

Tuto Brightway – Part 1

Téo Lavis

PhD Student in CEA & GSCOP

17/06/2024



Brightway

Program

1. Theory of BW

- A matrix approach
- Architecture
- Activities & exchanges
- Performing a LCA in BW

2. Practice

- Setup your project
- Database import
- LCA: simple, comparative, multi-category
- Data visualization (simple)
- Contribution analysis
- Uncertainties

Bright what?

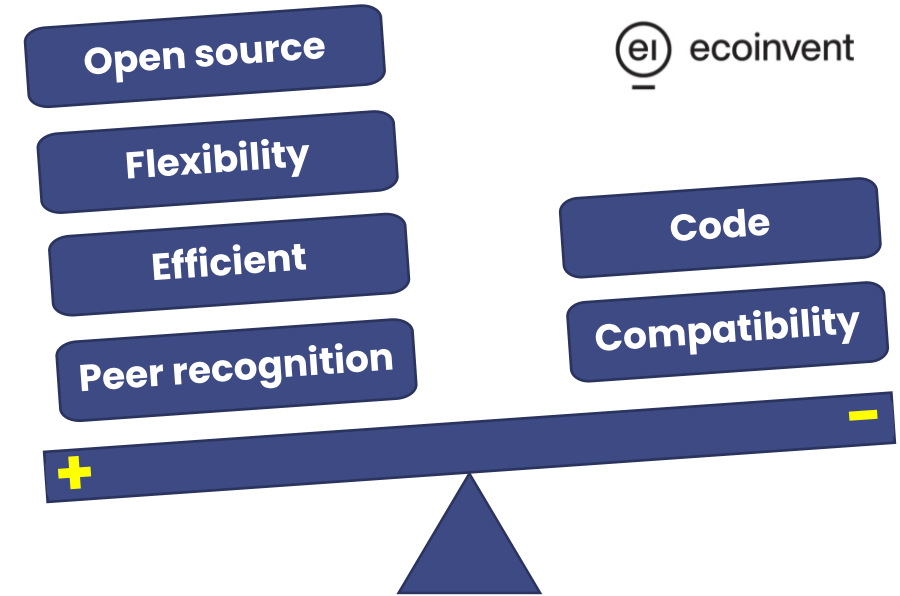
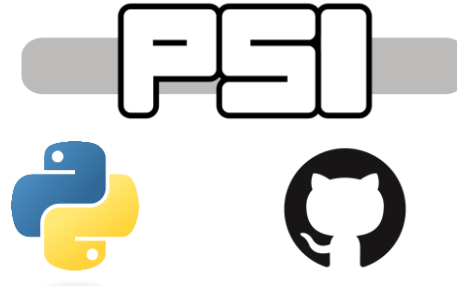
Actors



Brightway



PAUL SCHERRER INSTITUT



Main contributors:



Chris Mutel
(Ecoinvent, ex-PSI)



Tomas Navarrete
Gutiérrez (LIST)



Romain Sacchi
(PSI)



Bernhard Steubing
(Leiden)



Adrian Haas
(VSHN)

References:

- **Society** → Départ de Sentier <https://www.d-d-s.ch/>
- **Article** → Mutel, Chris. (2017) <https://doi.org/10.21105/joss.00236>
- **Documentation** → <https://docs.brightway.dev/en/latest/> et le blog de développement de Cmutel : <https://chris.mutel.org/>
- **Code** → <https://github.com/brightway-lca>
- **Tuto** (Notebook) a short selection :
 - <https://github.com/brightway-lca/brightway2/blob/master/notebooks/Getting%20Started%20with%20Brightway2.ipynb>
 - https://github.com/maximikos/Brightway2_Intro/blob/master/BW2_tutorial.ipynb
 - <https://github.com/PoutineAndRosti/Brightway-Seminar-2017/blob/master/Day%201%20AM/2%20-%20BW%20structure%20and%20first%20LCAs.ipynb>
 - If unsolvable problems → <https://stackoverflow.com/> with label 'Brightway' (very quick answer)

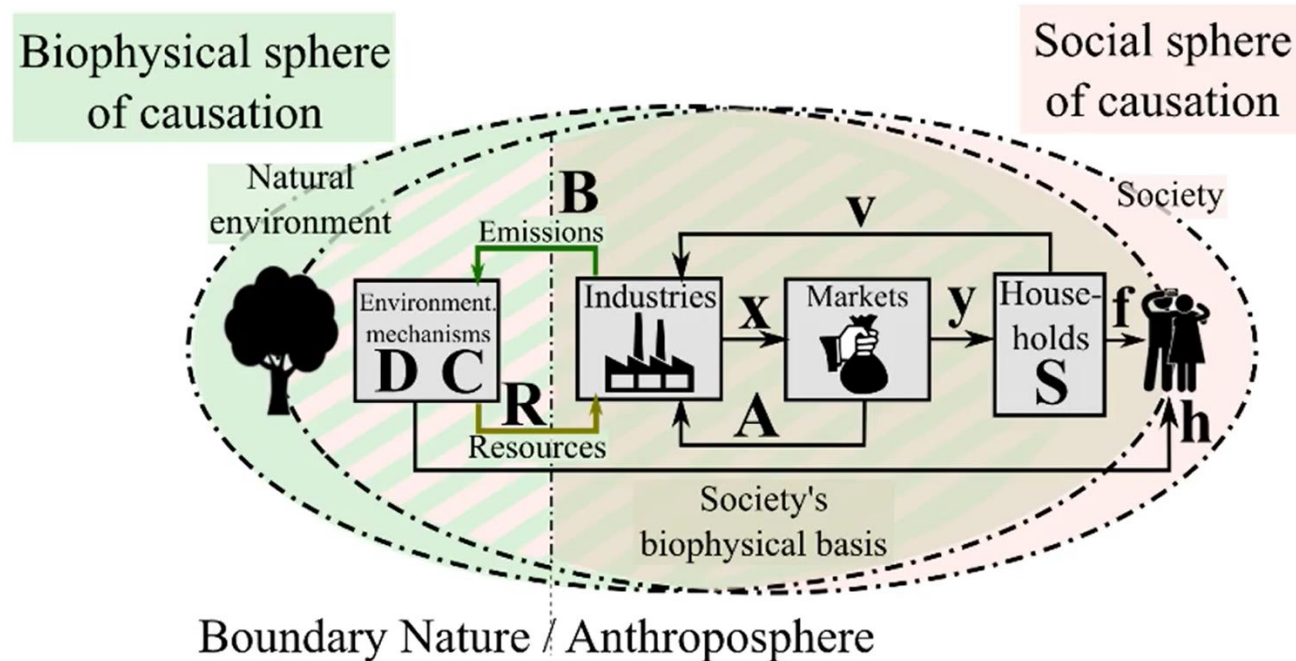


1. Theory of BW

Theory of BW

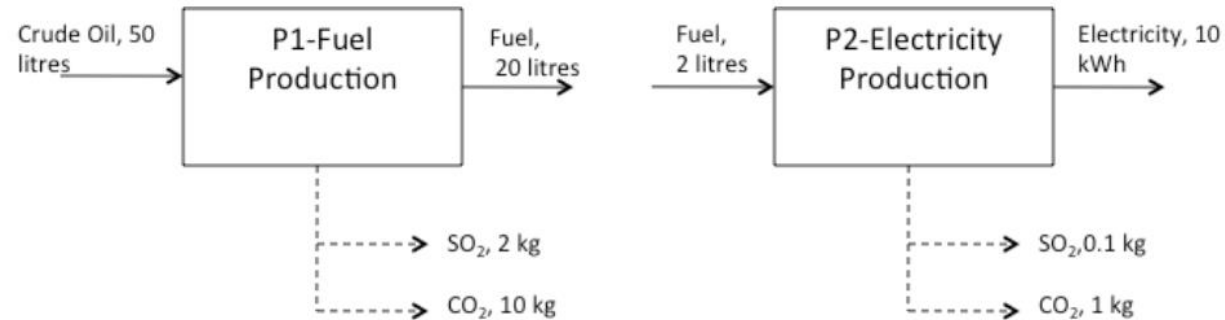
LCA in BW can be summed up by this equation (matrix vision) :

$$h = D \cdot C \cdot B \cdot (I - A)^{-1} \cdot S \cdot f \sim h = C \cdot B \cdot A^{-1} \cdot f$$



To go further: Scott Matthews et al. « LCA Book ». In Life Cycle Assessment: Quantitative Approaches for Decisions That Matter. Carnegie Mellon University, 2014.
 EcoSD, Advanced LCA Methodologies & Tools : Uncertainties & Impact Assessment, 2020
<http://www.teaching.industrialecology.uni-freiburg.de/>

Theory of BW



Let's assume $Y = \begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix}$ with Y_1 the final fuel production and Y_2 electricity production, and $X = \begin{pmatrix} X_1 \\ X_2 \end{pmatrix}$ is a scaling factor that represents the production capacity. We can write :

$$\begin{cases} 20X_1 - 2X_2 = Y_1 \\ 0X_1 + 10X_2 = Y_2 \end{cases}$$

Equation can be
rewrite with matrix :

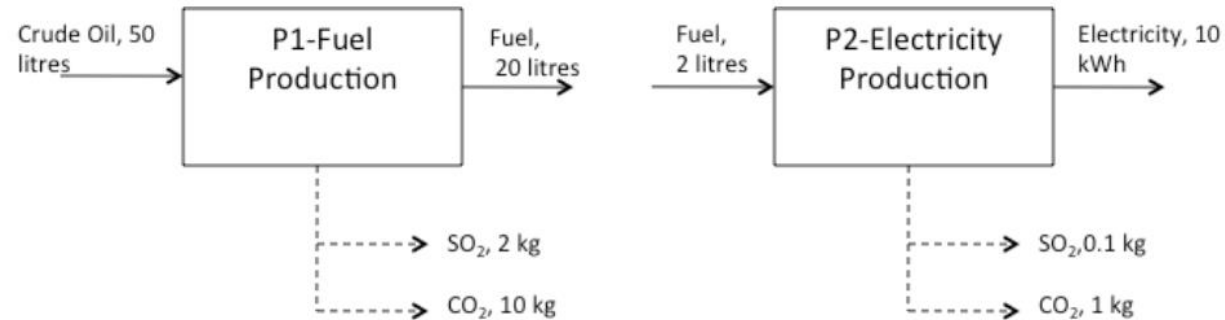
$$A \cdot X = Y \Leftrightarrow X = A^{-1} \cdot Y$$

$$A = \begin{matrix} & \begin{matrix} \text{P1} & \text{P2} \end{matrix} \\ \begin{matrix} \downarrow & \downarrow \end{matrix} & \begin{bmatrix} 20 & -2 \\ 0 & 10 \end{bmatrix} \end{matrix} \begin{matrix} \leftarrow \text{Fuel} \\ \leftarrow \text{Electricity} \end{matrix}$$

A is the **technosphere matrix**
→ Rows = flows ; Columns = Activities

$$A^{-1} = \begin{bmatrix} 0.05 & 0.01 \\ 0 & 0.1 \end{bmatrix}$$

Theory of BW



Exercise : If we want to produce 1000 kWh, how much fuel do I need ?

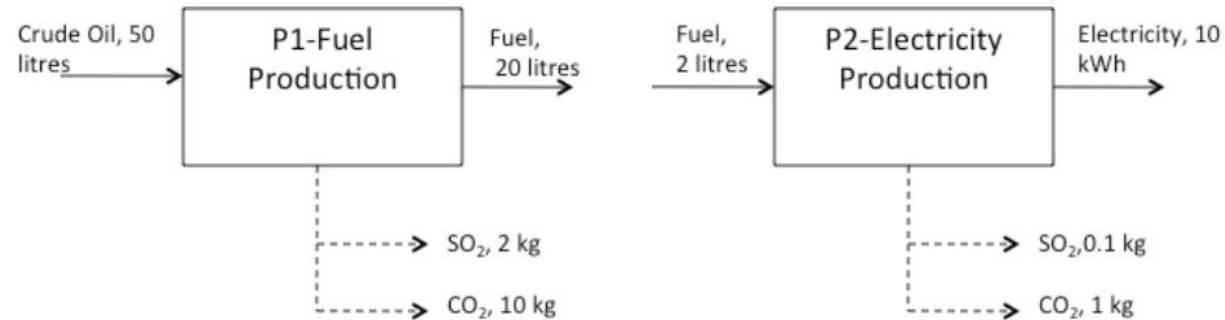
Answer:

$$\text{Produce 1000 kWh} \rightarrow Y = \begin{bmatrix} 0 \\ 1000 \end{bmatrix} \Rightarrow X = \begin{bmatrix} 10 \\ 100 \end{bmatrix}$$

$$\text{Reminder : } \begin{cases} 20X_1 - 2X_2 = Y_1 \\ 0X_1 + 10X_2 = Y_2 \end{cases} (2)$$

Which means I need to do **10× the P1-process** to do **100× the P2-process** (to get 1000 kWh),

Theory of BW



If we add the **biosphere matrix B**, which represents the emissions for each process, we get **b**, **the life cycle inventory** (i.e. all elementary flows to meet the demand f)

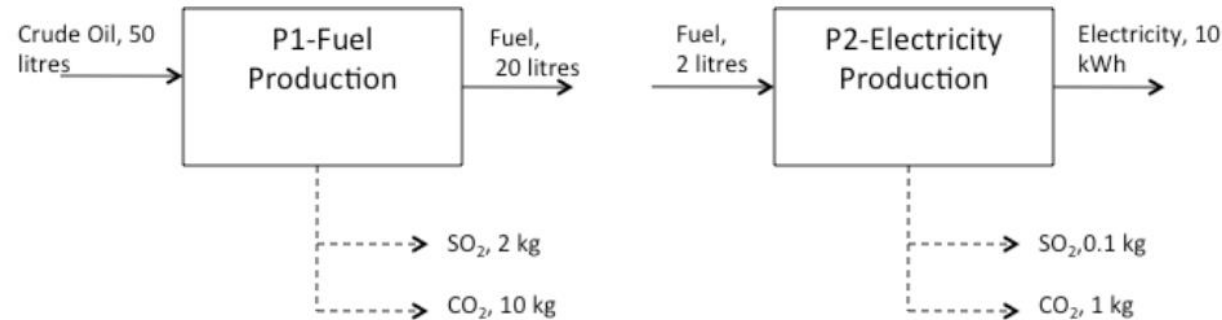
$$\mathbf{b} = \mathbf{B} \cdot \mathbf{X} = \mathbf{B} \cdot \mathbf{A}^{-1} \cdot \mathbf{Y} \quad (4)$$

$$\text{with : } \mathbf{B} = \begin{bmatrix} -50 & 0 \\ 2 & 0.1 \\ 10 & 1 \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} 20 & -2 \\ 0 & 10 \end{bmatrix}$$

columns → **activities** (ie. the process)
rows → **intermediary flows** (for A) or **elementary flows** (for B).

Theory of BW



Exercise : What are the emissions that result from the production of 1000 kWh ?

Reminder :

$$B = \begin{bmatrix} -50 & 0 \\ 2 & 0.1 \\ 10 & 1 \end{bmatrix}$$

Answer:

$$\text{Produce 1000 kWh} \rightarrow Y = \begin{bmatrix} 0 \\ 1000 \end{bmatrix} \Rightarrow X = \begin{bmatrix} 10 \\ 100 \end{bmatrix} \Rightarrow b = B \cdot X = B \cdot A^{-1} \cdot Y$$

Numerical application : $b = \begin{bmatrix} -500 \\ 30 \\ 200 \end{bmatrix}$ which means we will extract 500L of crude oil and emits 30kg SO₂ and 200kg CO₂

Theory of BW

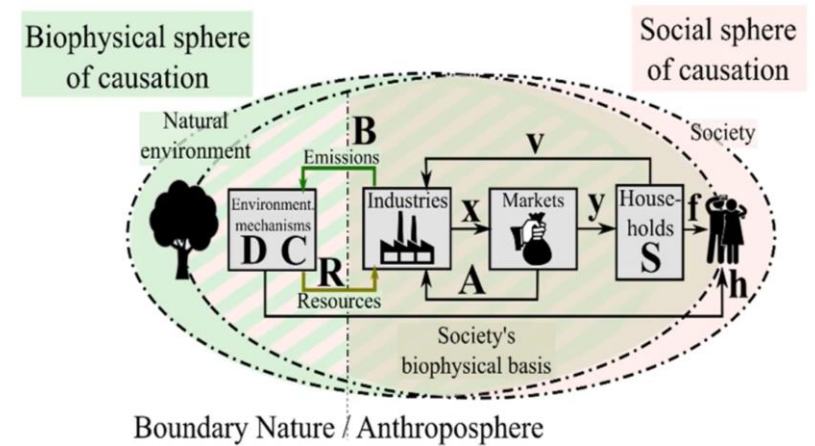
Go back to our main equation :

s = supply array, what is needed to supply my functional unit

$$h = C \cdot B \cdot A^{-1} \cdot f$$

b = **life cycle inventory**, all the elementary flows to meet my demand

h represents the characterized impacts for one impact category



Nomenclature:

f = final demand vector $[p \times 1]$

A = **technosphere matrix** $[p \times q]$

B = **biosphere matrix** $[r \times q]$

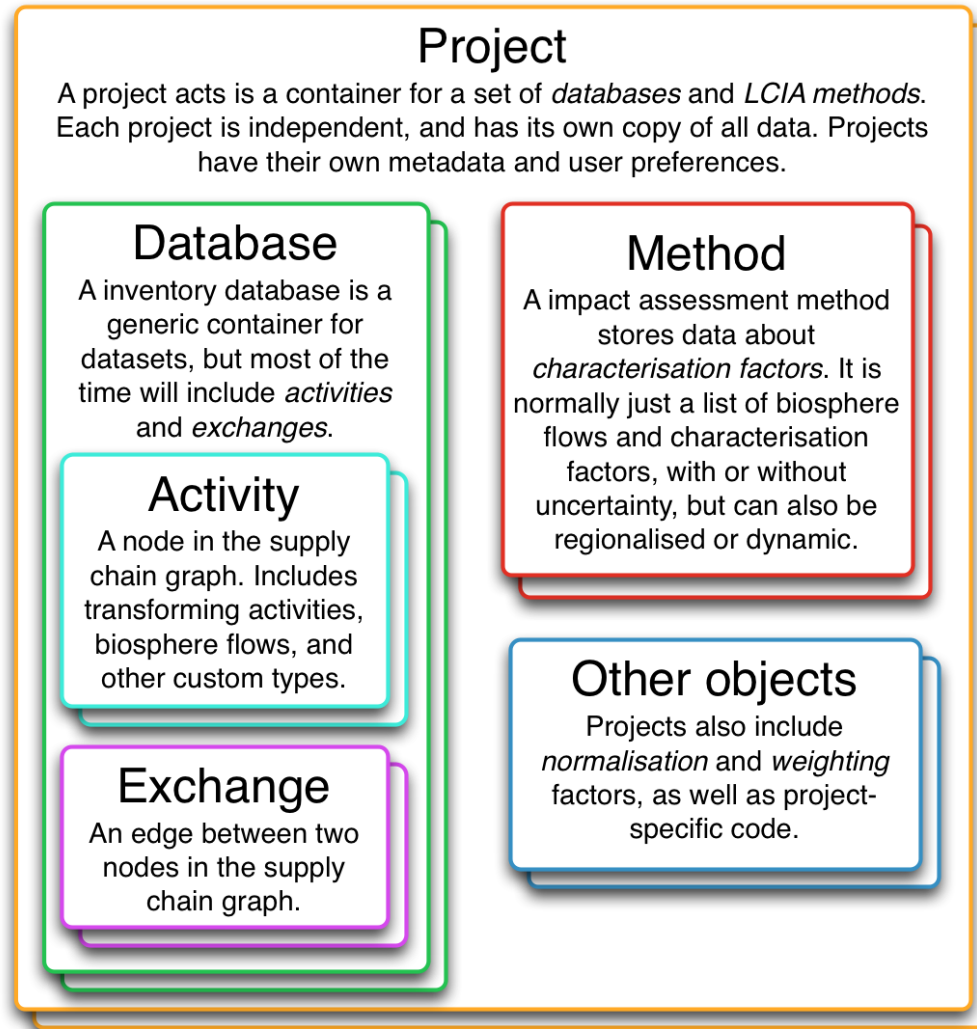
C = **characterization matrix** $[p \times q]$

h = characterised inventory matrix $[1]$

columns → **activities/process** $[q]$

rows → **intermediary flows** for A $[p]$ or **elementary flows** for B $[r]$

Architecture



- Data in Brightway2 is structured in a hierarchy.
- At the top level, we have **projects**.
 - contains **database** and **LCIA methods**
 - totally independent of other projects
 - saved as subdirectories in the file system
- The database contain the activities connected with exchanges

NB : BW and Activity Browser share the same projects/databases

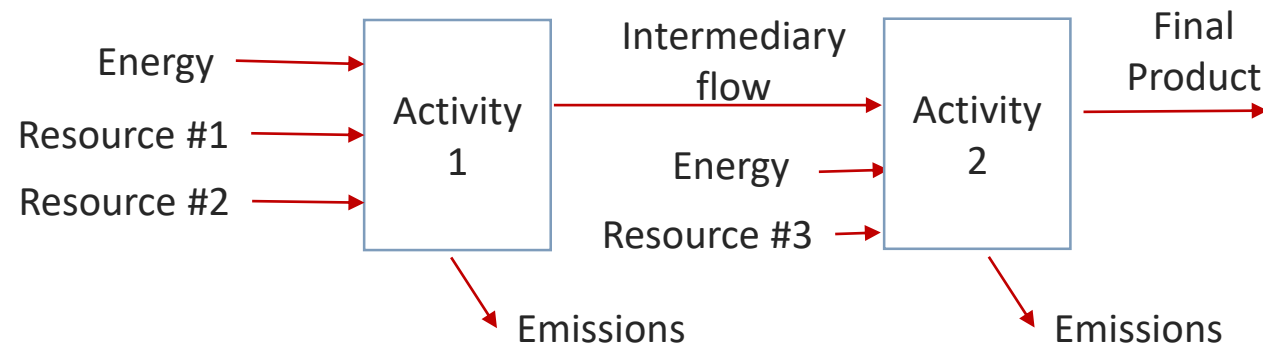
Activities & exchanges

Activities = **nodes** in the database

- include transforming and **market/product activities** (technosphere or intermediary flows)
- but also **biosphere flows** (biosphere flows)

Exchanges = **edges between two nodes** → inputs or outputs of several types :

- **biosphere** : consumption of resource or emissions,
- **technosphere** process,
- **production** : output amount



Linking activity datasets within and between databases requires a **unique identifier or a code** = a string of numbers and letters

```
In [7]: activity[0], activity[1]
Out[7]: ('ecoinvent 3.2 cutoff', 'ab2f7a551a06a59de9191065128233e4')

In [8]: activity == ('ecoinvent 3.2 cutoff', 'ab2f7a551a06a59de9191065128233e4')
Out[8]: True
```

LCA in BW

1. Translate the functional unit into a demand array
2. Find the right parameter arrays, and ask matrix builder for matrices
3. Solve the linear system $As=F$
4. Multiply the result by the LCIA CFs, if a LCIA method is present

FU → {activity : amount}

Select a characterization method

Produce A and B matrices and solves $As=f$

Compute the impacts .lci

Analysis : contribution, MC, sensitivity...

```
[11]: NMC111_pack=[NMC111_pack for NMC111_pack in eidb if
                'market for battery, Li-ion, NMC111, rechargeable, prismatic' in str(NMC111_pack)][0]
fu_NMC={NMC111_pack.key:1}
fu_NMC
```

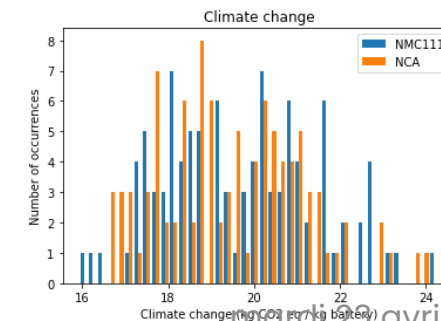
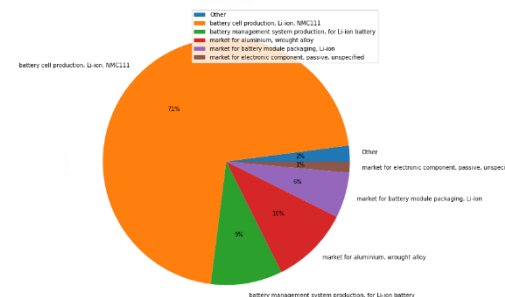
```
[11]: (('ecoinvent 3.8_cutoff_ecoSpold02', 'bb7113838446330b7ffd1d91f72a1201'): 1)
```

```
[14]: list_methods = [method for method in bw.methods if "CML v4.8 2016"
                    in str(method) and not "no LT" in str(method)]
list_methods[1]
```

```
[14]: ('CML v4.8 2016', 'climate change', 'GWP 100a')
```

```
[16]: lca = bw.LCA(fu_NMC, list_methods[1])
lca.lci()
lca.lcia()
print("La fabrication d'1kg de pack NMC111 émet {:f}{:}"
      .format(lca.score, bw.Method(list_methods[1]).metadata['unit']))
```

La fabrication d'1kg de pack NMC111 émet 16.739185kg CO2-Eq





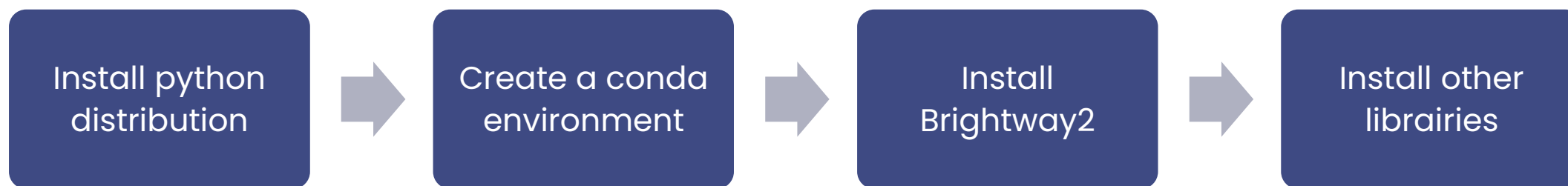
2. Practice

Installation

All resources and documentation can be found in open-source on Github and official website:

- Collaborative tool whose evolution and improvement are open to all
- Installation guide
- Tutorial
- Practical examples
- Open-access code

Procédure d'installation un peu difficile, dans les grandes lignes :



- Procedure detailed on <https://docs.brightway.dev/en/latest/content/installation/index.html>
- A simplified installation can be set up with the yaml file tuto-env.yml which create a dedicated environment tuto and install libraries all at once, in a compatible and stable version (see README.md for the detailed instructions)

Partie pratique

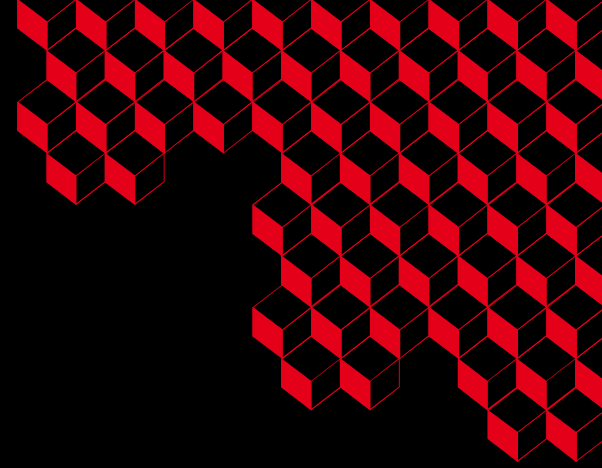


Setup :

- ✓ Create a working directory on your computer (`/Documents/tuto_bw/` for instance)
- ✓ Clone the repository or copy/Paste `BW2_tutorial_perso_ecoinvent.ipynb` in your working directory

Open Jupyter :

- ✓ Open a anaconda/miniconde console
- ✓ Run `conda activate bw2` (replace bw2 with the envt name where bw is installed)
- ✓ Open a Jupyter interface: run `jupyter lab`
- ✓ A web (localhost) window open with Jupyter interface
- ✓ Dig into your folder to reach your working directory
- ✓ Open up the Notebook `BW2_tutorial_perso_ecoinvent.ipynb`
- ✓ Best practice: when over, go in `File/Shut down` to properly turn off the server. (or you can also `Ctrl + C` in the console to force it to stop)



Thank you for your attention



Teo Lavisse

3rd year PhD Student

CEA Liten & GSCOP

teo.lavisse@cea.fr