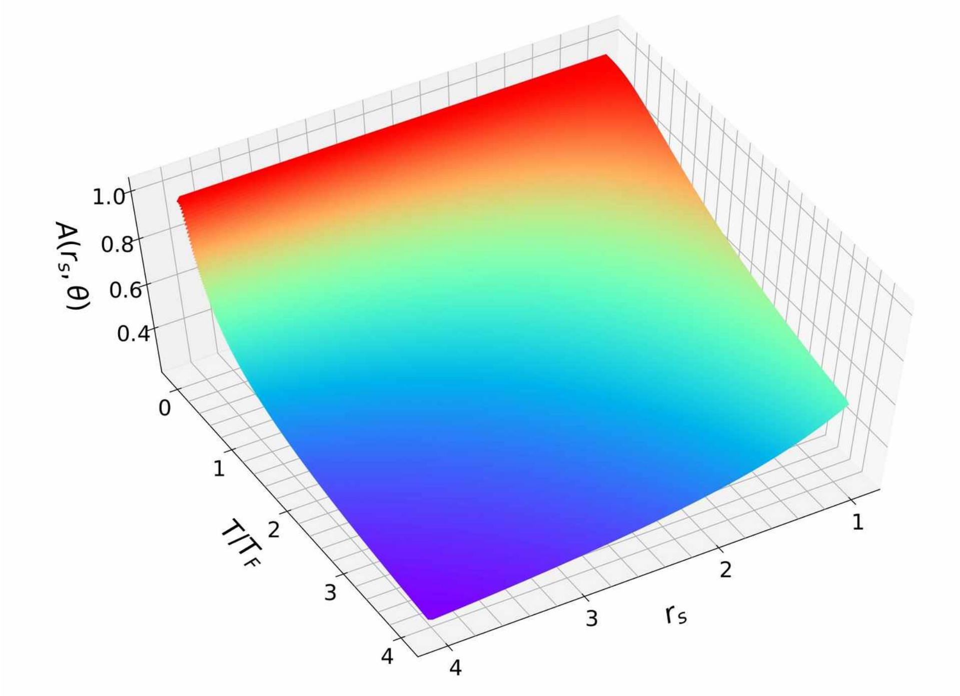
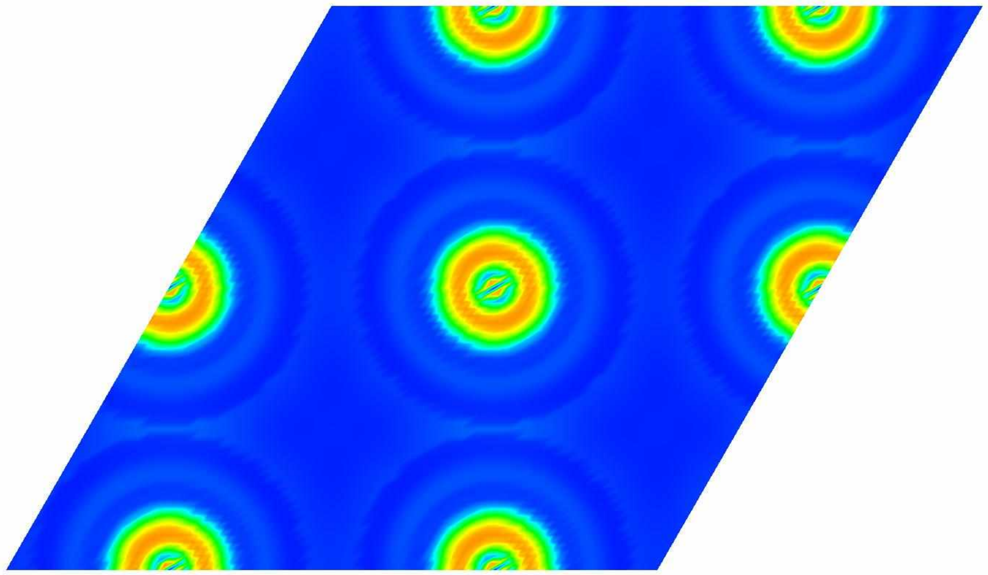
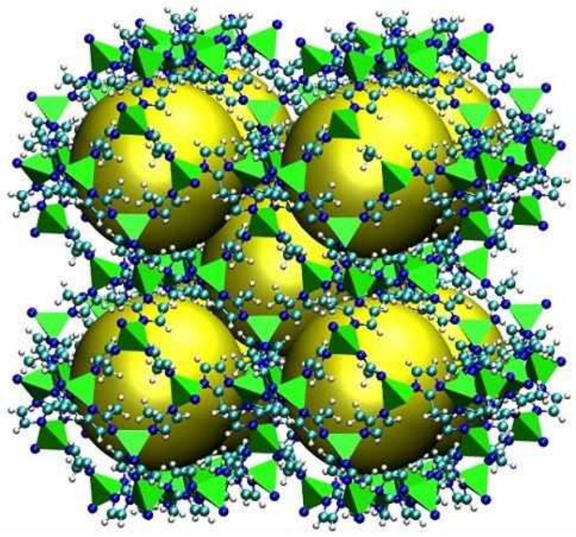




Software engineering for validating finite-temperature XC-functional

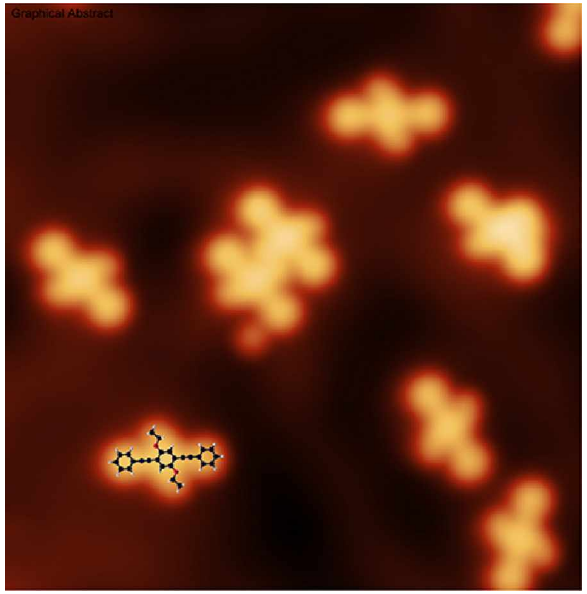
Lokamani · m.lokamani@hzdr.de · FWCC-HZDR · www.hzdr.de

XC – e**X**change **C**orrelation



Structure prediction [NIST]

Material properties [AI generated]

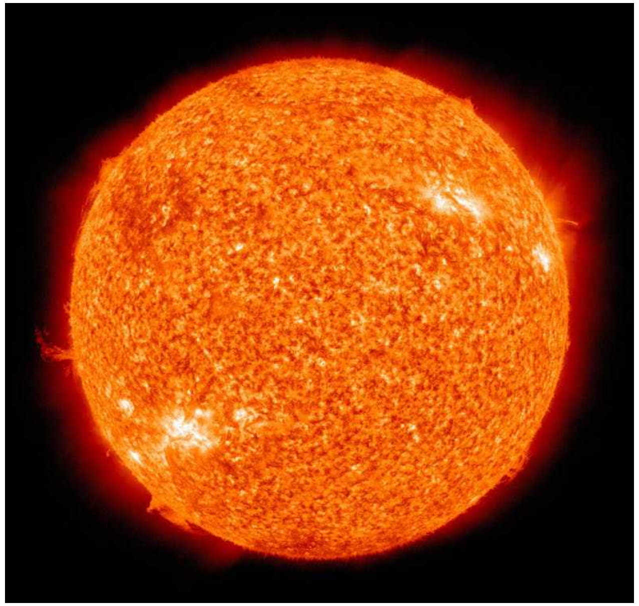


Surface science

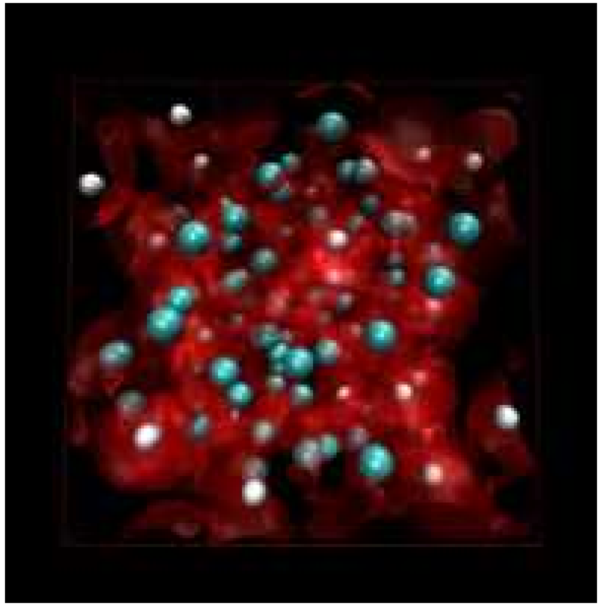
https://doi.org/10.1016/j.susc.2021.121877

- Main ingredient of **DFT** → exchange-correlation **XC**
- **DFT → D**ensity **F**unctional **T**heory at ground state (ambient conditions)
- Study properties of matter e.g. conductivity, bulk modulus
- Density as input for evaluating **XC** contributions

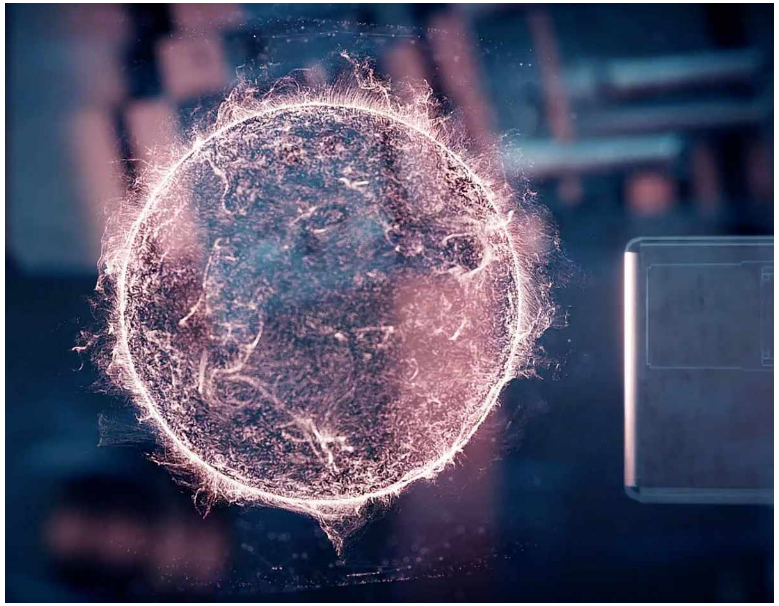
Matter under extreme conditions



Fusion in stars



Metallic hydrogen in Jupiter's core



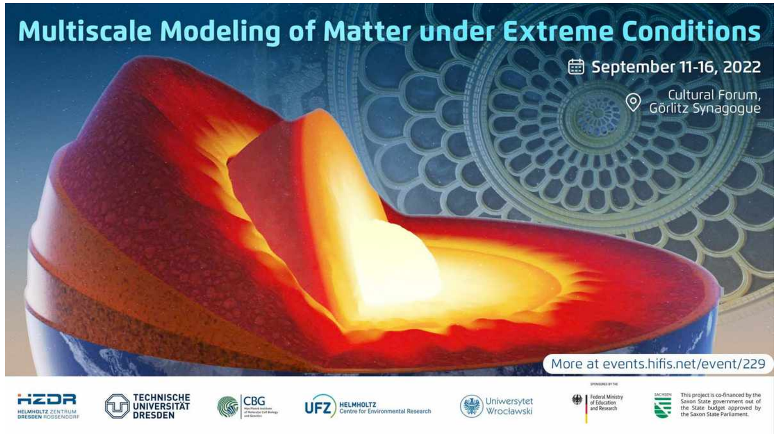
Laser-matter interaction and excited electronic states

- **DFT** can be used to study matter under extreme conditions
- But we need explicit **t**emperature dependent **XC** to improve accuracy

- Gold-standard → **P**ath **I**ntegral **M**onte **C**arlo (PIMC), very expensive and smaller systems
- Form of accurate **tXC** are known → implementation in legacy codes, not trustworthy

Steps toward an efficient implementation

Density functional theory


Multiscale Modeling of Matter under Extreme Conditions
September 11-16, 2022
Cultural Forum, Görlitz Synagogue
More at events.hifis.net/event/229
HZDR HELMHOLTZ ZENTRUM DRESDEN ROSSENDORF
TECHNISCHE UNIVERSITÄT DRESDEN
CBG
UFZ HELMHOLTZ Centre for Environmental Research
Uniwersytet Wrocławski
SPONSORED BY THE
Federal Ministry of Education and Research
SACHSEN
This project is co-financed by the Saxon State government out of the State budget approved by the Saxon State Parliament.

Initial discussion on implementing an explicit **t**emperature dependent **XC** functional **tPBE**

Hard-coded implementation of **tPBE**

Theory Department
Max Planck Institute of Microstructure Physics

The Elk Code

Hackathon at MPI in Halle

VASP LIBXC CP2K FHI-aims The ab initio materials simulation package

Challenges while reverse engineering maple definition of **XC** in LIBXC & modifying interfaces to various ab-initio codes

HIFIS

Automated validation and first step toward democratizing **t**emperature dependent **XC** for the scientific community

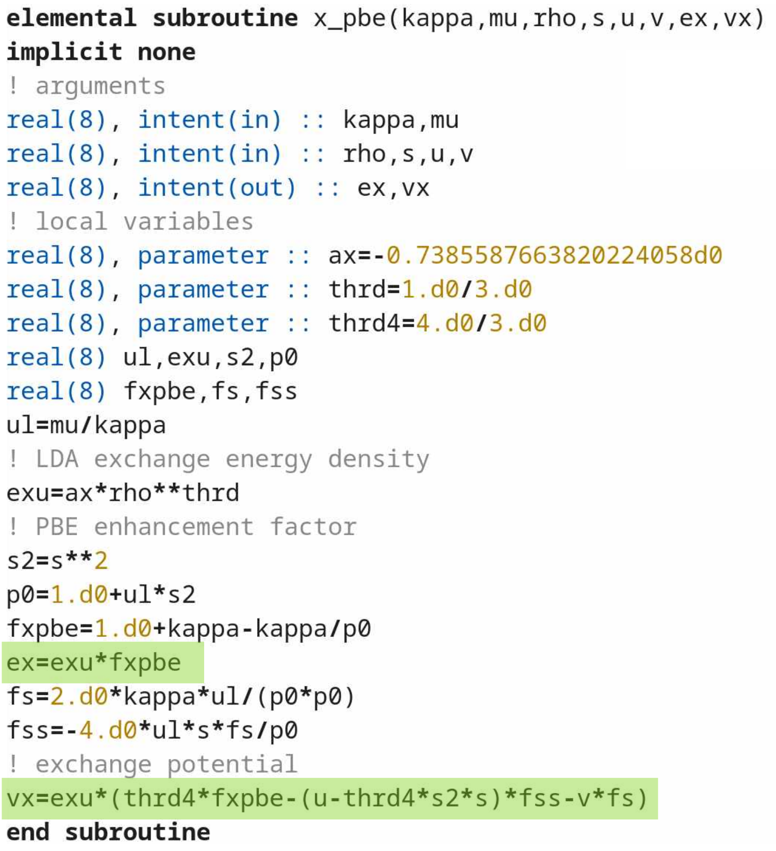
Implementing thermal PBE in elk

Advantages

- Code easily accessible/readable
- found the **X** and the derivative in **x_pbe.f90**
- found the **C** and the derivative in **c_pbe.f90**
- Quick validation

$$\theta = \frac{T}{T_F} \longleftrightarrow r_s = (\frac{3}{4\pi n})^{1/3}$$

The Elk Code



```
elemental subroutine x_pbe(kappa,mu,rho,s,u,v,ex,vx)
implicit none
! arguments
real(8), intent(in) :: kappa,mu
real(8), intent(in) :: rho,s,u,v
real(8), intent(out) :: ex,vx
! local variables
real(8), parameter :: ax=-0.7385587663820224058d0
real(8), parameter :: thrd=1.d0/3.d0
real(8), parameter :: thrd4=4.d0/3.d0
real(8) ul,exu,s2,p0
real(8) fxpbe,fs,fss
ul=mu/kappa
! LDA exchange energy density
exu=ax*rho**thrd
! PBE enhancement factor
s2=s**2
p0=1.d0+ul*s2
fxpbe=1.d0+kappa-kappa/p0
ex=exu*fxpbe
fs=2.d0*kappa*ul/(p0*p0)
fss=-4.d0*ul*s*fs/p0
! exchange potential
vx=exu*(thrd4*fxpbe-(u-thrd4*s2*s)*fss-v*fs)
end subroutine
```

Challenges

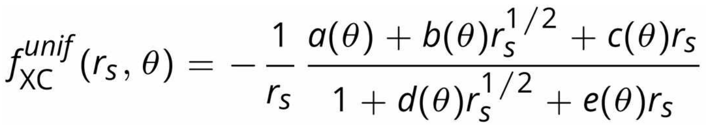
- Accurate derivative with respect to $\mathbf{r_s}$ needed
- Hidden dependencies → $\boldsymbol{\theta}$ depends on $\mathbf{T_F}$ which in turn depend on $\mathbf{r_s}$
- Cumbersome and potential source of error if higher orders of derivatives are required

Implementing thermal PBE in LIBXC

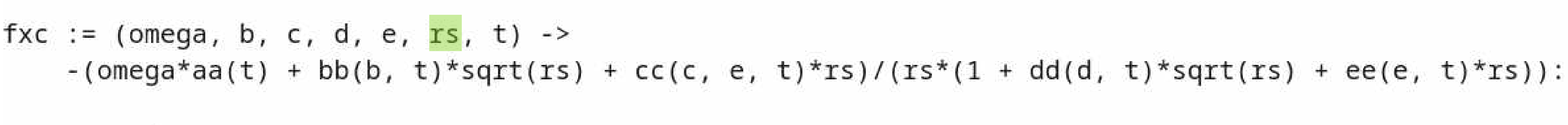
LIBXC

Advantages

- Libxc use symbolic definitions for **XC**



$$f_{\mathrm{XC}}^{unif}(r_s,\theta) = -\frac{1}{r_s}\frac{a(\theta)+b(\theta)r_s^{1/2}+c(\theta)r_s}{1+d(\theta)r_s^{1/2}+e(\theta)r_s}$$

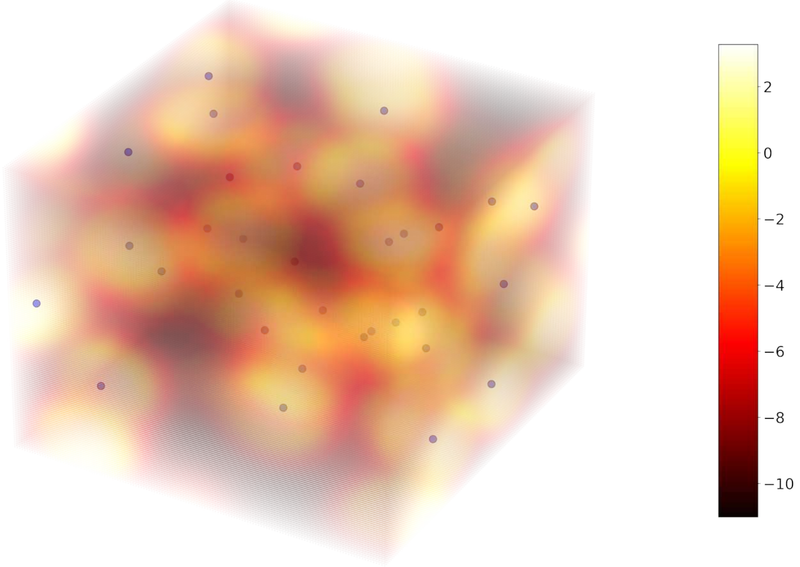


```
fxc := (omega, b, c, d, e, rs, t) ->
    -(omega*aa(t) + bb(b, t)*sqrt(rs) + cc(c, e, t)*rs)/(rs*(1 + dd(d, t)*sqrt(rs) + ee(e, t)*rs)):
```

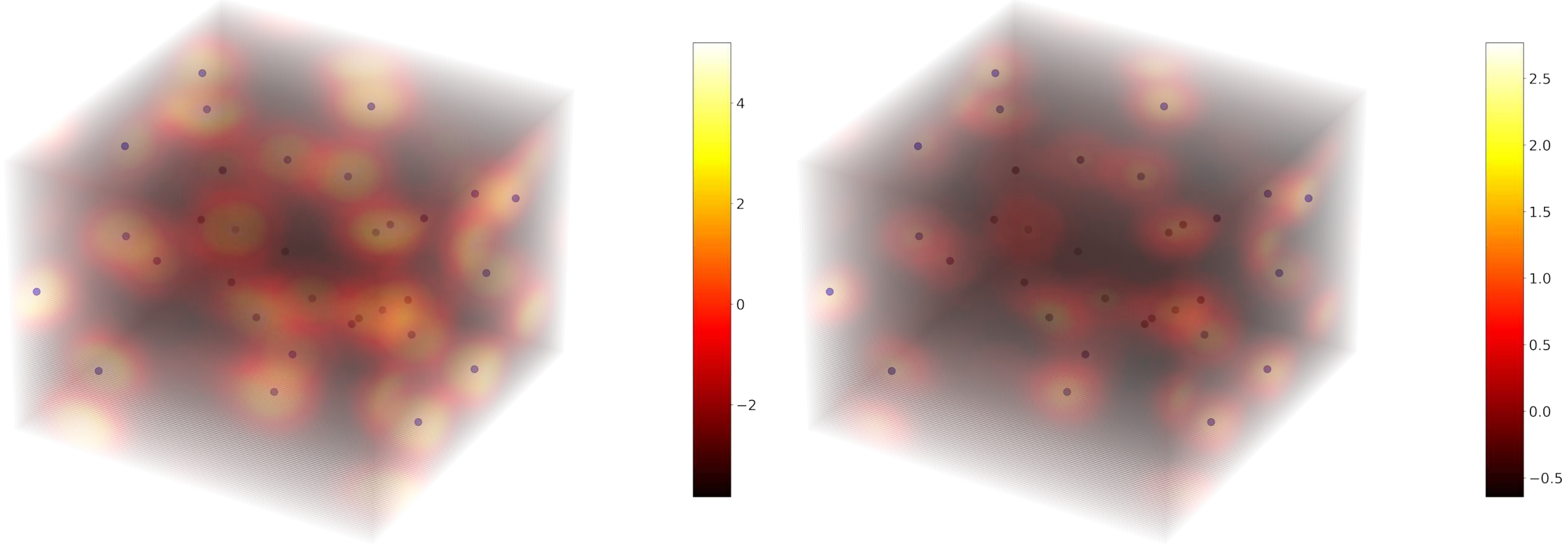
- automated evaluation of derivatives to arbitrary order
- Interface present in many ab-initio code
- First step toward democratizing temperature dependent **XC** functionals
  - no restriction to proprietory ab-initio codes or legacy codes

Results

- Visualizing thermal **XC** effects on electron density
- Relative differences between **tPBE** and **PBE**
- 32 Hydrogen atoms at density $\mathbf{r_s} = 4$



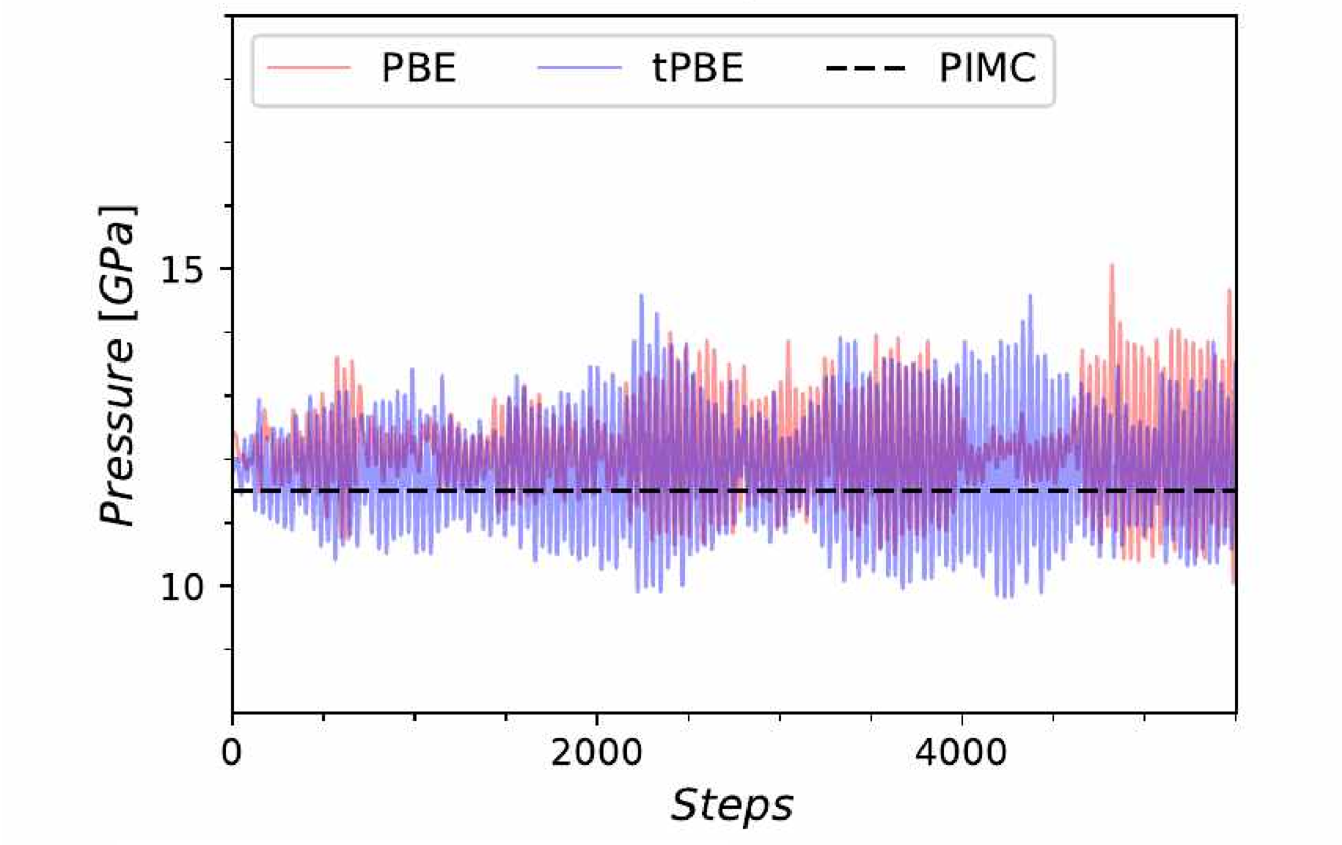
Fermi temperature with pronounced thermal **XC** effect

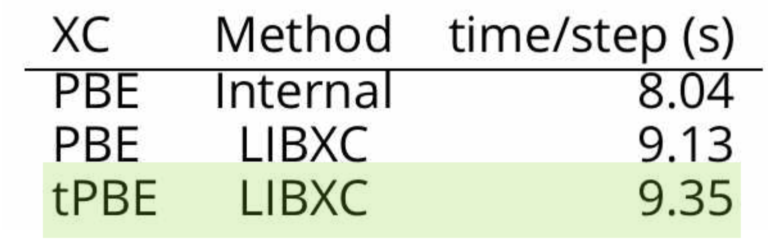


Thermal **XC** effects diminish with increasing temperature

Results

- Ab-initio quantum molecular dynamics
- Implementation more accurate than pure **XC** without temperature effects
- efficient



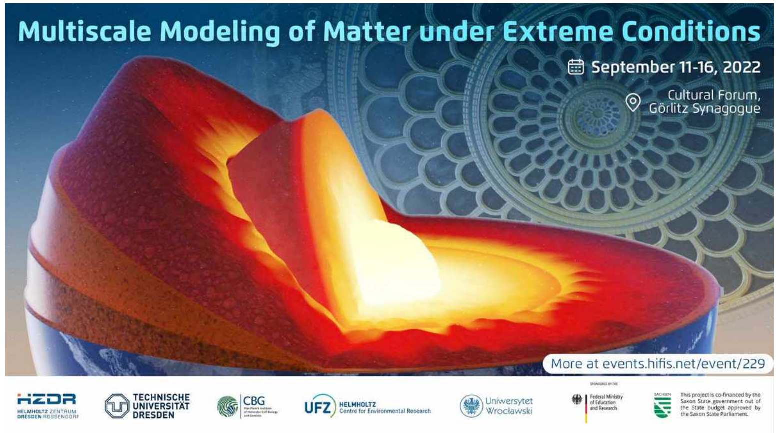


XC	Method	time/step (s)
PBE	Internal	8.04
PBE	LIBXC	9.13
tPBE	LIBXC	9.35

tPBE efficiency comparable with **PBE** in LIBXC

Steps toward an efficient implementation

Density functional theory



Multiscale Modeling of Matter under Extreme Conditions

September 11-16, 2022

Cultural Forum, Görlitz Synagogue

More at events.hifis.net/event/229

HZDR
TECHNISCHE UNIVERSITÄT DRESDEN
CBG
UFZ HELMHOLTZ Centre for Environmental Research
Uniwersytet Wrocławski

Initial discussion on implementing an explicit temperature dependent **XC** functional **tPBE**

✓

Hard-coded implementation of **tPBE**

Theory Department
Max Planck Institute of Microstructure Physics

μΦ

The Elk Code

Hackathon at MPI in Halle

✓

VASP

LIBXC

CP2K

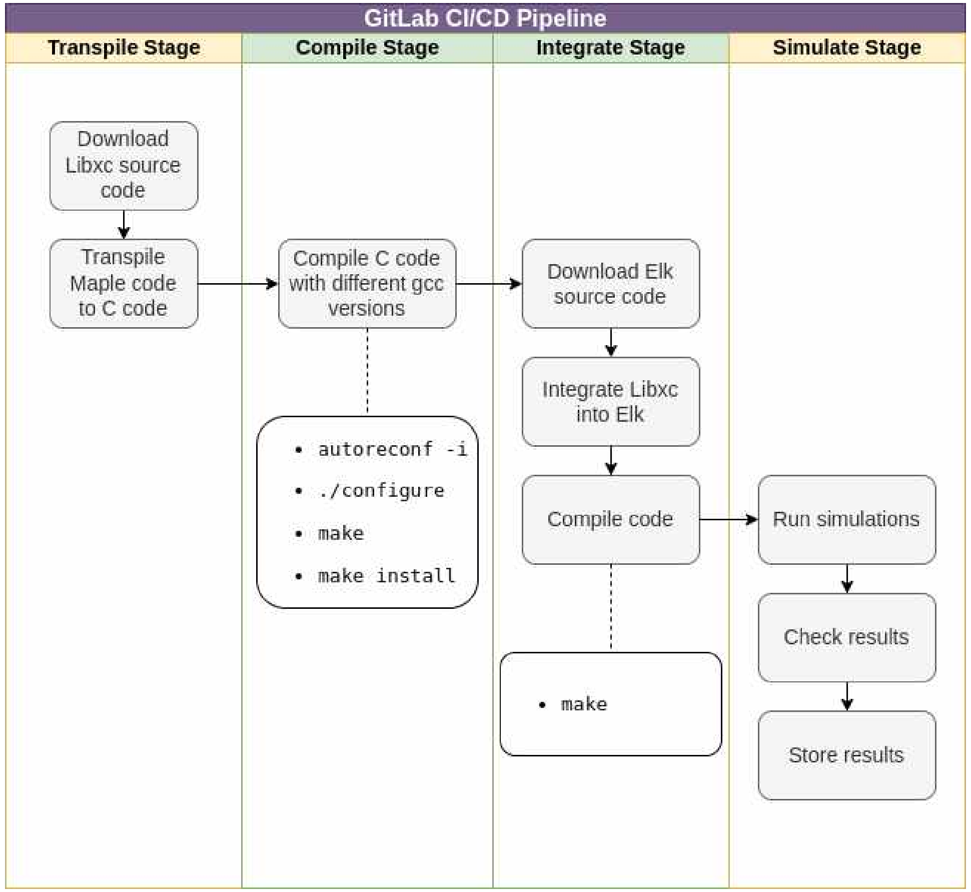
FHI-aims
The ab initio materials simulation package

Challenges while reverse engineering maple definition of **XC** in LIBXC & modifying interfaces to various ab-initio codes

HIFIS

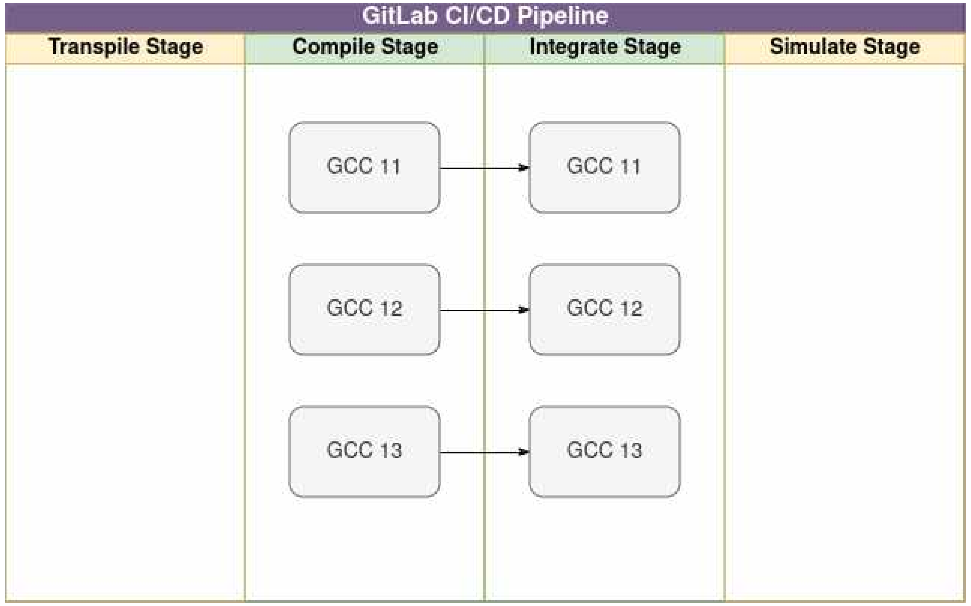
Automated validation and first step toward democratizing **t**emperature dependent **XC** for the scientific community

Implementation phase of a GitLab CI/CD Pipeline



```
• autoreconf -i
• ./configure
• make
• make install
```

```
• make
```



“Once you have clear documentation about how to set projects up, translating it into a GitLab CI pipeline is straightforward.”

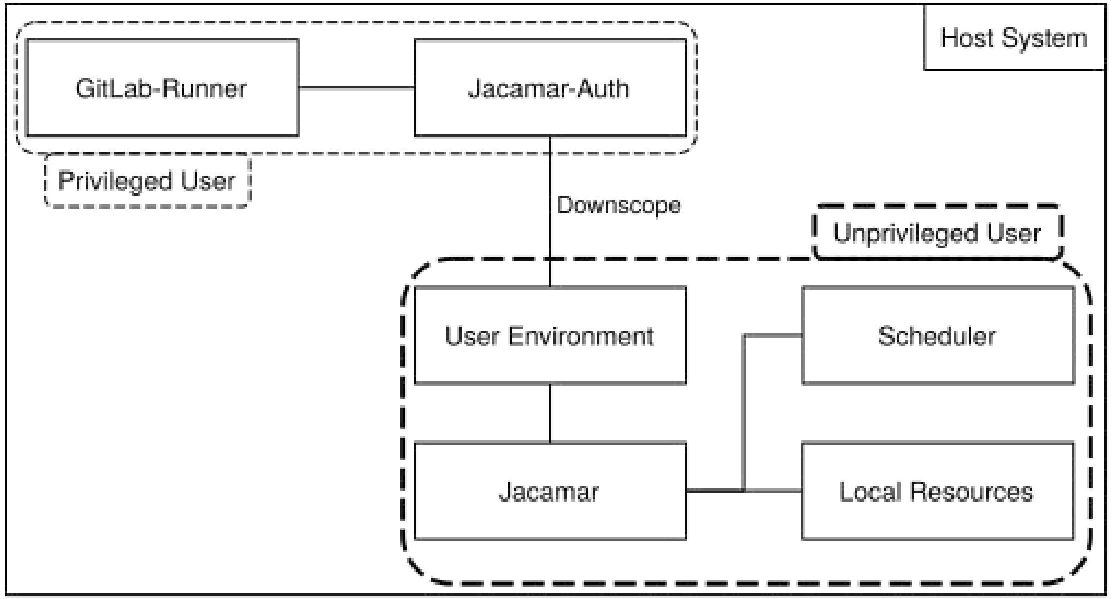
HIFIS

Utilize GitLab CI Runners on HPC systems

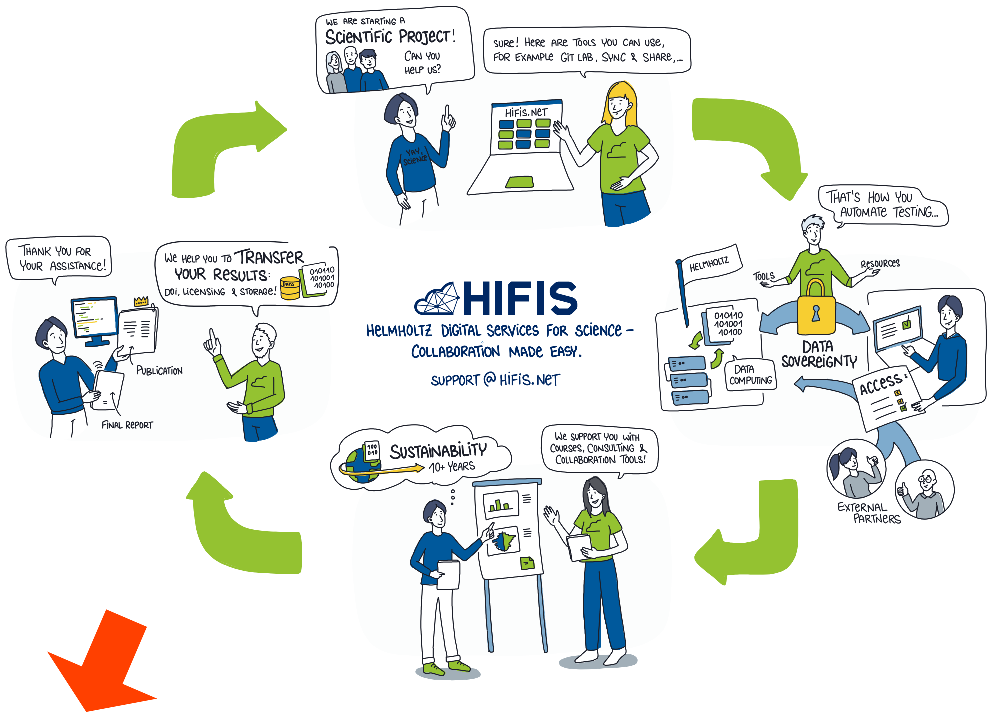
Jacamar project

- HPC focused CI/CD driver
  using GitLab's custom exector model
- GitLab Runner at HZDR
- Work-in-progress
  - Prototype implementation
  - Tackling issues regarding user management, workspace, permission and isolation

- https://codebase.helmholtz.cloud/fwcc/gitlab-hpc-driver



(see https://ecp-ci.gitlab.io/docs/admin/jacamar/introduction.html)

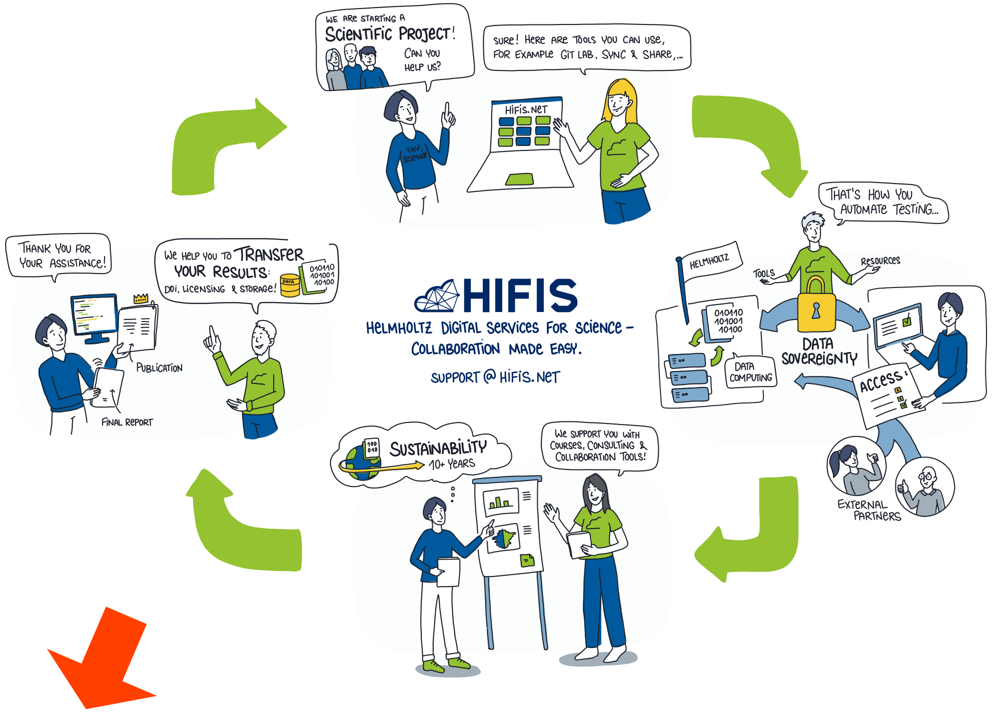

We are starting a Scientific Project! Can you help us?
Sure! Here are tools you can use, for example Git Lab, Sync & Share,...
Yay, science
HIFIS.NET
That's how you automate testing...
Helmholtz
Tools
Resources
010110 101001 10100
Data Computing
Data Sovereignty
Access:
External Partners
Sustainability 10+ Years
100 010
We support you with courses, consulting & collaboration tools!
Thank you for your assistance!
We help you to transfer your results: DOI, licensing & storage!
Data
Publication
Final Report
HIFIS
Helmholtz Digital Services for Science – Collaboration made easy.
support@hifis.net

HPC Labs
FWCC-HZDR

Collaborative community-led structure for evaluating, verifying and improving the quality of research software and code

CASUS
CENTER FOR ADVANCED SYSTEMS UNDERSTANDING

Machine Learning for Materials Design Department


We are starting a Scientific Project! Can you help us?
Sure! Here are tools you can use, for example Git Lab, Sync & Share,...
Hifis.net
That's how you automate testing...
Helmholtz
Tools
Resources
Data Sovereignty
Data Computing
Access
External Partners
Thank you for your assistance!
We help you to transfer your results: DOI, licensing & storage!
Publication
Final Report
HIFIS
Helmholtz Digital Services for Science – Collaboration made easy.
support@hifis.net
Sustainability 10+ years
We support you with courses, consulting & collaboration tools!

HPC Labs
FWCC-HZDR

Collaborative community-led structure for evaluating, verifying and improving the quality of research software and code

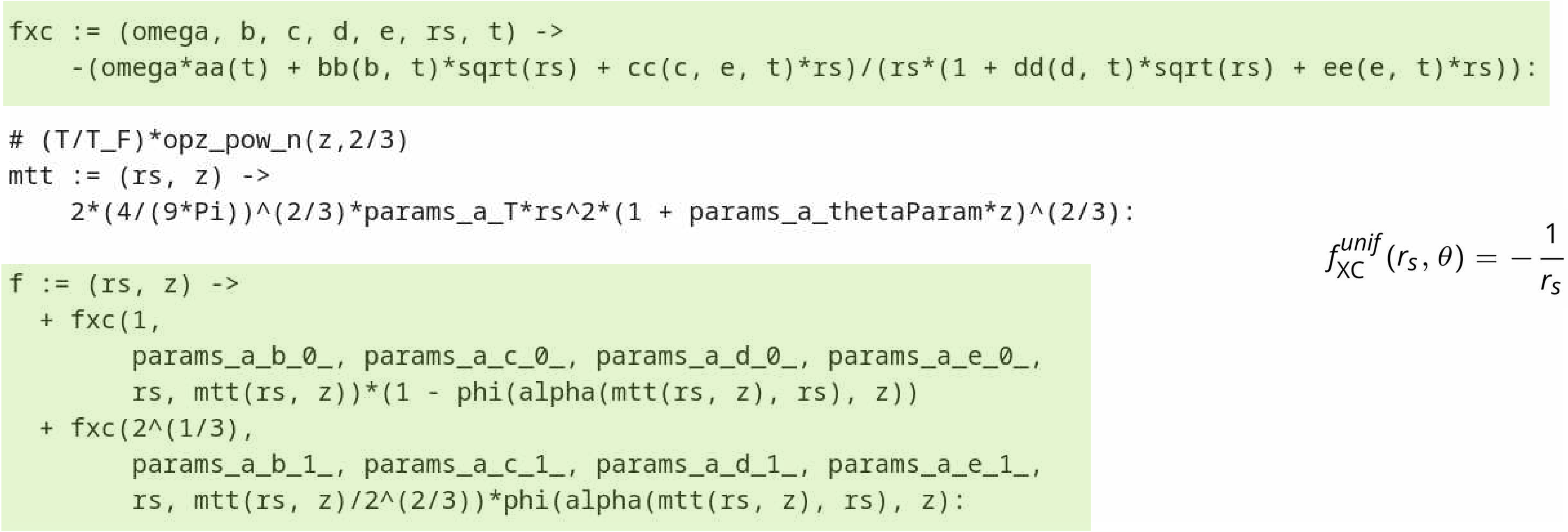
CASUS
CENTER FOR ADVANCED SYSTEMS UNDERSTANDING

Thank you for your attention

Machine Learning for Materials Design Department

Implementing thermal PBE in LIBXC

- Refactoring maple recipe for **tPBE**

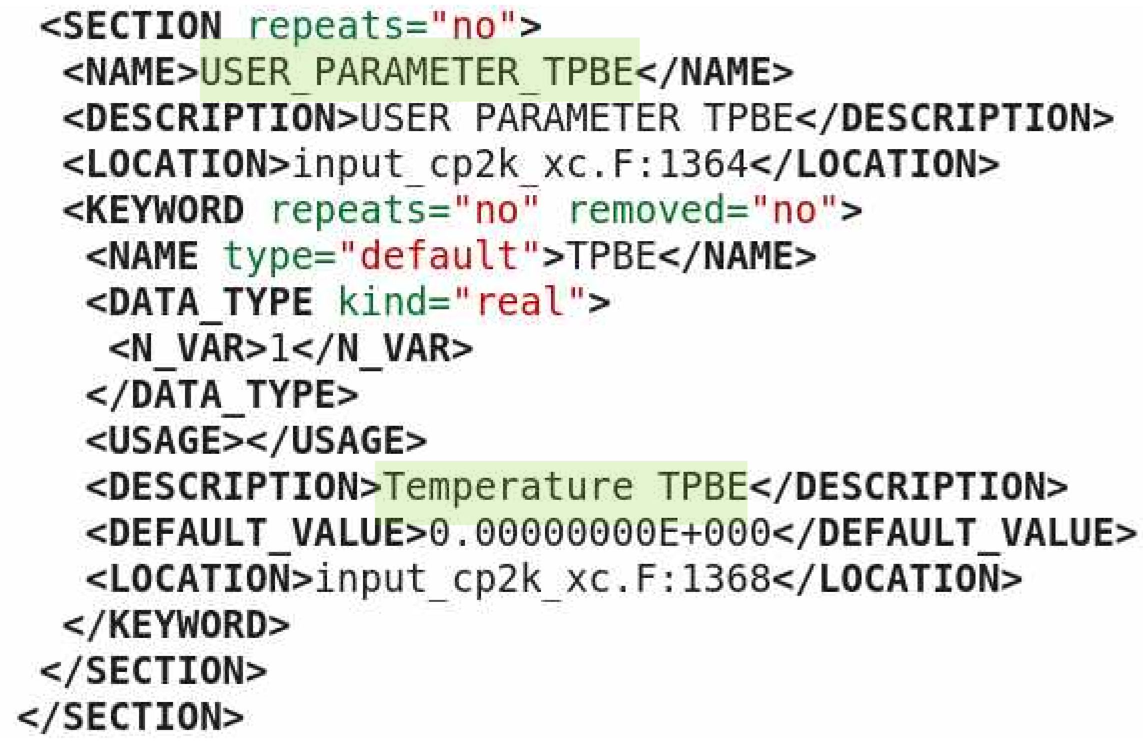


```
fxc := (omega, b, c, d, e, rs, t) ->
    -(omega*aa(t) + bb(b, t)*sqrt(rs) + cc(c, e, t)*rs)/(rs*(1 + dd(d, t)*sqrt(rs) + ee(e, t)*rs)):

# (T/T_F)*opz_pow_n(z,2/3)
mtt := (rs, z) ->
    2*(4/(9*Pi))^(2/3)*params_a_T*rs^2*(1 + params_a_thetaParam*z)^(2/3):

f := (rs, z) ->
  + fxc(1,
        params_a_b_0_, params_a_c_0_, params_a_d_0_, params_a_e_0_,
        rs, mtt(rs, z))*(1 - phi(alpha(mtt(rs, z), rs), z))
  + fxc(2^(1/3),
        params_a_b_1_, params_a_c_1_, params_a_d_1_, params_a_e_1_,
        rs, mtt(rs, z)/2^(2/3))*phi(alpha(mtt(rs, z), rs), z):
```

$$f_{XC}^{unif}(r_s, \theta) = -\frac{1}{r_s}\frac{a(\theta) + b(\theta) r_s^{1/2} + c(\theta) r_s}{1 + d(\theta) r_s^{1/2} + e(\theta) r_s}$$

Implementing thermal PBE in LIBXC

- Modfiy CP2K interface to pass the temperature parameter to LIBXC



```
 <SECTION repeats="no">
  <NAME>USER_PARAMETER_TPBE</NAME>
  <DESCRIPTION>USER PARAMETER TPBE</DESCRIPTION>
  <LOCATION>input_cp2k_xc.F:1364</LOCATION>
  <KEYWORD repeats="no" removed="no">
   <NAME type="default">TPBE</NAME>
   <DATA_TYPE kind="real">
    <N_VAR>1</N_VAR>
   </DATA_TYPE>
   <USAGE></USAGE>
   <DESCRIPTION>Temperature TPBE</DESCRIPTION>
   <DEFAULT_VALUE>0.00000000E+000</DEFAULT_VALUE>
   <LOCATION>input_cp2k_xc.F:1368</LOCATION>
  </KEYWORD>
 </SECTION>
</SECTION>
```