh Rajan, Alexandre F.T. Yokochi

<https://www.crystallography.net/cod/>

experimental structures in .cif format!

# Initialize calculation

- Some basic settings needed:

```
export EDITOR="vi"  
export SCRATCH=./
```

```
$ module load WIEN2k/21.1-nsc1-intel-2018b-eb  
$ mkdir test; cp test.struct test; cd test  
$ init_lapw
```

- Sets up structure symmetry and input files  
    sgroup **important for correct symmetry!**
- Quite many files generated!



# Example: h-BN supercell setup

- From .cif to .struct
- Reduce .struct symmetry (MULT=1)
- Create a 4x4x2 supercell
- If interested in 1st B atom position, break supercell symmetry by tagging it in .struct

# init\_lapw

```
| [weiol@tetralith1 t1]$ init_lapw
| next is setrmt
| Automatic determination of RMTs. Please specify the desired RMT reduction
| compared to almost touching spheres.
| Typically, for a single calculation just hit enter, for force minimization
| use 1-5; for volume effects you may need even larger reductions.
|
| Enter reduction in %
| enter
| Use old or new scheme (o/N)
| enter
|
| specify nn-bondlength factor: (usually=2) [and optionally dlimit, dstmax (about
| 1.d-5, 20)]
| DSTMAX: 27.9450136800000
| iix,iiy,iiz 2 2 2 37.8548400000000
| 37.8548400000000 50.3513760000000
| NAMED ATOM: B 1 Z changed to IATNR+1000 to determine equivalency
|
| ERROR !!!!!!!!!!!!!!!
| RMT( 1)=2.00000 AND RMT(100)=2.00000
| SUMS TO 4.00000 GT NNN-DIST= 2.73194
|
| ...
|
| WARNING: Mult not equal. PLEASE CHECK outputnn-file
| WARNING: Mult not equal. PLEASE CHECK outputnn-file
|
| ...
```

# nn

```
| NN created a new t1.struct_nn file
| NN created a new CASE.STRUCT_NN FILE
| 0.347u 0.019s 0:00.44 79.5%0+0k 0+0io 0pf+0w
| atom Z    RMT-max  RMT
| 1    5.0    1.29    1.29
| 2    5.0    1.29    1.29
|
| ...
| 126   7.0    1.43    1.43
| 127   7.0    1.43    1.43
| 128   7.0    1.43    1.43
| file   t1.struct_setrmt  generated
| Do you want to accept these radii; discard them; or rerun setRmt (a/d/r):
| enter
| next is nn
| > nn(16:20:46) specify nn-bondlength factor: (usually=2) [and optionally dlimit, dstmax
| (about
| 1.d-5, 20)]
| enter
```

# case.outputnn

```
| opens a new file "t1.outputnn" in vi
|
| blebleble
| H                128
|      RELA
| 18.927420 18.927420 25.175688 90.000000 90.000000 120.000000
|      1      0.08333333  0.16666667  0.12500000
|      1
| NAMED ATOM: B 1      Z changed to IATNR+1000 to determine equivalency
| B 1      NPT= 781  R0=0.00010000 RMT=  1.2900  Z: 5.0
|      1.0000000 0.0000000 0.0000000
|      0.0000000 1.0000000 0.0000000
|      0.0000000 0.0000000 1.0000000
|      2      0.33333333  0.16666667  0.12500000
|      1
| B      NPT= 781  R0=0.00010000 RMT=  1.2900  Z: 5.0
|      1.0000000 0.0000000 0.0000000
|      0.0000000 1.0000000 0.0000000
|      0.0000000 0.0000000 1.0000000
|      3      0.58333333  0.16666667  0.12500000
|      1
| B      NPT= 781  R0=0.00010000 RMT=  1.2900  Z: 5.0
|      1.0000000 0.0000000 0.0000000
|      0.0000000 1.0000000 0.0000000
|      0.0000000 0.0000000 1.0000000
|      4      0.83333333  0.16666667  0.12500000
|      1
| ...
| quit file with :q
```



# sgroup

```
.....  
|  
| WARNING: Mult not equal. PLEASE CHECK outputnn-file  
| WARNING: ityp not equal. PLEASE CHECK outputnn-file  
|  
| NN created a new t1.struct_nn file  
| NN created a new CASE.STRUCT_NN FILE  
| 0.358u 0.010s 0:45.45 0.7%0+0k 0+0io 0pf+0w  
|-----> check in t1.outputnn for overlapping spheres,  
|          coordination and nearest neighbor distances  
| nn has created a new struct file with different multiplicities  
|-----> DO YOU WANT TO USE THE NEW t1.struct_nn file (y/n)  
| enter  
|-----> continue with sgroup or edit the t1.struct file (c/e)  
| enter
```

# case.outputsgroup

```
| opens a new file "t1.outputsgroup" in vi
|
| warning: !!! Number of inequivalent atoms has changed.
|           !!! Old value= 128      New value= 30
|
| warning: !!! Struct file is not consistent with space group found.
|
|-----|
| Bravais lattice: Hexagonal
|
|      a           b           c
| 18.92742000    18.92742000    25.17568800
|      alpha      beta      gamma
| 90.00000000    90.00000000    120.00000000
|
|===== Decomposition of new basis vectors over input basis ====
| 1.000000    0.000000    0.000000    <--- 1
| 0.000000    1.000000    0.000000    <--- 2
| 0.000000    0.000000    1.000000    <--- 3
|
|===== Number of atoms in cell: 128
|===== Atom positions:
|
| 0.00000000    0.00000000    0.00000000
| B1
| ...
| quit file with :q
```

# sgroup: output

```
next is sgroup
> sgroup (16:24:17) 0.002u 0.006s 0:00.00 0.0%0+0k 0+0io 0pf+0w
Names of point group: -6m2 -6m2 D3h
Names of point group: m m Cs
Names of point group: mm2 mm2 C2v
Names of point group: mm2 mm2 C2v
Names of point group: mm2 mm2 C2v
...
Names of point group: m m Cs
Names of point group: mm2 mm2 C2v
Names of point group: mm2 mm2 C2v
Names of point group: -6m2 -6m2 D3h
!warning: !!! Struct file is not consistent with space group found.
!Number and name of space group: 187 (P -6 m 2)
!warning: !!! Number of inequivalent atoms has changed.
! !!! Old value= 128 New value= 30
!warning: !!! Struct file is not consistent with space group found.
!-----> check in t1.outputsgroup for proper symmetry, compare
! with your struct file and later with t1.outputs
! sgroup has also produced a new struct file based on your old one.
! If you see warnings above, consider to use the newly generated
! struct file, which you can view (edit) now.
!-----> continue with symmetry (old case.struct) or use/edit t1.struct_sgroup ? (c/e)
e
```

- **edit** to use sgroup suggestion, check file & quit with :q (if using vi)



# sgroup: new struct

```
|-----> Do you want to use the new struct file ? (y/n)  
|y
```

- accept new struct file (otherwise old one used)

```
|-----  
| patchsymm has found inconsistencies in your struct file  
| it comes probably from rounding errors in the last digits of your positions  
| Site 39 Ops 2 New: 0.58333333 0.16666666 0.25000000 Diff: -0.10000D-07 0.00000D+00 0.00000D+00 <  
| Site 40 Ops 2 New: 0.83333334 0.41666667 0.25000000 Diff: 0.00000D+00-0.10000D-07 0.00000D+00 <  
| Site 41 Ops 2 New: 0.58333333 0.41666667 0.25000000 Diff: 0.10000D-07 0.10000D-07 0.00000D+00 <  
| Site 42 Ops 2 New: 0.58333333 0.16666666 0.75000000 Diff: -0.10000D-07 0.00000D+00 0.00000D+00 <  
| Site 43 Ops 2 New: 0.83333334 0.41666667 0.75000000 Diff: 0.00000D+00-0.10000D-07 0.00000D+00 <  
|  
| ...  
| Site 44 12 Ops 12 New: 0.58333333 0.41666667 0.75000000 Diff: 0.10000D-07 0.10000D-07 0.00000D+00 <  
| Site 106 23 Ops 12 New: 0.33333334 0.16666667 0.00000000 Diff: 0.11102D-15 0.10000D-07 0.00000D+00 <  
| Site 107 23 Ops 12 New: 0.83333333 0.16666667 0.00000000 Diff: -0.10000D-07-0.10000D-07 0.00000D+00 <  
| Site 108 23 Ops 12 New: 0.83333333 0.66666666 0.00000000 Diff: 0.10000D-07-0.22204D-15 0.00000D+00 <  
| Site 122 28 Ops 12 New: 0.33333334 0.16666667 0.50000000 Diff: 0.11102D-15 0.10000D-07 0.00000D+00 <  
| Site 123 28 Ops 12 New: 0.83333333 0.16666667 0.50000000 Diff: -0.10000D-07-0.10000D-07 0.00000D+00 <  
| Site 124 28 Ops 12 New: 0.83333333 0.66666666 0.50000000 Diff: 0.10000D-07-0.22204D-15 0.00000D+00 <  
| do you want to use t1.struct_patchsymm ? (Y/n)  
|enter
```

- symmetry problem!? Accept patch (usually)  restarts from beginning!

# sgroup: again

```
| next is sgroup  
| > sgroup (16:32:11) 0.002u 0.004s 0:00.01 0.0%0+0k 0+0io 0pf+0w  
| Names of point group: -6m2 -6m2 D3h  
| Names of point group: m m Cs  
| Names of point group: mm2 mm2 C2v  
|  
| ...  
|  
| Number and name of space group: 187 (P -6 m 2)  
|-----> check in t1.outputsgroup for proper symmetry, compare  
| with your struct file and later with t1.outputs  
| sgroup has also produced a new struct file based on your old one.  
| If you see warnings above, consider to use the newly generated  
| struct file, which you can view (edit) now.  
|-----> continue with symmetry (old case.struct) or use/edit t1.struct_sgroup ? (c/e)  
| enter  
| next is symmetry  
| > symmetry (16:32:16) SPACE GROUP DOES NOT CONTAIN INVERSION  
| 0.020u 0.012s 0:00.07 42.8%0+0k 0+0io 0pf+0w  
|-----> check in t1.outputs the symmetry operations,  
| the point symmetries and compare with results from sgroup  
| if you find errors (often from rounding errors of positions), apply x patchsymm  
|-----> continue with lstart or edit the t1.struct_st file (c/e/x)  
| enter
```

- this time struct file seems OK!  ...if not, might restart cycle again

# lstart

```
next is lstart
CREATE A NEW t1.inst FILE with PROPER ATOMS
If necessary specify switches for instgen_lapw (or press ENTER):
-up (default) -dn -nm (non-magnetic) -ask
enter
30 Atoms found: with labels B 1 B 2 B 3 B 4 B 5 B 6 B 7 B 8 B 9 B 10 B 11 B 12 B 13 B
14 B 15 N 1 N 2 N 3 N 4 N 5 N 6 N 7 N 8 N 9 N 10 N 11 N 12 N 13 N 14 N 15
|generate atomic configuration for atom 1 : B 1
|generate atomic configuration for atom 2 : B 2
|generate atomic configuration for atom 3 : B 3
|
|...
|generate atomic configuration for atom 29 : N 14
|generate atomic configuration for atom 30 : N 15
|> lstart (16:33:53) SELECT XCPOT:
| recommended: PBE [(13) GGA of Perdew-Burke-Ernzerhof 96]
| LDA [( 5)]
| WC [(11) GGA of Wu-Cohen 2006]
| PBESOL [(19) GGA of Perdew etal. 2008]
|13
| SELECT ENERGY to separate core and valence states:
| recommended: -6.0 Ry (check how much core charge leaks out of MT-sphere)
| ALTERNATIVELY: specify charge localization
| (between 0.97 and 1.0) to select core state
|-6
```

- selected PBE for XC and recommended energy separation between core/valence



# case.outputst

opens a new file "t1.outputst" in vi

...

	E-up(Ry)	E-dn(Ry)	Occupancy	q/sphere	core-state
1S	-13.270546	-13.249235	1.00 1.00	0.9989	T
2S	-0.728058	-0.639738	1.00 1.00	0.2252	F
2P*	-0.295469	-0.216441	1.00 0.00	0.2162	F

TOTAL CORE-CHARGE: 2.000000  
TOTAL CORE-CHARGE INSIDE SPHERE: 1.997822  
TOTAL CORE-CHARGE OUTSIDE SPHERE: 0.002178

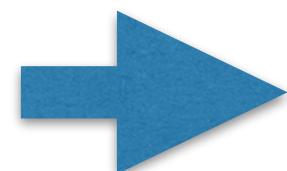
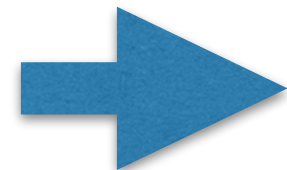
...

	E-up(Ry)	E-dn(Ry)	Occupancy	q/sphere	core-state
1S	-28.235802	-28.124001	1.00 1.00	1.0000	T
2S	-1.460885	-1.124776	1.00 1.00	0.6338	F
2P*	-0.610907	-0.304312	1.00 0.00	0.5866	F
2P	-0.609450	-0.303079	2.00 0.00	0.5858	F

TOTAL CORE-CHARGE: 2.000000  
TOTAL CORE-CHARGE INSIDE SPHERE: 1.999991  
TOTAL CORE-CHARGE OUTSIDE SPHERE: 0.000009

...

quit file with :q



# Continue...

```
-----  
| Atomic configuration for atom: B 1   Z=   5.00  
|      E-up(Ry)      E-dn(Ry)  Occupancy  q/sphere  core-state  
|  1S      -13.270546   -13.249235   1.00   1.00    0.9989   T  
|  2S      -0.728058    -0.639738   1.00   1.00    0.2252   F  
|  2P*     -0.295469    -0.216441   1.00   0.00    0.2162   F  
|  
| Atomic configuration for atom: N 1   Z=   7.00  
|      E-up(Ry)      E-dn(Ry)  Occupancy  q/sphere  core-state  
|  1S      -28.235802   -28.124001   1.00   1.00    1.0000   T  
|  2S      -1.460885    -1.124776   1.00   1.00    0.6338   F  
|  2P*     -0.610907    -0.304312   1.00   0.00    0.5866   F  
|  2P      -0.609450    -0.303079   2.00   0.00    0.5858   F  
| LSTART ENDS  
| 1.101u 0.035s 0:47.38 2.3%0+0k 0+0io 0pf+0w  
| -----> continue with kgen or edit the t1.inst file and rerun lstart (c/e)  
| enter  
|-----
```

- After this step, the setup of case.in1/in1c and case.in2/in2c starts

# case.in1\_st

opens a new file "t1.in1\_st" in vi

```
| WFFIL EF= 0.50000          (WFFIL, WFPRI, ENFIL, SUPWF)
| 7.00      10  4  ELPA pxq BL 64 (R-MT*K-MAX,MAX L IN WF,V-NMT,LIB)
| 0.30      2  0          (GLOBAL E-PARAMETER WITH n OTHER CHOICES, global APW/LAPW)
| 0  0.30      0.0000 CONT 1
| 1  0.30      0.0000 CONT 1
| 0.30      2  0          (GLOBAL E-PARAMETER WITH n OTHER CHOICES, global APW/LAPW)
| 0  0.30      0.0000 CONT 1
| 1  0.30      0.0000 CONT 1
| 0.30      2  0          (GLOBAL E-PARAMETER WITH n OTHER CHOICES, global APW/LAPW)
| 0  0.30      0.0000 CONT 1
| 1  0.30      0.0000 CONT 1
| 0.30      2  0          (GLOBAL E-PARAMETER WITH n OTHER CHOICES, global APW/LAPW)
| 0  0.30      0.0000 CONT 1
| 1  0.30      0.0000 CONT 1
| ...
| 0.30      3  0          (GLOBAL E-PARAMETER WITH n OTHER CHOICES, global APW/LAPW)
| 0  0.30      0.0000 CONT 1
| 0 -1.16      0.0010 CONT 1
| 1  0.30      0.0000 CONT 1
| 0.30      3  0          (GLOBAL E-PARAMETER WITH n OTHER CHOICES, global APW/LAPW)
| 0  0.30      0.0000 CONT 1
| 0 -1.16      0.0010 CONT 1
| 1  0.30      0.0000 CONT 1
| K-VECTORS FROM UNIT:4  -9.0      1.5  1029  emin / de (emax=Ef+de) / nband
|
| quit file with :q
```



# case.in1\_st

opens a new file "t1.in1\_st" in vi

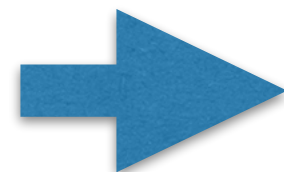
```
WFFIL EF= 0.50000 (WFFIL, WFPRI, ENFIL, SUPWF)
| 7.00 10 4 ELPA pxq BL 64 (R-MT*K-MAX,MAX L IN WF,V-NMT,LIB)
| 0.30 2 0 (GLOBAL E-PARAMETER WITH n OTHER CHOICES, global APW/LAPW)
| 0 0.30 0.0000 CONT 1
| 1 0.30 0.0000 CONT 1
| 0.30 2 0 (GLOBAL E-PARAMETER WITH n OTHER CHOICES, global APW/LAPW)
| 0 0.30 0.0000 CONT 1
| 1 0.30 0.0000 CONT 1
| 0.30 2 0 (GLOBAL E-PARAMETER WITH n OTHER CHOICES, global APW/LAPW)
| 0 0.30 0.0000 CONT 1
| 1 0.30 0.0000 CONT 1
| 0.30 2 0 (GLOBAL E-PARAMETER WITH n OTHER CHOICES, global APW/LAPW)
| 0 0.30 0.0000 CONT 1
| 1 0.30 0.0000 CONT 1
```

...

```
0.30 3 0 (GLOBAL E-PARAMETER WITH n OTHER CHOICES, global APW/LAPW)
0 0.30 0.0000 CONT 1
0 -1.16 0.0010 CONT 1
1 0.30 0.0000 CONT 1
0.30 3 0 (GLOBAL E-PARAMETER WITH n OTHER CHOICES, global APW/LAPW)
0 0.30 0.0000 CONT 1
0 -1.16 0.0010 CONT 1
1 0.30 0.0000 CONT 1
K-VECTORS FROM UNIT:4 -9.0 1.5 1029 emin / de (emax=Ef+de) / nband
```

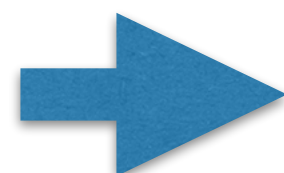
quit file with :q

RKMAX



ELPA /  
SCALA

Energy  
range



# case.in2\_st

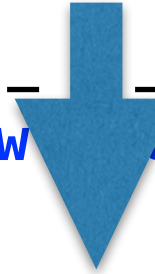
opens a new file "t1.in2\_st" in vi

```
|TOT          (TOT,FOR,QTL,EFG,FERMI)
| -12.0    512.0    0.50 0.05  1  EMIN, NE, ESEPERMIN, ESEPER0, iqtlsave
|TETRA      0.000    (GAUSS,ROOT,TEMP,TETRA,ALL      eval)
|  0 0  2 0  3 3  4 0  5 3  6 0  6 6
|  0 0  1 1 -1 1  2 0  2 2 -2 2  3 1 -3 1  3 3 -3 3  4 0  4 2 -4 2  4 4 -4 4  5 1 -5 1  5 3 -5 3
|5 5 -5 5  6 0  6 2 -6 2  6 4 -6 4  6 6 -6 6
|  0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
|  0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
|  0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
|  0 0  2 0  3 3  4 0  5 3  6 0  6 6
|
|...
|  0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
|  0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
|  0 0  2 0  3 3  4 0  5 3  6 0  6 6
|  0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
|  0 0  1 1 -1 1  2 0  2 2 -2 2  3 1 -3 1  3 3 -3 3  4 0  4 2 -4 2  4 4 -4 4  5 1 -5 1  5 3 -5 3
|5 5 -5 5  6 0  6 2 -6 2  6 4 -6 4  6 6 -6 6
|  0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
|  0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
|  0 0  2 0  3 3  4 0  5 3  6 0  6 6
| 12.00          GMAX
|NOFILE          FILE/NOFILE write recprlist
```

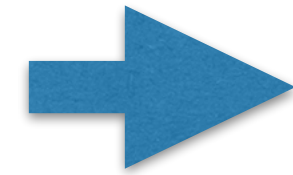
quit file with :q

No. of electrons

# case.in2\_st

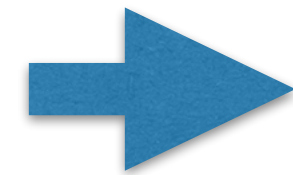


Fermi-  
method



```
opens a new file "t1.in2_st" in vi
|TOT          (TOT, FOR, QTL, EFG, FERMI)
| -12.0      512.0  0.50 0.05  1  EMIN, NE, ESEPERMIN, ESEPER0, iqtlsave
|TETRA      0.000  (GAUSS, ROOT, TEMP, TETRA, ALL eval)
| 0 0  2 0  3 3  4 0  5 3  6 0  6 6
| 0 0  1 1 -1 1  2 0  2 2 -2 2  3 1 -3 1  3 3 -3 3  4 0  4 2 -4 2  4 4 -4 4  5 1 -5 1  5 3 -5 3
|5 5 -5 5  6 0  6 2 -6 2  6 4 -6 4  6 6 -6 6
| 0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
| 0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
| 0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
| 0 0  2 0  3 3  4 0  5 3  6 0  6 6
|...
| 0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
| 0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
| 0 0  2 0  3 3  4 0  5 3  6 0  6 6
| 0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
| 0 0  1 1 -1 1  2 0  2 2 -2 2  3 1 -3 1  3 3 -3 3  4 0  4 2 -4 2  4 4 -4 4  5 1 -5 1  5 3 -5 3
|5 5 -5 5  6 0  6 2 -6 2  6 4 -6 4  6 6 -6 6
| 0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
| 0 0  1 0  2 0  2 2  3 0  3 2  4 0  4 2  4 4  5 0  5 2  5 4  6 0  6 2  6 4  6 6
| 0 0  2 0  3 3  4 0  5 3  6 0  6 6
|12.00      GMAX
|NOFILE      FILE/NOFILE write recprlist
|quit file with :q
```

GMAX





# Continue...

```
-----> in t1.in1_st select RKmax ( usually 5.0 - 9.0 ), LVNS and L0s
-----> in t1.in2_st select GMAX and Fermi-Energy method
> inputfiles prepared(16:41:50)
| inputfiles prepared
| inversion is NOT present
> inputfiles for lapw1c/2c prepared, no inversion present(16:41:50)
| next is kgen
> kgen(16:41:50) 12 symmetry operations without inversion
| inversion added (non-spinpolarized non-so calculation)
| NUMBER OF K-POINTS IN WHOLE CELL: (0 allows to specify 3 divisions of G)
1
-----
| opens the file "t1.klist"
| 1 0 0 0 1 1.0 -7.0 1.5 1 k, div: ( 1 1 1)
| END
| quit file with :q
-----
| length of reciprocal lattice vectors: 0.383 0.383 0.250 1.154 1.154 0.751
| 1 k-points generated, ndiv= 1 1 1
| 2D-BZ: better use TEMP in case.in2
| KGEN ENDS
| 0.001u 0.008s 0:29.49 0.0%0+0k 0+0io 0pf+0w
| -----> check in t1.klist number of generated K-points
| WARNING: 2D-BZ, better use TEMP and not TETRA in case.in2
| -----> continue with dstart or execute kgen again or exit (c/e/x)
| enter
| next is dstart
|> dstart -c -p > & .mist(16:44:06)
```

# case.outputd

opens a new file "t1.outputd" in vi

---

## STRUCTURAL INFORMATION

---

```
SUBSTANCE                = blebleble

LATTICE                   = H
LATTICE CONSTANTS ARE    = 18.9274200  18.9274200  25.1756880
NUMBER OF ATOMS IN UNITCELL = 30
MODE OF CALCULATION IS   =
convention (i,j)        (j)
                        ax ay az
                        (i) bx by bz
                        cx cy cz
```

...

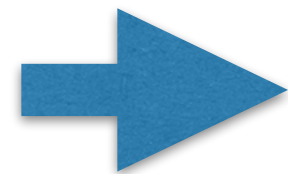
```
rmt(min)*kmax = 7.00000
lgmin = 10.85271
lgmax = 12.00000
```

```
!cutoff distance for superposition: 14.707
!unitcell range: 2 2 2
```

...

quit file with :q

GMAX >  
GMIN ?



# ...and finishing

```
| C T F  
|DSTART ENDS  
|8.383u 0.087s 0:09.21 91.8%0+0k 0+0io 0pf+0w  
|  
|-----> check in t1.outputd if gmax > gmin, normalization  
|-----> new t1.in0 generated  
|-----> do you want to perform a spinpolarized calculation ? (n/y)  
|n  
|  init_lapw finished ok
```

- sgroup finds correct symmetry? Real or complex (case.in1/in1c)?  
- complex runs: don't forget -c flag!
- Select XC function & c/v e- separation energy, core charge leak?
- **RKMAX** = 5 - 9 (H atom: 3 - 5) - bad scaling!
- Fermi-method: usually **TEMP 0.002**
- increase GMAX if warning in case.outputd
- k-points: shift or no shift? - DOS, spectroscopy, no shift!



# Files & :log

```
[weiol@tetralith1 t1]$ ls
dstart.def      t1.bva      t1.in2_ls    t1.inq_st    t1.outputs   t1.rspup     t1.struct_sgroup
dstart.error    t1.clmsum   t1.in2_st    t1.inst      t1.outputsgroup t1.sigma     t1.struct_st
kgen.def        t1.dayfile  t1.in2_sy    t1.kgen      t1.outputsgroup1 t1.sptdn     t1.test
:log            t1.in0      t1.inc       t1.klist     t1.outputst   t1.sptup     t1.vspdn_st
lstart.def      t1.in0_st   t1.inc_st    t1.nnshells  t1.potdn     t1.struct    t1.vsp_st
new_super.clmsum t1.in0_std  t1.inm       t1.outputd   t1.potup     t1.struct_nn
nn.def          t1.in1c     t1.inm_restart_st t1.outputkgen t1.rsigma    t1.struct_orig
patchsymm.def   t1.in1_st   t1.inm_st    t1.outputnn  t1.rsp       t1.struct_patchsymm
symmetry.def    t1.in2c     t1.inq       t1.outputpatch t1.rspdn     t1.struct_setrmt

[weiol@tetralith1 t1]$ cat :log
|> (init_lapw) options:
|Wed Nov 17 16:18:27 CET 2021> (x_lapw) nn -f t1
|Wed Nov 17 16:20:46 CET 2021> (x) nn
|Wed Nov 17 16:24:17 CET 2021> (x) sgroup
|Wed Nov 17 16:29:37 CET 2021> (x) patchsymm
|Wed Nov 17 16:30:29 CET 2021> (x) nn
|Wed Nov 17 16:31:45 CET 2021> (x) sgroup
|Wed Nov 17 16:31:55 CET 2021> (x) patchsymm
|Wed Nov 17 16:32:00 CET 2021> (x) nn
|Wed Nov 17 16:32:11 CET 2021> (x) sgroup
|Wed Nov 17 16:32:16 CET 2021> (x) symmetry
|Wed Nov 17 16:33:53 CET 2021> (x) lstart
|Wed Nov 17 16:41:50 CET 2021> (x) kgen
|Wed Nov 17 16:44:06 CET 2021> (x) dstart -c -p
```

# Quick initialization

- Can use if `case.struct` already properly setup

```
$ init_lapw -h - check options
```

- Example: Al unit cell, 4000 k-pts, more states in core

```
$ init_lapw -b -red 0 -numk 4000 -ecut -4.5
```

- Example: NaCl 64 atoms perturbed supercell, 1 k-pt

```
$ init_lapw -b -red 0 -numk 1 -fermit 0.002
```

# w2web

- The Wien2k Graphical User Interface (GUI)
- GUI alternative to scripts, such as `init_lapw`
- Good start for first time users
- Tool for structure setup
- Generate input files, check post-processing
- With experience, use scripts



# w2web @NSC

- Don't use for heavy work! (login node)
- Login with [ThinLinc](#) virtual desktop & open terminal
- First time, set up with username & **unique** password
- Open link in **Firefox**    **Applications > Internet > Firefox**
- When finished, close browser & w2web

```
$ module load WIEN2k/21.1-nsc1-intel-2018b-eb  
$ w2web
```

```
$ ps -ef | grep w2web  
$ kill -9 process-id
```

- find your w2web process & remove

test1@tetralith1.nsc.liu.se — Mozilla Firefox

test1@tetralith1.nsc.liu.se x +

tetralith1.nsc.liu.se:7895/index.pl?SID=507956

Session: **test1** 15:57:36 idle  
/home/weiol/WIEN2k/21-1/test1 [refresh] [no refresh]

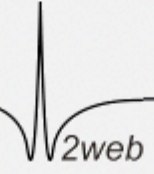
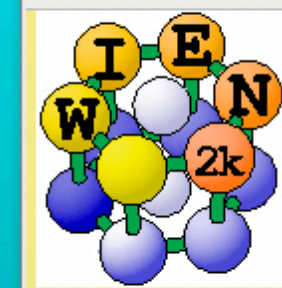
### w2web, the fully web-enabled interface to WIEN2k

**Session Name:** test1  
**Session ID:** 507956  
**Directory:** /home/weiol/WIEN2k/21-1/test1  
**Last changed:** Mon Nov 8 14:50:18 2021  
**Comments:**

- spin polarized calculation
- AFM calculation
- complex calculation (no inversion)
- parallel calculation

---

w2web © luitz.at

- [Execution >>]
- [StructGen™]
- [initialize calc.]
- [run SCF]
- [single prog.]
- [optimize(V,c/a)]
- [mini. positions]
- [Utils. >>]
- [<< Tasks]
- [El. Dens.]
- [DOS]
- [XSPEC]
- [TELNES3]
- [OPTIC]
- [Bandstructure]
- [Files >>]
- [struct file(s)]
- [input files]
- [output files]
- [SCF files]
- [Session Mgmt. >>]
- [change session]
- [change dir.]
- [change info]
- [Configuration]
- Usersguide
- [html-Version]
- [pdf-Version]

Idea and realization by  
[luitz.at] © 2001-2006

- Trash
- Home
- Figure\_1.png
- Old Firefox Data





# 3. Running job

Weine Olovsson

National Supercomputer Centre (NSC), Linköping University

SNIC training, online mini workshop @NSC 18<sup>th</sup> Nov 2021



# Before running

- Initialization steps OK?
- Computational demand?
- Several steps with new/updated input files?

# Small calculations

- Few atoms, DFT
- Run as serial job (or few cores) \$ run\_lapw
- Switch from **ELPA** to **ScaLAPACK** in case.in1

```
-----  
|WFFIL EF=.49676693479561029399 (WFFIL, WFPRI, ENFIL, SUPWF)  
| 7.00 10 4 ELPA pxq BL 64 (R-MT*K-MAX,MAX L IN WF,V-NMT,LIB)  
| 0.30 2 0 (GLOBAL E-PARAMETER WITH n OTHER CHOICES, global APW/LAPW)  
| 0 0.30 0.0000 CONT 1  
| 1 0.30 0.0000 CONT 1  
|K-VECTORS FROM UNIT:4 -7.5 1.5 5 emin / de (emax=Ef+de) / nband  
-----
```

```
-----  
|WFFIL EF=.49676693479561029399 (WFFIL, WFPRI, ENFIL, SUPWF)  
| 7.00 10 4 SCALA pxq BL 64 (R-MT*K-MAX,MAX L IN WF,V-NMT,LIB)  
| 0.30 2 0 (GLOBAL E-PARAMETER WITH n OTHER CHOICES, global APW/LAPW)  
| 0 0.30 0.0000 CONT 1  
| 1 0.30 0.0000 CONT 1  
|K-VECTORS FROM UNIT:4 -7.5 1.5 5 emin / de (emax=Ef+de) / nband  
-----
```

# Heavy calculations

- Many inequivalent atoms, large basis set
- GW, hybrid calcs.
- Is it reasonable to run?
- Level of approximations
- Running efficiently (parallelization)



# Job scripts: MPI

```
#!/bin/bash
#SBATCH -A SNIC-xxx-yyy
#SBATCH -N 1
#SBATCH --exclusive
#SBATCH -t 12:00:00

module add WIEN2k/12.1-nsc1-intel-2018b-eb
# export SCRATCH=$SNICTMP

# set .machines for parallel job
# lapw0 running on one node
echo -n "lapw0: " > .machines
echo -n $(hostlist -e $SLURM_JOB_NODELIST | tail -1) >> .machines
echo "$i:8" >> .machines

# run one mpi job on each node (splitting k-mesh over nodes)
for i in $(hostlist -e $SLURM_JOB_NODELIST)
do
    echo "1:$i:32 " >> .machines
done
echo granularity:1 >> .machines
echo extrafine:1 >> .machines

run_lapw -p

# if you need the vector files for continuing calculations, e.g.
# for DOS, spectrum etc. copy them to your current directory:
# cp $SNIC_TMP/* .
```

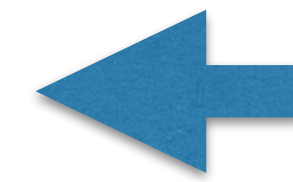
- A job script “run.sh”
- Typical example
- 1 node = 32 cores
- 1 job with 32 MPI-ranks
- Submit with:  
sbatch run.sh

# Monitoring calculations

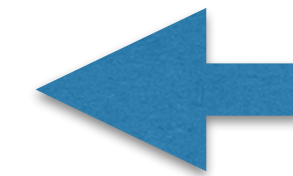
- Size of calculation     \$ grep :RKM \*.scf
- Scf convergence       \$ grep :ENE \*.scf  
                          \$ grep :DIS \*.scf
- Job run progress with timing  
    :log
- Output messages, problems  
    case.dayfile  
    slurm-1234567.out  
    \*error\*
  - Did the calculation converge?
  - Error files empty at finished run?

# :log file example

```
> (init_lapw) options:
Mon Nov 1 16:12:54 CET 2021> (x_lapw) nn -f t1
Mon Nov 1 16:12:55 CET 2021> (x) nn
Mon Nov 1 16:13:07 CET 2021> (x) sgroup
Mon Nov 1 16:13:19 CET 2021> (x) patchsymm
Mon Nov 1 16:13:24 CET 2021> (x) nn
Mon Nov 1 16:13:28 CET 2021> (x) sgroup
Mon Nov 1 16:13:41 CET 2021> (x) patchsymm
Mon Nov 1 16:13:42 CET 2021> (x) nn
Mon Nov 1 16:13:44 CET 2021> (x) sgroup
Mon Nov 1 16:13:46 CET 2021> (x) symmetry
Mon Nov 1 16:13:48 CET 2021> (x) lstart
Mon Nov 1 16:14:41 CET 2021> (x) kgen
Mon Nov 1 16:14:57 CET 2021> (x) dstart -c -p
|> (run_lapw) options: -p -i 100
Mon Nov 1 16:18:30 CET 2021> (x) lapw0 -p
Mon Nov 1 16:18:40 CET 2021> (x) lapw1 -p -c
Mon Nov 1 16:22:20 CET 2021> (x) lapw2 -p -c
Mon Nov 1 16:22:27 CET 2021> (x) sumpara -d
Mon Nov 1 16:22:28 CET 2021> (x) lcore
Mon Nov 1 16:22:28 CET 2021> (x) mixer
...
Mon Nov 1 17:03:30 CET 2021> (x) lapw0 -p
Mon Nov 1 17:03:44 CET 2021> (x) lapw1 -p -c
Mon Nov 1 17:07:13 CET 2021> (x) lapw2 -p -c
Mon Nov 1 17:07:19 CET 2021> (x) sumpara -d
Mon Nov 1 17:07:20 CET 2021> (x) lcore
Mon Nov 1 17:07:20 CET 2021> (x) mixer
Mon Nov 1 17:24:13 CET 2021> (x) kgen
Mon Nov 1 17:24:17 CET 2021> (x) kgen
Mon Nov 1 17:24:49 CET 2021> (x) lapw1 -c -p
Mon Nov 1 17:56:13 CET 2021> (x) lapw2 -c -p -qtl
```



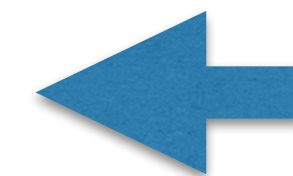
\$ init\_lapw



run\_lapw -p -i 100

scf end 

\$ x kgen

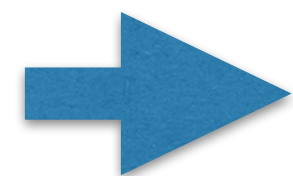


x lapw1 -c -p

x lapw2 -c -p -qtl

# Output: case.dayfile

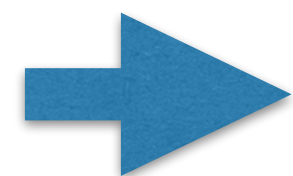
```
Calculating t1 in /proj/theophys/users/weiol/lapw/21-1/r-BN/hex/128at/t1
on n1145 with PID 56203
using WIEN2k_21.1 (Release 14/4/2021) in /software/sse/manual/WIEN2k/21.1/nsc1
|
| start      (Mon Nov  1 16:18:30 CET 2021) with lapw0 (100/99 to go)
| cycle 1    (Mon Nov  1 16:18:30 CET 2021) (100/99 to go)
|
|> lapw0 -p (16:18:30) starting parallel lapw0 at Mon Nov  1 16:18:30 CET 2021
|----- .machine0 : 8 processors
|38.374u 2.925s 0:07.12 579.9%  0+0k 808+0io 24pf+0w
|> lapw1 -p -c (16:18:40) starting parallel lapw1 at Mon Nov  1 16:18:40 CET 2021
|-> starting parallel LAPW1 jobs at Mon Nov  1 16:18:40 CET 2021
|running LAPW1 in parallel mode (using .machines)
|1 number_of_parallel_jobs
|  n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145
|n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145
|n1145(1) 6153.646u 519.944s 3:38.54 3053.7%  0+0k 288+0io 4pf+0w
| Summary of lapw1para:
|  n1145      k=1      user=6153.65      wallclock=218.54
|6154.633u 520.753s 3:40.67 3025.0%  0+0k 1128+96io 9pf+0w
|> lapw2 -p -c (16:22:20) running LAPW2 in parallel mode
|  n1145 126.920u 17.449s 0:05.09 2836.1% 0+0k 0+232io 0pf+0w
| Summary of lapw2para:
|  n1145      user=126.92      wallclock=5.09
|127.433u 17.769s 0:07.09 2047.8%  0+0k 0+336io 0pf+0w
|> lcore (16:22:28) 0.177u 0.029s 0:00.31 61.2%  0+0k 0+0io 0pf+0w
|> mixer (16:22:28) 1.039u 0.090s 0:01.26 88.8%  0+0k 0+0io 0pf+0w
|:ENERGY convergence: 0 0.0001 0
|:CHARGE convergence: 0 0.0000 0
|ec cc and fc_conv 0 1 1
|
| cycle 2    (Mon Nov  1 16:22:30 CET 2021) (99/98 to go)
```





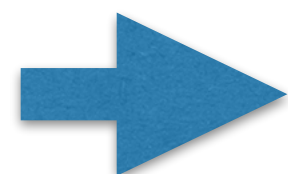
# Output: case.dayfile

```
| cycle 13 (Mon Nov 1 17:03:29 CET 2021) (88/87 to go)
|
|> lapw0 -p (17:03:30) starting parallel lapw0 at Mon Nov 1 17:03:30 CET 2021
|----- .machine0 : 8 processors
|36.368u 2.203s 0:06.04 638.4% 0+0k 0+0io 0pf+0w
|> lapw1 -p -c (17:03:36) starting parallel lapw1 at Mon Nov 1 17:03:44 CET 2021
|-> starting parallel LAPW1 jobs at Mon Nov 1 17:03:44 CET 2021
|running LAPW1 in parallel mode (using .machines)
|1 number_of_parallel_jobs
| n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145
|n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145 n1145
|n1145(1) 6085.728u 514.268s 3:26.94 3189.3% 0+0k 0+0io 0pf+0w
| Summary of lapw1para:
| n1145 k=1 user=6085.73 wallclock=206.94
|6086.803u 514.944s 3:29.50 3151.1% 0+0k 0+96io 0pf+0w
|> lapw2 -p -c (17:07:13) running LAPW2 in parallel mode
| n1145 123.664u 17.198s 0:04.50 3130.0% 0+0k 0+232io 0pf+0w
| Summary of lapw2para:
| n1145 user=123.664 wallclock=4.5
|124.181u 17.517s 0:06.39 2217.3% 0+0k 0+336io 0pf+0w
|> lcore (17:07:20) 0.161u 0.025s 0:00.24 75.0% 0+0k 0+0io 0pf+0w
|> mixer (17:07:20) 1.032u 0.092s 0:01.19 94.1% 0+0k 0+0io 0pf+0w
|:ENERGY convergence: 1 0.0001 .0000208400000000
|:CHARGE convergence: 0 0.0000 .0002632
|ec cc and fc_conv 1 1 1
|
|> stop
```



# Output: slurm-JOBID.out

```
| LAPW0 END
|[1] Done /software/sse/easybuild/prefix/software/impi/2018.3.222-
|iccifort-2018.3.222-GCC-7.3.0-2.30/bin64/mpiexec.hydra -bootstrap ssh -np 8
|-machinefile .machine0 /software/sse/manual/WIEN2k/21.1/nsc1/lapw0_mpi lapw0.def >> .time00
| LAPW1 END
|[1] + Done ( cd $PWD; $t $ttt; rm -f .lock_$lockfile[$p] )
|>> .time1_$loop
|LAPW2 - FERMI; weights written
| LAPW2 END
|[1] Done ( cd $PWD; $t $ttt $vector_split; rm -f .lock_$lockfile[$p]
|) >> .time2_$loop
| SUMPARA END
| CORE END
| MIXER END
|ec cc and fc_conv 0 1 1
|...
|in cycle 13 ETEST: .0001015650000000 CTEST: .0014922
| LAPW0 END
|[1] Done /software/sse/easybuild/prefix/software/impi/2018.3.222-
|iccifort-2018.3.222-GCC-7.3.0-2.30/bin64/mpiexec.hydra -bootstrap ssh -np 8
|-machinefile .machine0 /software/sse/manual/WIEN2k/21.1/nsc1/lapw0_mpi lapw0.def >> .time00
| LAPW1 END
|[1] + Done ( cd $PWD; $t $ttt; rm -f .lock_$lockfile[$p] )
|>> .time1_$loop
|LAPW2 - FERMI; weights written
| LAPW2 END
|[1] Done ( cd $PWD; $t $ttt $vector_split; rm -f .lock_$lockfile[$p]
|) >> .time2_$loop
| SUMPARA END
| CORE END
| MIXER END
|ec cc and fc_conv 1 1 1
|> stop
```



# Job scripts: k-pt groups with MPI

```
....  
#SBATCH -n 32  
....  
for i in $(hostlist -e $SLURM_JOB_NODELIST)  
do  
  echo "1:$i:16 " >> .machines  
  echo "1:$i:16 " >> .machines  
done  
echo granularity:1 >> .machines  
echo extrafine:1 >> .machines  
....
```

- 1 node = 32 cores
- k-pts in 2 groups with 16 MPI-ranks
- e.g. calculation with 2 k-pts

```
....  
#SBATCH -n 64  
....  
for i in $(hostlist -e $SLURM_JOB_NODELIST)  
do  
  echo "1:$i:4 " >> .machines  
  echo "1:$i:4 " >> .machines  
  echo "1:$i:4 " >> .machines  
  echo "1:$i:4 " >> .machines  
  echo "1:$i:4 " >> .machines  
  echo "1:$i:4 " >> .machines  
  echo "1:$i:4 " >> .machines  
  echo "1:$i:4 " >> .machines  
done  
echo granularity:1 >> .machines  
echo extrafine:1 >> .machines  
....
```

- 2 nodes = 64 cores
- k-pts in 8 groups with 4 MPI-ranks per node

# Job scripts: fully k-pt parallel

```
...  
#SBATCH -n 32  
...  
for i in $(hostlist -e $SLURM_JOB_NODELIST)  
do  
  for j in {1..32}  
  do  
    echo "1:$i:1 " >> .machines  
  done  
done  
echo granularity:1 >> .machines  
echo extrafine:1 >> .machines  
...
```

- 1 node = 32 cores
- 32 *independent* serial runs

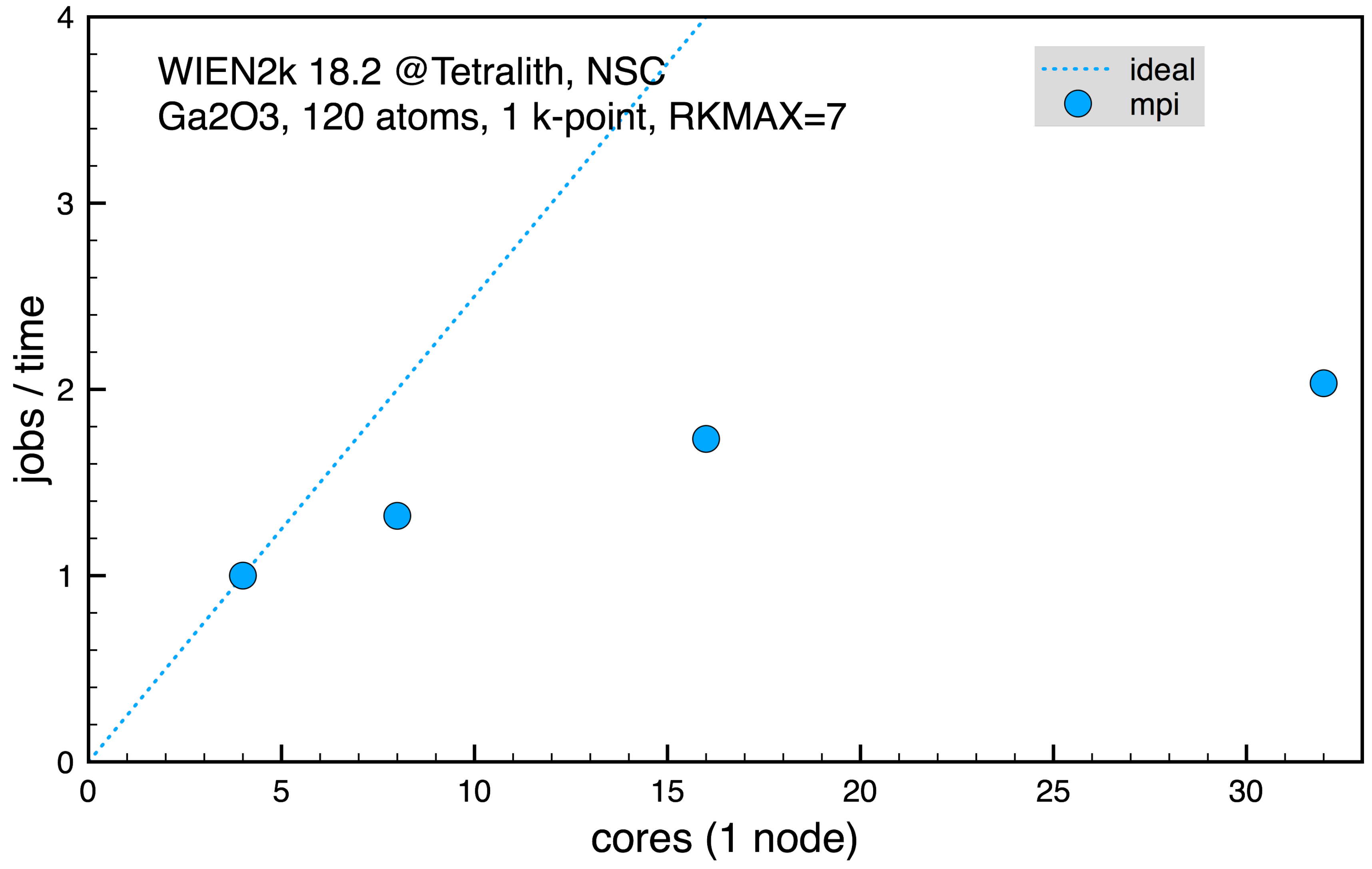
```
...  
#SBATCH -n 32  
...  
for i in $(hostlist -e $SLURM_JOB_NODELIST)  
do  
  for j in {1..16}  
  do  
    echo "1:$i:1 " >> .machines  
  done  
done  
...
```

- 1 node = 32 cores
- 16 independent serial runs (half node)
- Idea: reducing memory usage

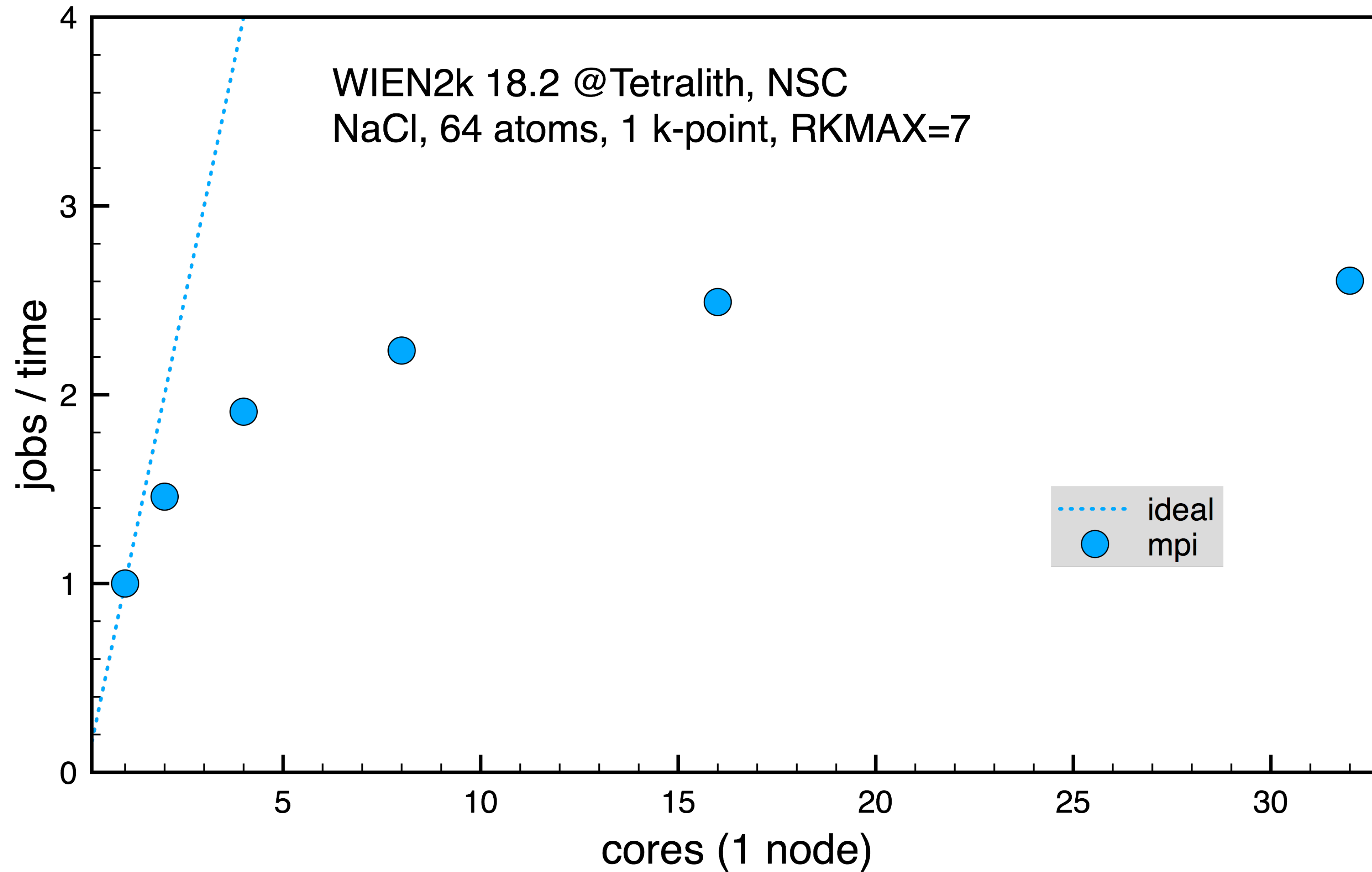
- A completely k-point parallel job, very efficient (“trivial” parallelism)



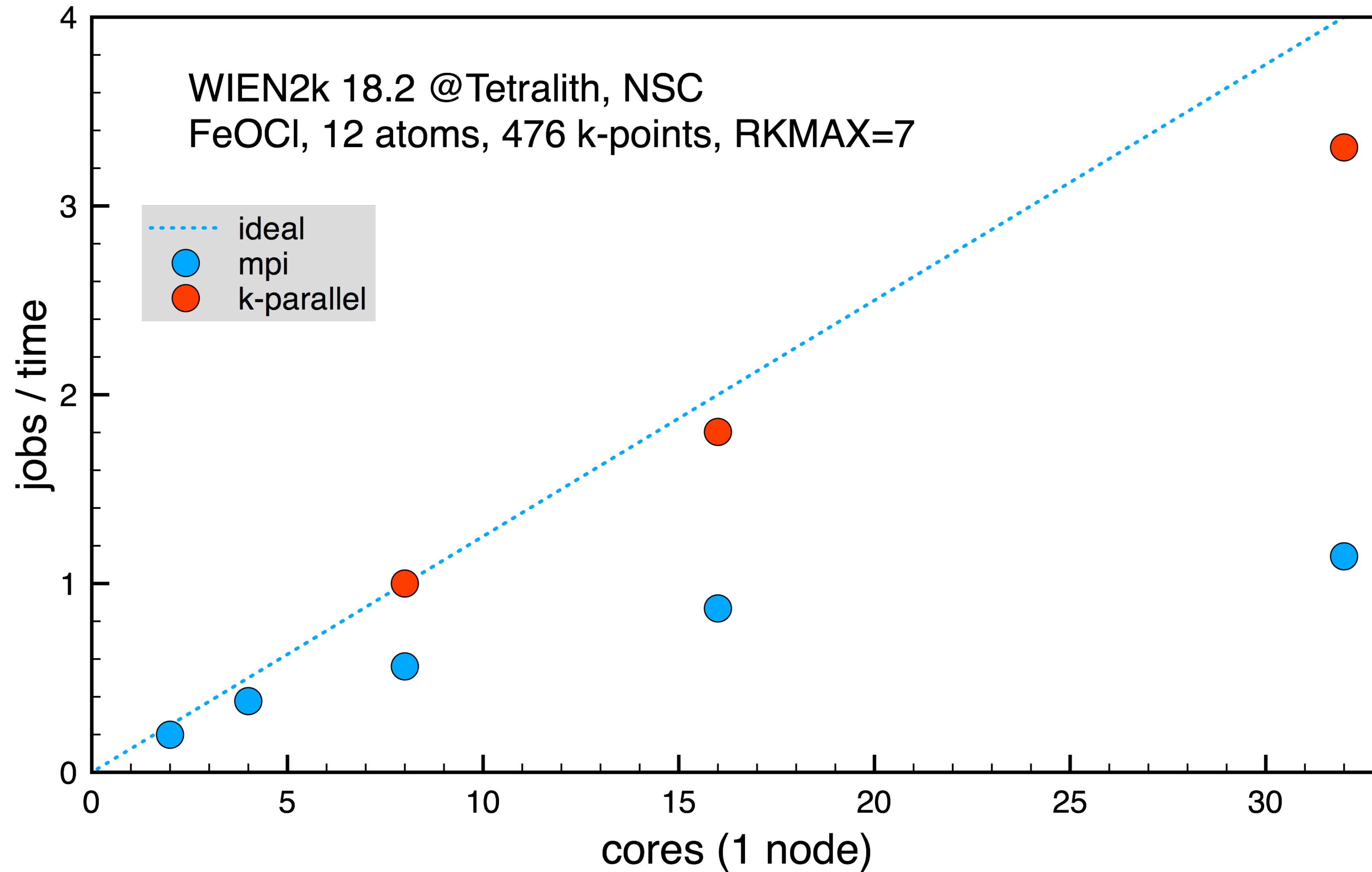
# Ga2O3 120 atoms



# NaCl 64 atoms



# FeOCl 12 atoms



# Further considerations

- Memory: allocate full node, reduce core usage
- OpenMP threading also available
- **Iterative diagonalization**, `run_lapw -p -it`
- Tool for “cleaning up” and saving a calculation



# 4. Hands-on & Questions

Weine Olovsson

National Supercomputer Centre (NSC), Linköping University

SNIC training, online mini workshop @NSC 18<sup>th</sup> Nov 2021

# Hands-on & Questions

- **Questions**, on screen demonstration
- Your own WIEN2k case?
- **Examples** for hands-on:
  - Al metal unit cell
  - NaCl perturbed lattice, 64 atoms
  - h-BN, from .cif, build supercell