Combining the power of computational thermochemistry with the convenience of Python programming:

My experience with ChemAppPy

Sabrine Khadhraoui [1], Markus A. Reuter [1], Johan Zietsman [2]

[1] SMS group GmbH, Germany

[2] Ex Mente technologies, South Africa

GTT Users' Meeting (Virtual), June 2021

Content

1. Background
2. Why ChemAppPy?
3. What I gained so far
4. Conclusion

1. My background

- I am metallurgical engineer specialised in modeling, control amd optimization of metallurgical furnaces (steelmaking, non-ferrous metallurgy)
- 10 years of experience using thermochemical application
  - FactSage ™ regularly
  - SimuSage for dynamic process modeling, in a close cooperation with GTT (oxygen converter, lead smelting in the TSL process)
- limited experience with coding:
  - applications developed mainly by co-workers (programmers)
  - moderate experience using the software Matlab:
    - university days
    - mathematical process modeling + solving numerical problems (with assistance)

2. Why ChemAppPy?

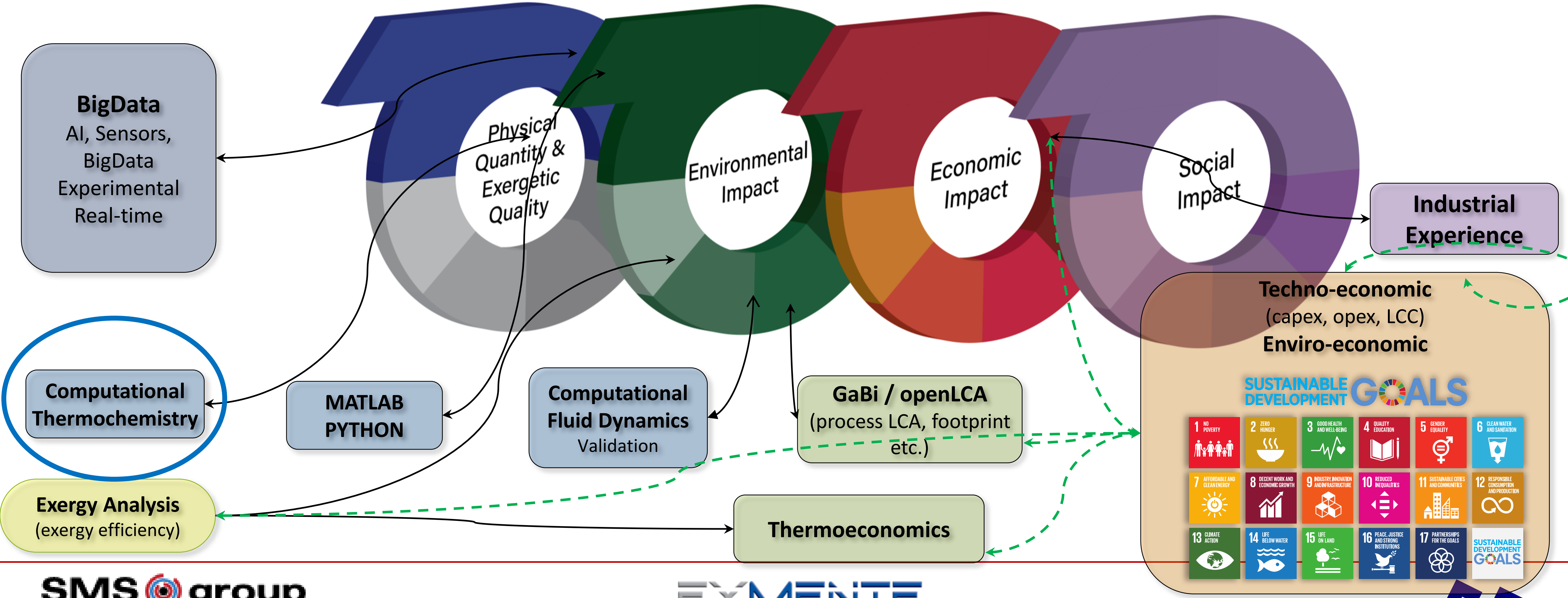
2. Why ChemAppPy?

- FactSage
  - a powerful tool that I use on a daily basis for evaluating the thermodynamic equilibrium state relevant to our processes
  - there has always been a need to use FactSage calculation options as an integrated module into a programming environment so that we can
    - automate thermochemical calculation tasks
      - inputs for the calculations
      - the post-processing of the results
    - integrate computational thermochemistry into existing digital platforms in SMS group

➡ those possibilities are provided by: ChemApp + ChemAppPy

2. Why ChemAppPy?

Digital TWINNING resource systems of SMS: Integrating our expertise, in-depth understanding of technology, and theoretical knowledge into digital platforms



BigData
AI, Sensors,
BigData
Experimental
Real-time

Physical Quantity & Exergetic Quality

Environmental Impact

Economic Impact

Social Impact

Industrial Experience

Techno-economic
(capex, opex, LCC)
Enviro-economic

Computational Thermochemistry

MATLAB PYTHON

Computational Fluid Dynamics
Validation

GaBi / openLCA
(process LCA, footprint etc.)

Exergy Analysis
(exergy efficiency)

Thermoeconomics

SUSTAINABLE DEVELOPMENT GOALS
1 NO POVERTY
2 ZERO HUNGER
3 GOOD HEALTH AND WELL-BEING
4 QUALITY EDUCATION
5 GENDER EQUALITY
6 CLEAN WATER AND SANITATION
7 AFFORDABLE AND CLEAN ENERGY
8 DECENT WORK AND ECONOMIC GROWTH
9 INDUSTRY, INNOVATION AND INFRASTRUCTURE
10 REDUCED INEQUALITIES
11 SUSTAINABLE CITIES AND COMMUNITIES
12 RESPONSIBLE CONSUMPTION AND PRODUCTION
13 CLIMATE ACTION
14 LIFE BELOW WATER
15 LIFE ON LAND
16 PEACE, JUSTICE AND STRONG INSTITUTIONS
17 PARTNERSHIPS FOR THE GOALS
SUSTAINABLE DEVELOPMENT GOALS

SMS group
SMS digital

EXMENTE
advancing through insight

2. Why ChemAppPy?

- **ChemApp** provides the flexibility that we are seeking
  - is an API (application programmer’s interface)
  - However, there was a language issue!
    - It has an interface to C and Fortran
    - For someone with a little programming experience, learning those coding languages is challenging
    - this was the case for me during learning DELPHI and C#
      - made me question the effort-benefit since I will use those programs only „occasionally“

  - Python was recommended to me as a much easier to learn coding language

2. Why ChemAppPy?

- **ChemAppPy**: became commercially available in 2019
  - made ChemApp available in a Python environment

  → we considered this as a chance to achieve our goals

  - The original functions in ChemApp are now grouped into few classes, which are easier to remember and work with:

    Info, Units, ThermochemicalSystem, EquilibriumCalculation, StreamCalculation, PhaseMapCalculation

  - Further additional functions are included in order to
    - make the calculations easier and quicker to do
    - post-processing and visualizing the calculation results.

3. What I gained so far

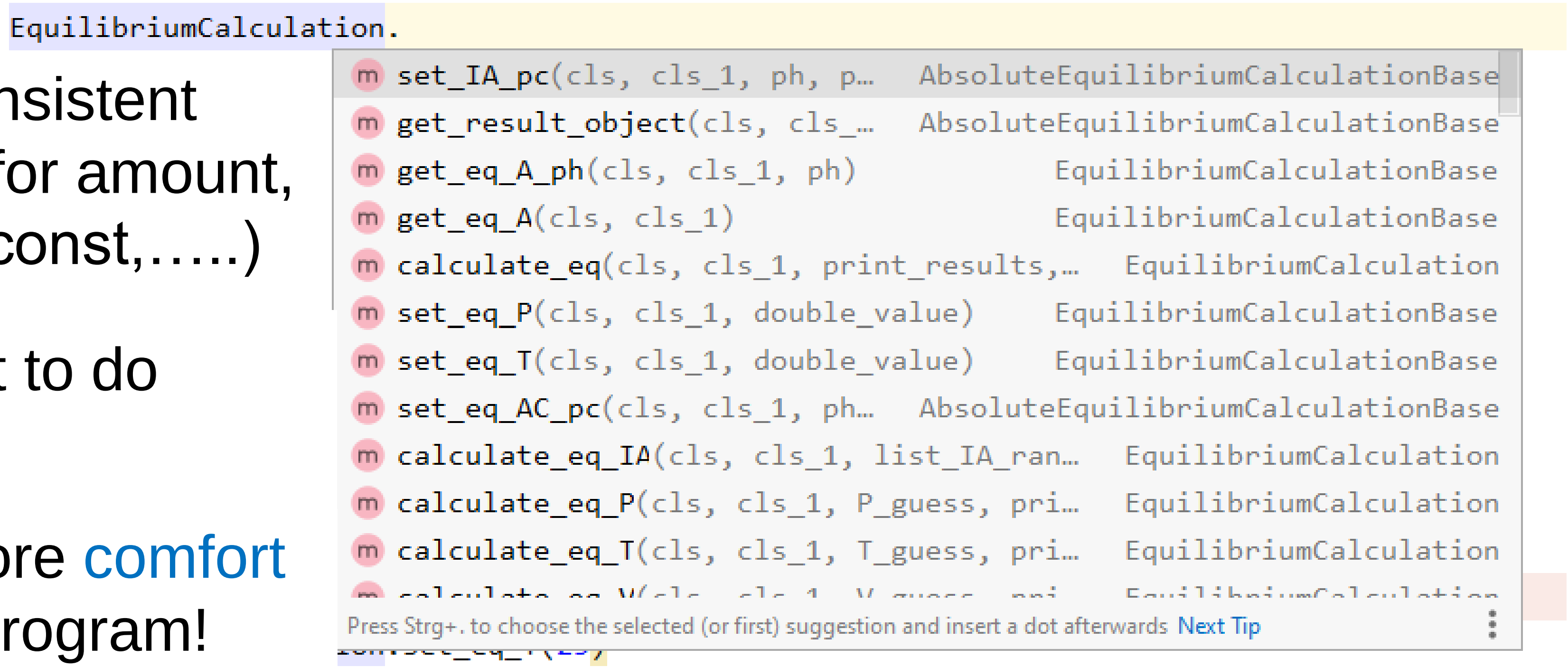
3. What I gained so far

- Programming with Python:
  - using thermochemistry in Python made me part of a large online community from which there is a lot to learn and exchange!
  - The effort to learn Python was considerably less than for other coding languages:
    - Python-Training is one part of ChemAppPy-Training: The fundamental concepts + the basic functions needed for ChemAppPy are introduced.
    - It took me about 2 weeks with an effort of 2-4 h/day to get the basics of Python and be able to work with it by myself
    - once getting over the „hurdle “ of learning how to code
      - I can now use many powerful python packages such as Pandas (data preparation and analysis) and Matplotlib (visualization)

3. What I gained so far

➢ Enjoying not only the power but also the friendliness of ChemAppPy

- The calculation functions are grouped into few Classes
- when the class is called, a list of the corresponding functions appears:



```
EquilibriumCalculation.
m set_IA_pc(cls, cls_1, ph, p…         AbsoluteEquilibriumCalculationBase
m get_result_object(cls, cls_…         AbsoluteEquilibriumCalculationBase
m get_eq_A_ph(cls, cls_1, ph)                  EquilibriumCalculationBase
m get_eq_A(cls, cls_1)                         EquilibriumCalculationBase
m calculate_eq(cls, cls_1, print_results,…         EquilibriumCalculation
m set_eq_P(cls, cls_1, double_value)           EquilibriumCalculationBase
m set_eq_T(cls, cls_1, double_value)           EquilibriumCalculationBase
m set_eq_AC_pc(cls, cls_1, ph…         AbsoluteEquilibriumCalculationBase
m calculate_eq_IA(cls, cls_1, list_IA_ran…         EquilibriumCalculation
m calculate_eq_P(cls, cls_1, P_guess, pri…         EquilibriumCalculation
m calculate_eq_T(cls, cls_1, T_guess, pri…         EquilibriumCalculation
m calculate_eq_V(cls, cls_1, V_guess, pri…         EquilibriumCalculation
Press Strg+. to choose the selected (or first) suggestion and insert a dot afterwards Next Tip
```

- The indexing adopted is consistent throughout the program (A for amount, ph for phase, pc for phase const,…..)

➡ focus more on what I want to do rather on how to do it

➡ less learning effort Vs. more comfort and fun working with the program!

Conclusion

- The need to flexibly use thermochemistry in SMS has been there for a while
- The development and availibility of ChemAppPy brought us closer to achieving our goals
  - we were now able to automate the thermochemical calculations and thus work more efficiently
  - we are heading into our goal of integrating ChemAppPy into our digital platforms in the scope a close cooperation between:
    - SMS group: R&D + Non-Ferrous Process and Technology
    - SMS digital
    - GTT and Ex Mente

Thank you for your attention!

Questions? → **please contact me under:**

Sabrine Khadhraoui
R&D Melting and Casting Technology

+49 211 / 881 – 5162
sabrine.khadhraoui@sms-group.com

SMS group GmbH

https://www.sms-group.com