

# Final presentation

Internship results of Maximilian Hanauske or  
how to use the code

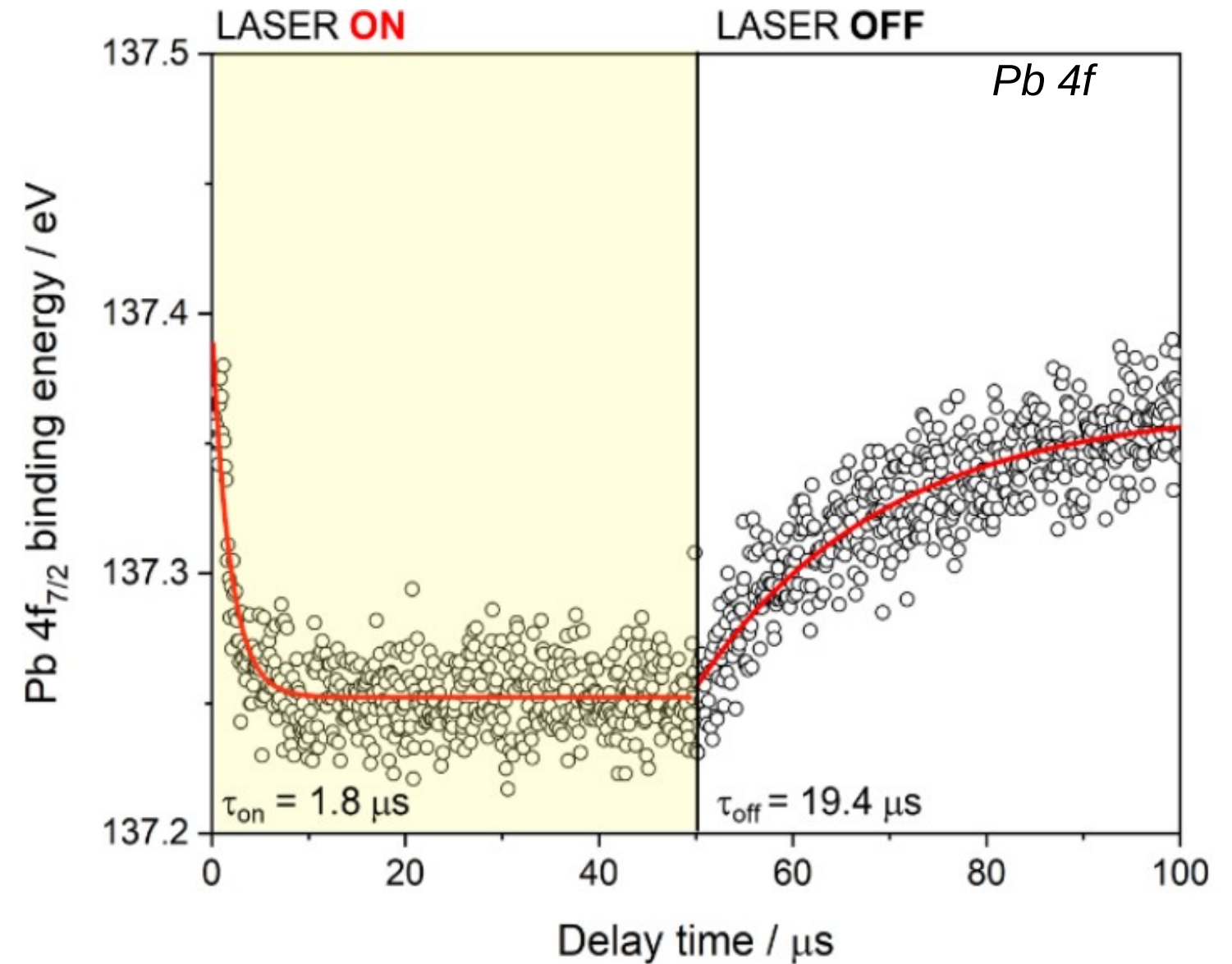
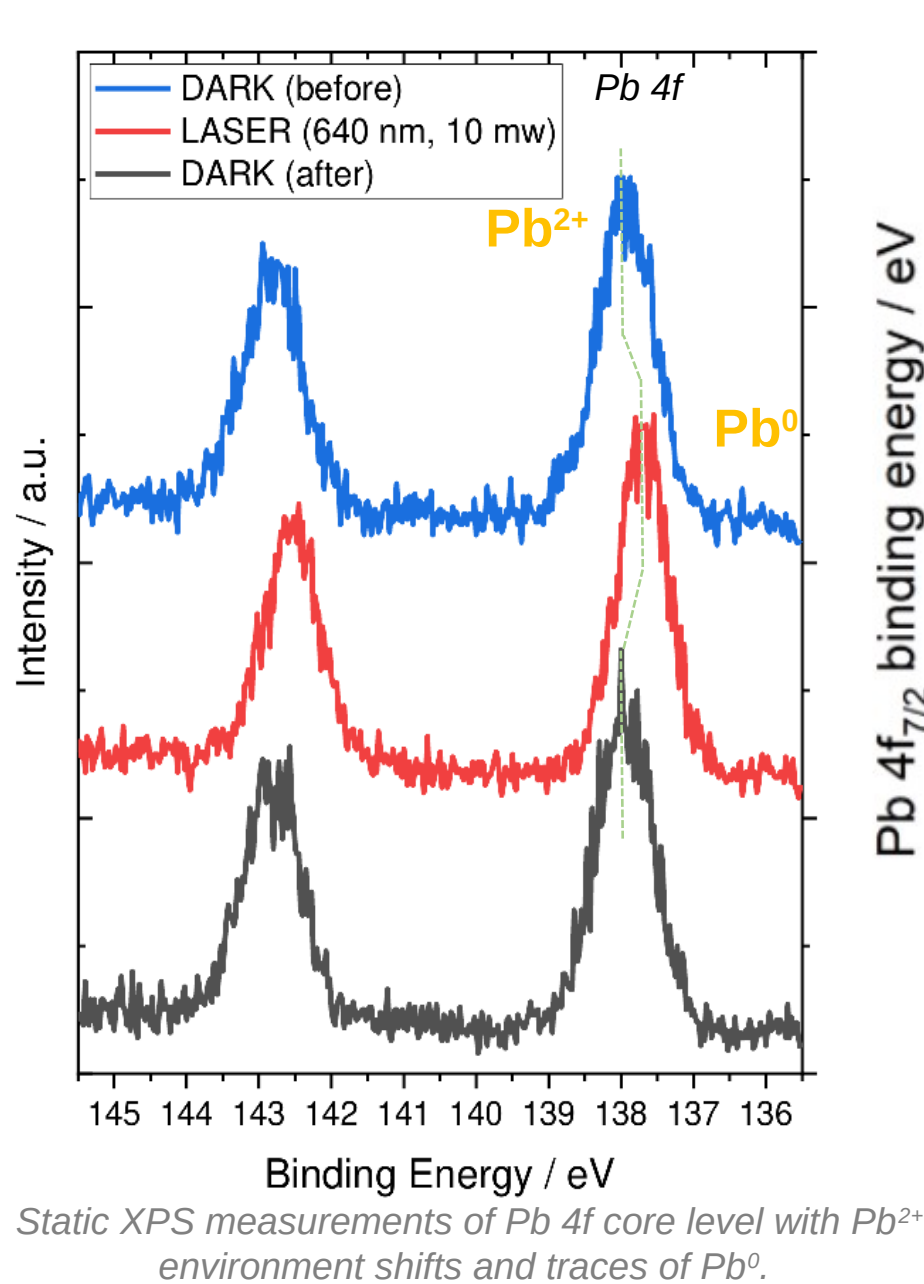
■ 04.08.2023





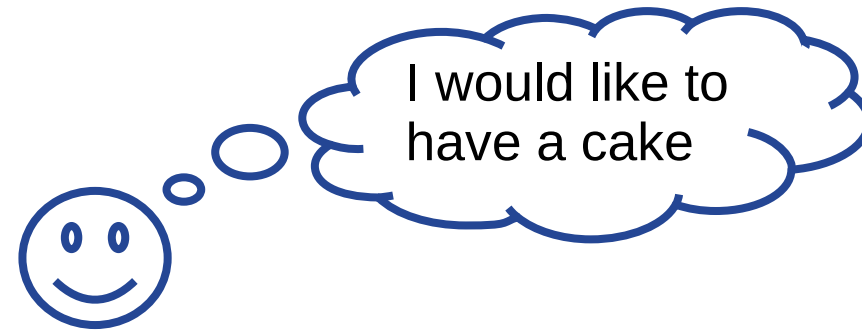
0. Background: What was the stage of affairs regarding Pb analysis when I arrived ?
1. Overview: How is the code structured
2. Understand your data
3. Distill your data
4. Fit the data
5. Summarize the fit results
6. Goals for the last week:
7. Outlook:
  - What do you expect from the last week ?
  - How do you want the code to be handed over ?

# Pb main features were quantifiable – but not minor features



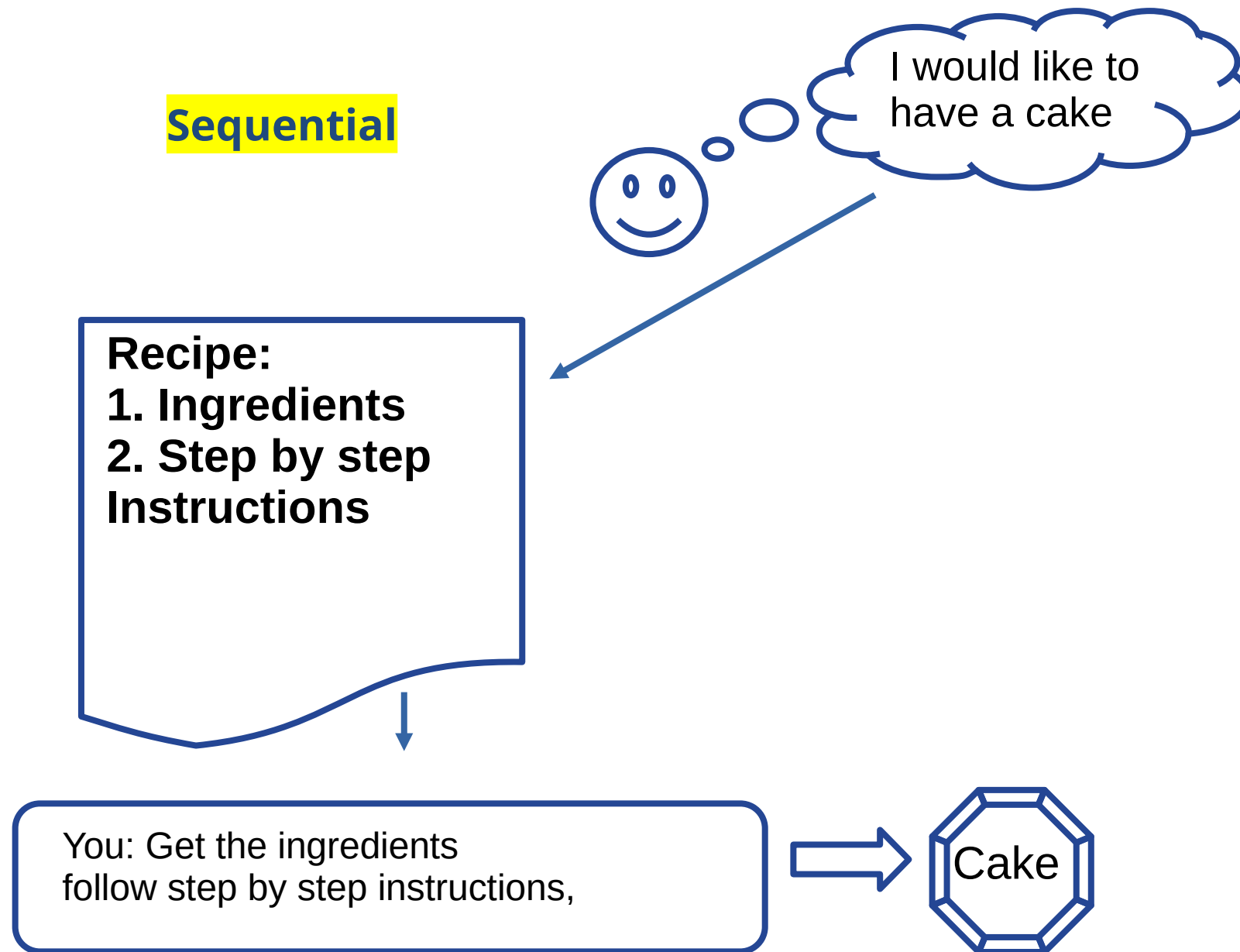
Pb 4f TR-PES data treatment showing the binding energy of the Pb 4f state as a function of time, as the laser light at 640 nm is turned on and off.

# Interlude: Object oriented makes readable, maintainable and reusable code

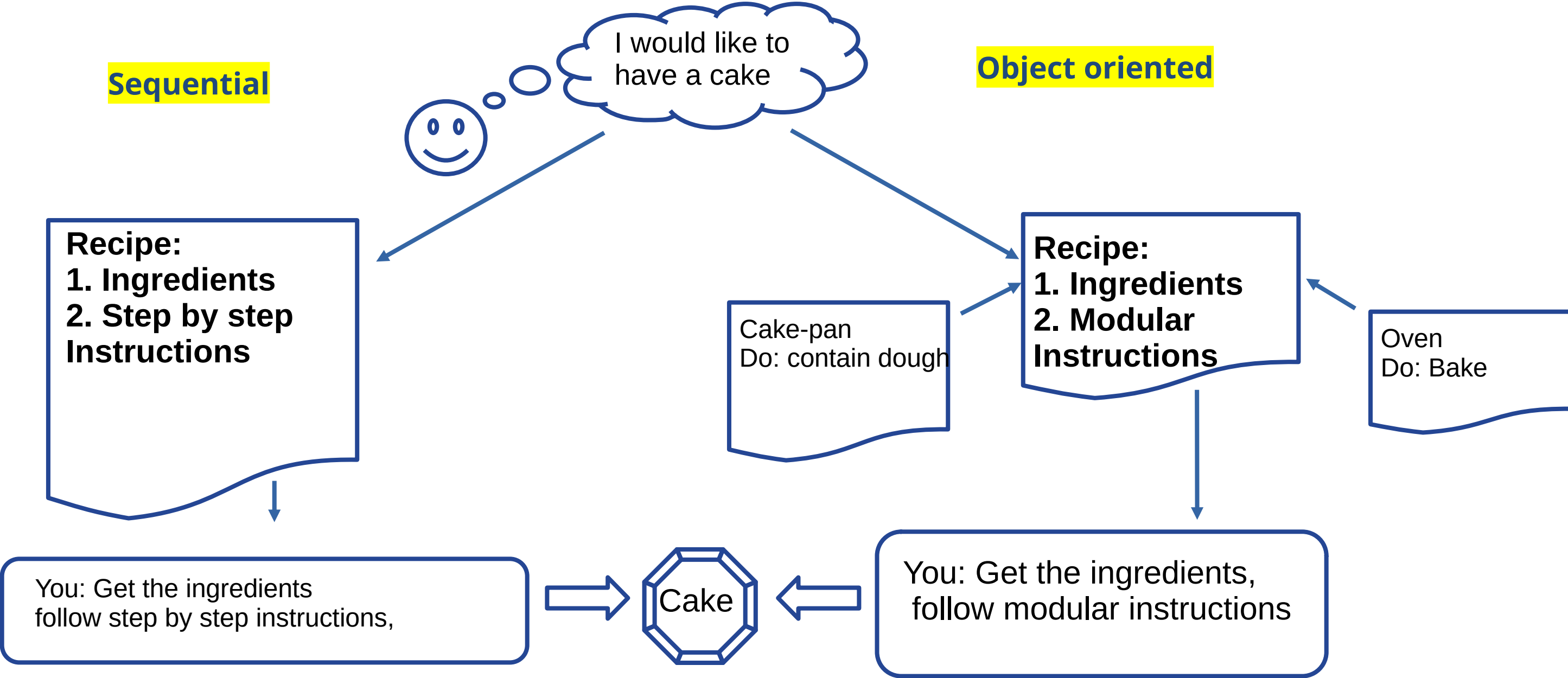


# Interlude: Object oriented makes readable, maintainable and reusable code

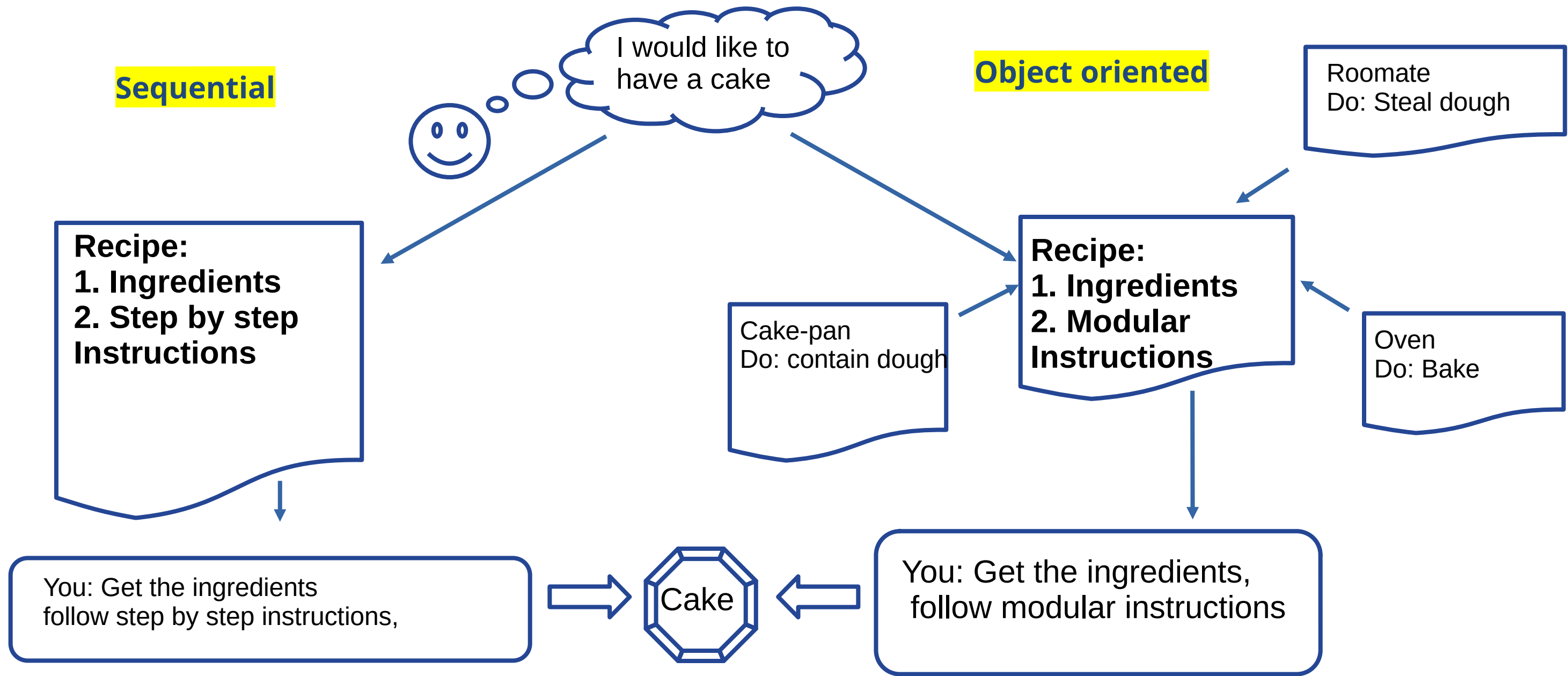
## Sequential



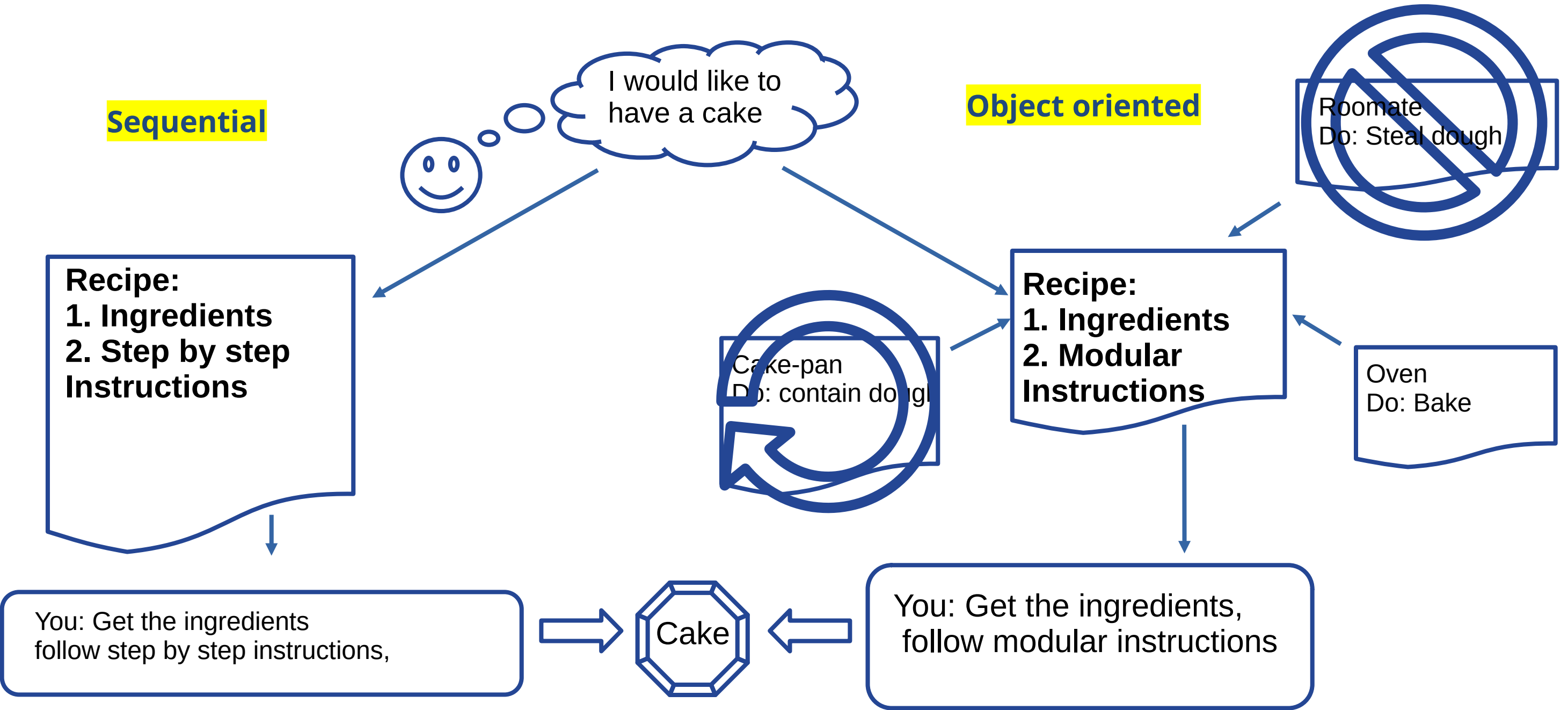
# Interlude: Object oriented makes readable, maintainable and reusable code



# Interlude: Object oriented makes readable, maintainable and reusable code

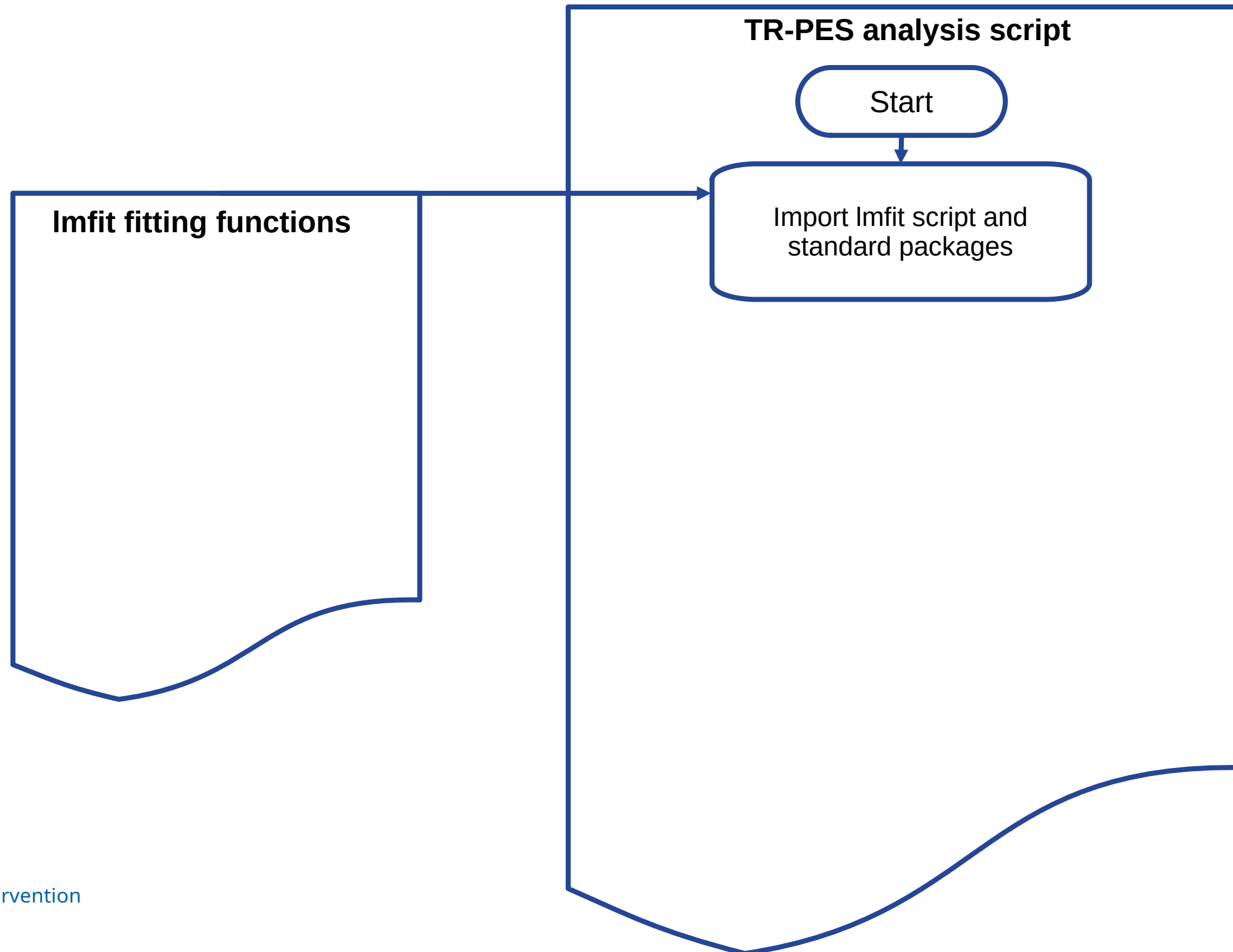


# Interlude: Object oriented makes readable, maintainable and reusable code

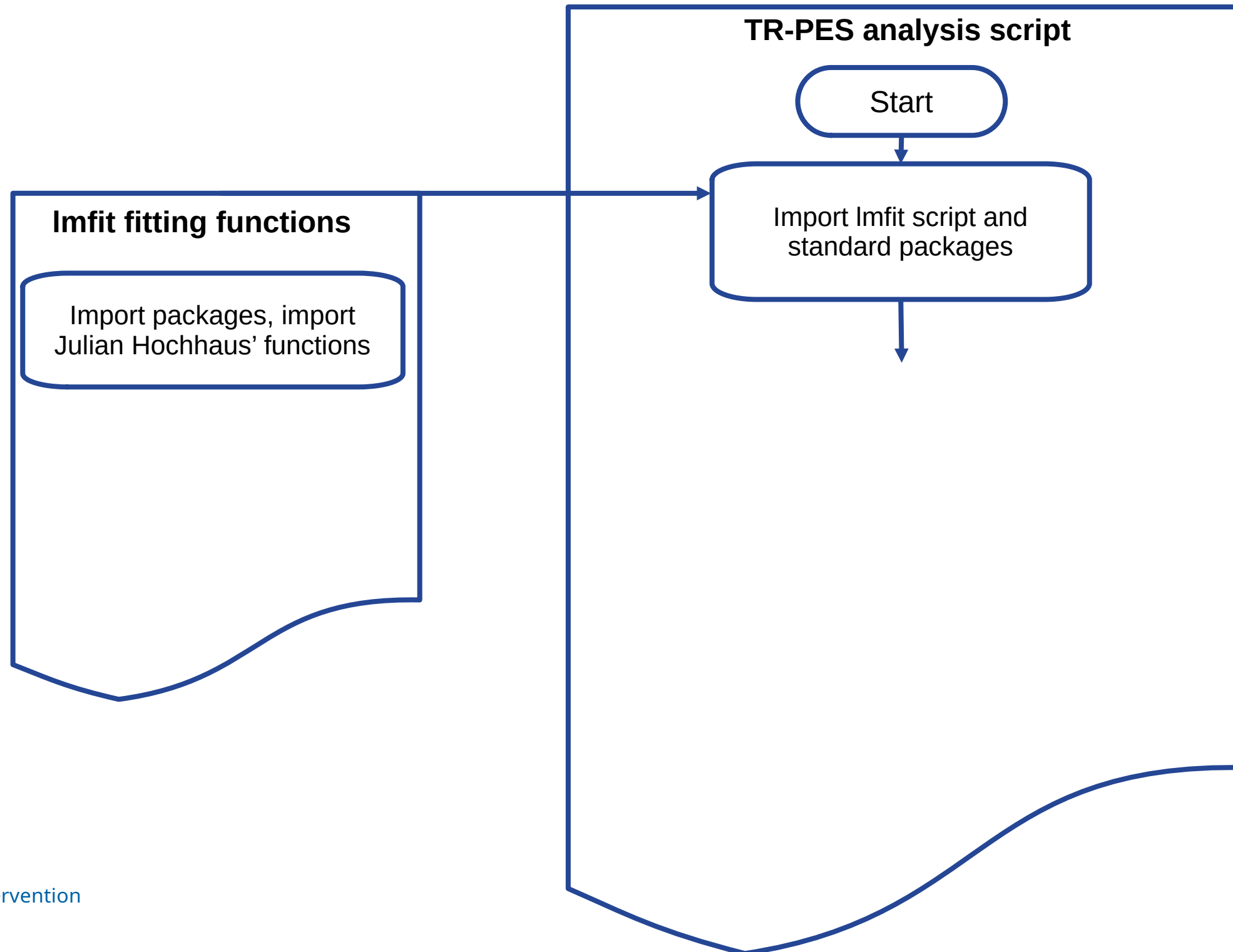




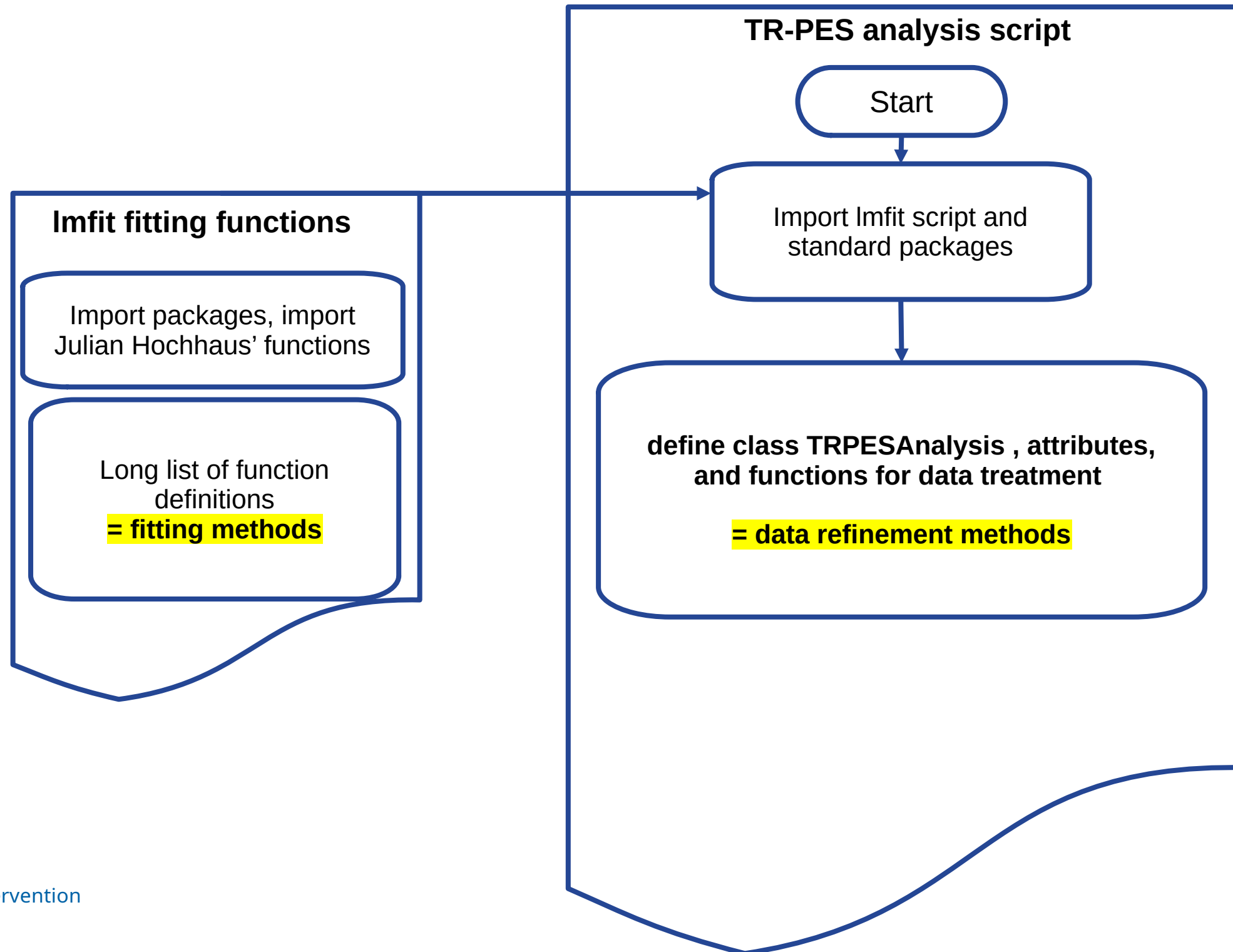
# Overview: One class containing all functions – experiment oriented usage



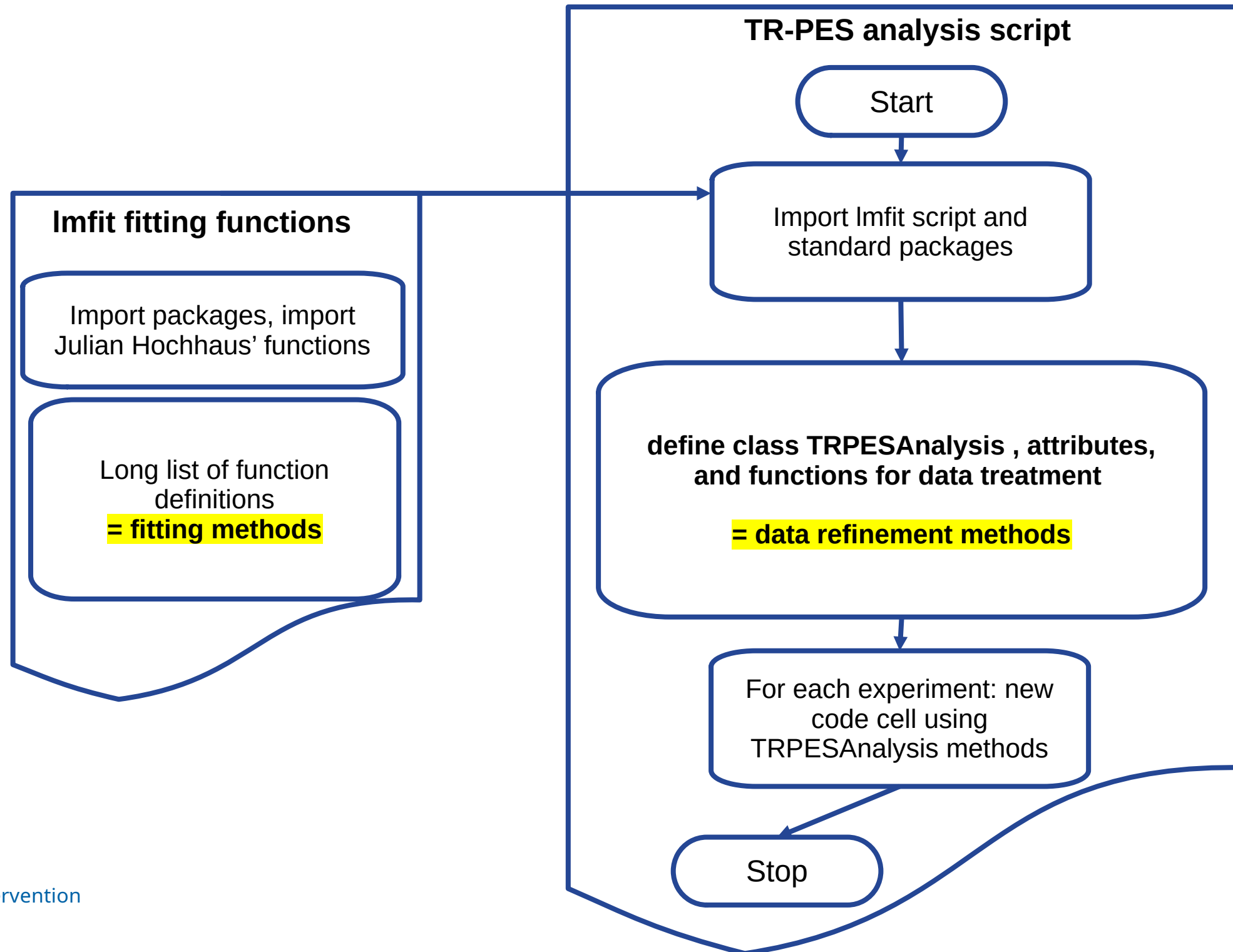
# Overview: One class containing all functions – experiment oriented usage



# Overview: One class containing all functions – experiment oriented usage

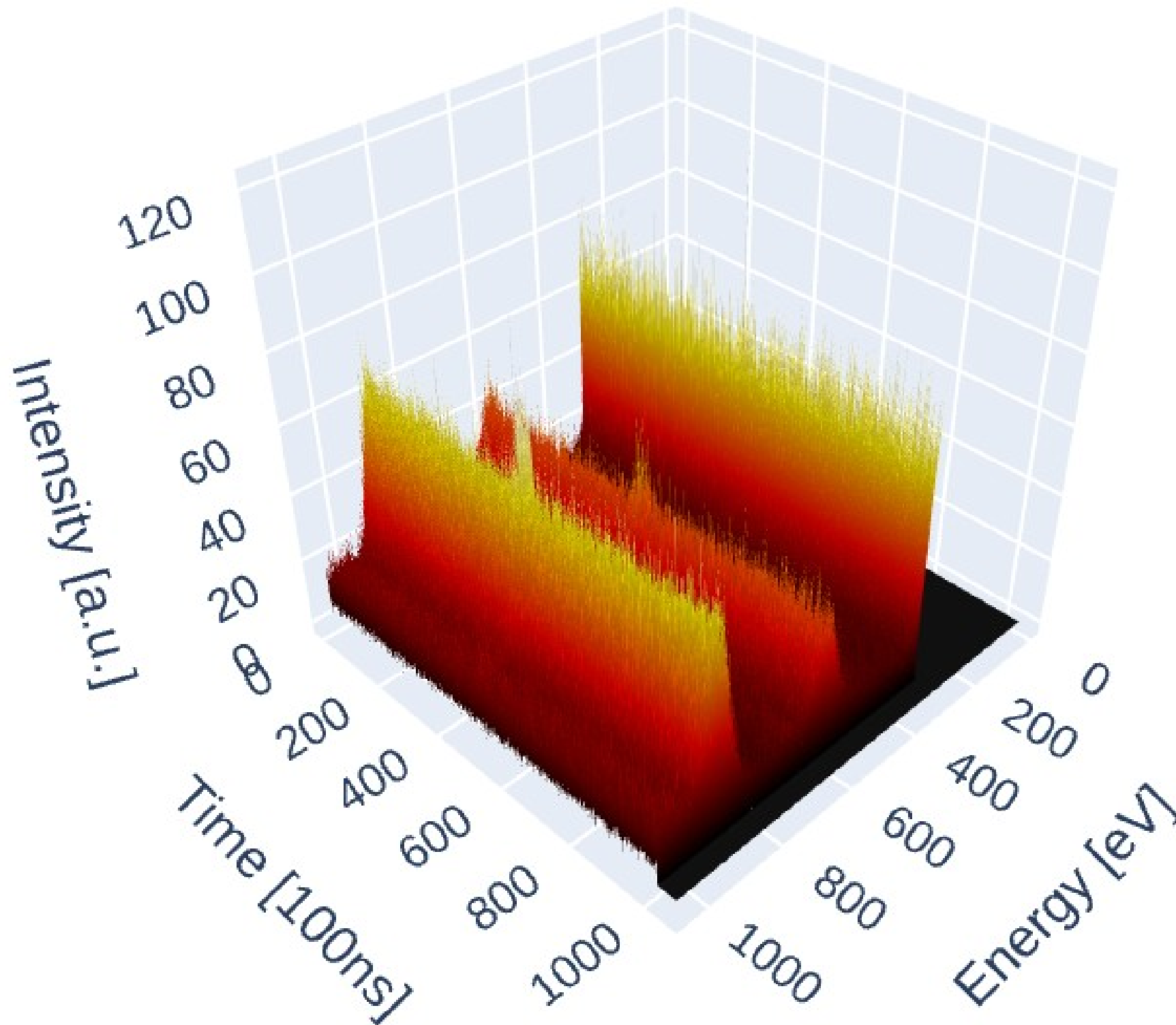


# Overview: One class containing all functions – experiment oriented usage





# 1. Understand your data



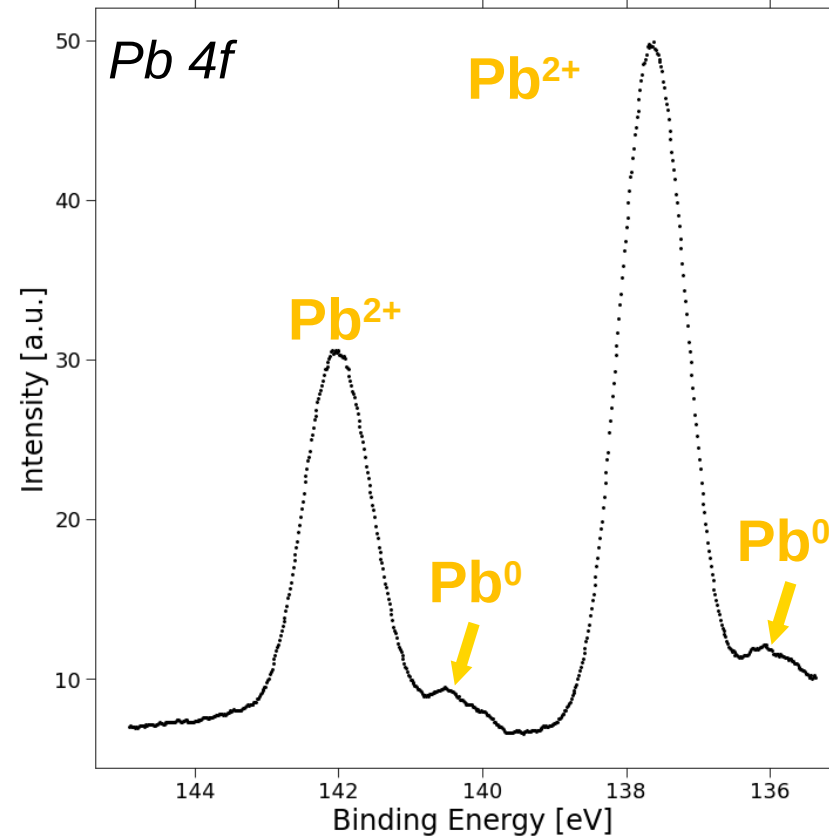
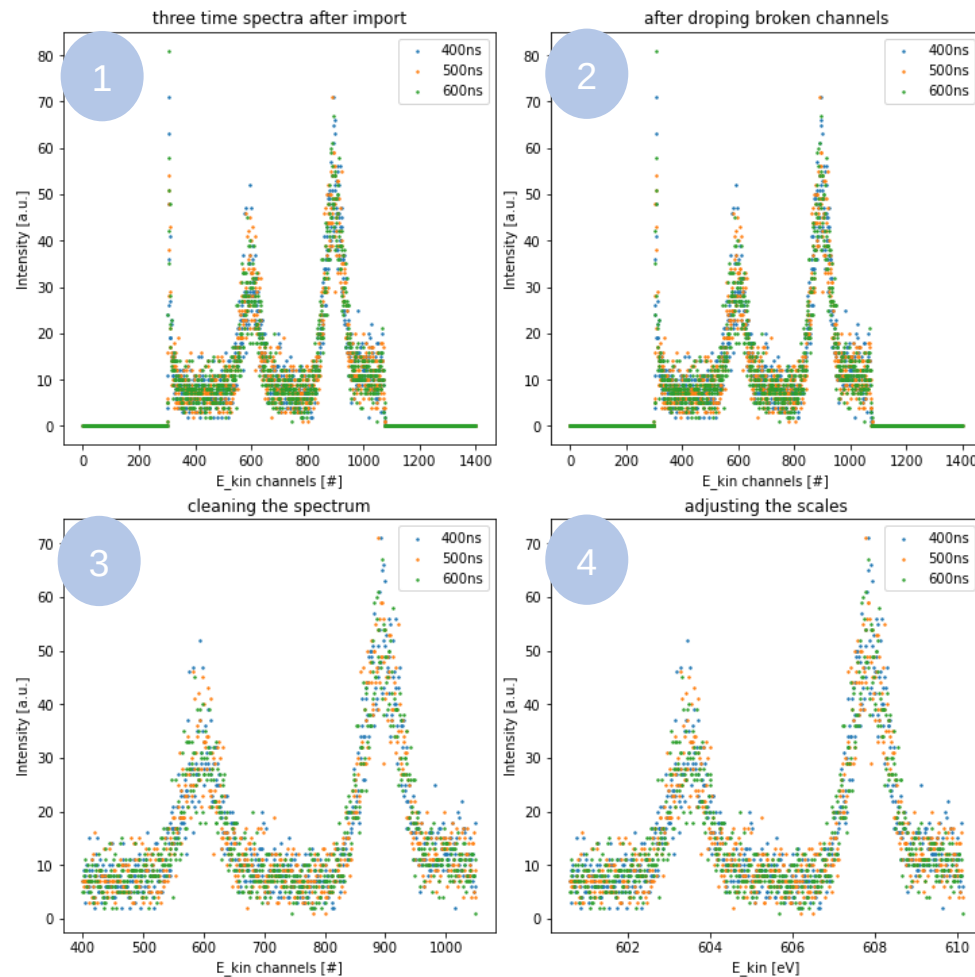
## **TRPESAnalysis.import\_data()**

- loads data and keeps them within the class object loaded as self.data and self.datac (datac for data cleaned)
  - prints data changes and plots the resulting changes

## **TRPESAnalysis.plot\_3d()**

- Takes self.data and plots the 3D data

## 2. Distill your data



### TRPESAnalysis.import\_data()

- loads data and
- 1. Plots raw spectrum
- 2. Drops buffer energy channels in time
- 3. Cuts data range to region of interest
- 4. Adjustes axis scales according to focus energy

### TRPESAnalysis.save\_data()

- saves individual slices of data cube into directory

### TRPESAnalysis.combine\_data\_files()

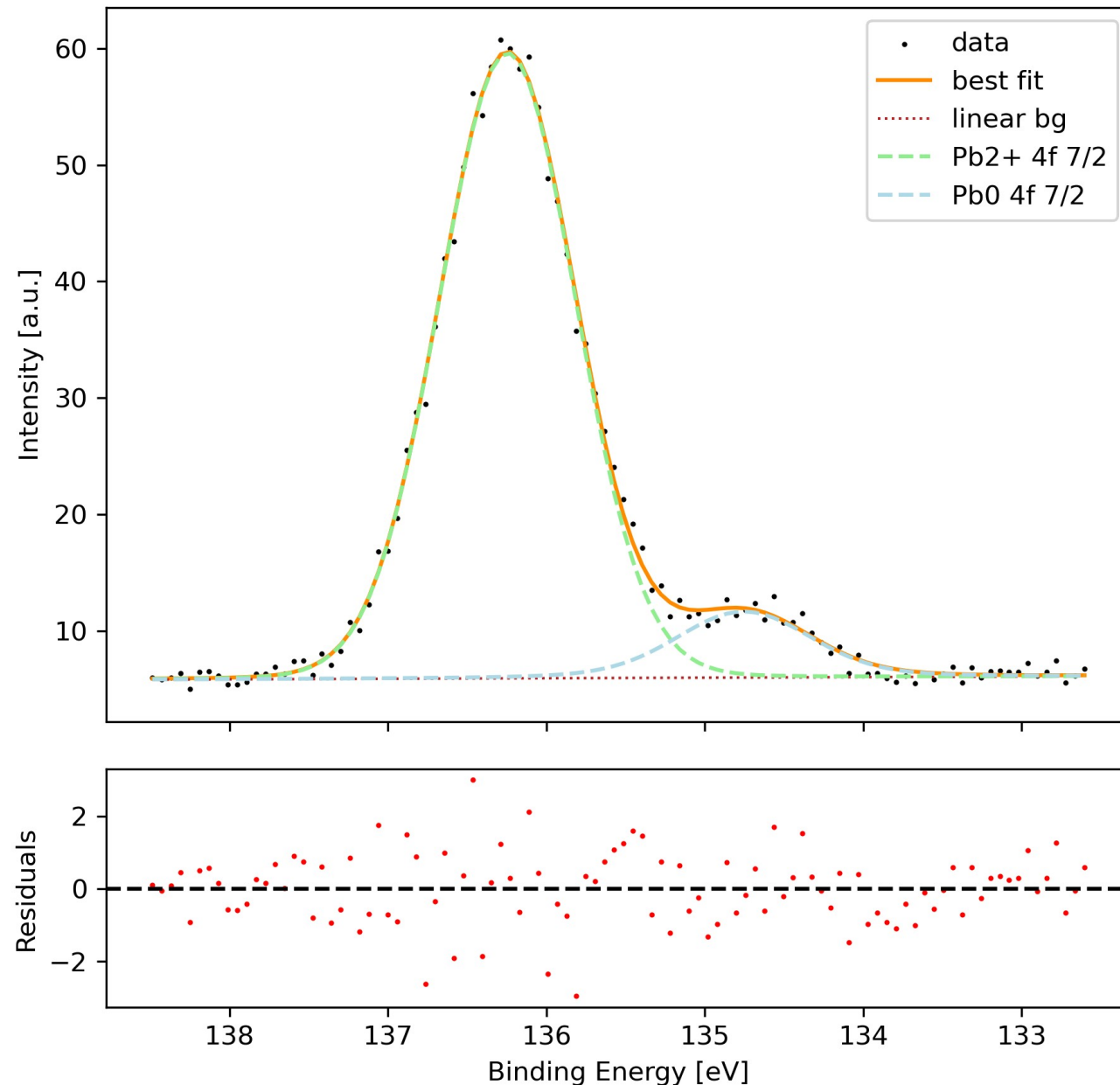
- loads csv files and adds intensity values together

### TRPESAnalysis.smooth\_data()

- takes self.datac and performs rolling average over time and energy. Resulting dataframe called self.datas (for data smoothed)

# 3. Fit your data

FACsPbBrI Pb 4f 7/2 data with 2 fft voigt peak and linear background



## fft\_voigt

- fast fourier transform accelerated voigt function definition

## FFTVoigtModel()

Lmfit model definition based on fft\_voigt definition – careful: **Amplitude in lmfit refers to area under function, not function amplitude**

**Multiple fit functions, some of them being deprecated, which is said in the docstrings**

e.g.

## fit\_two\_fftvoigt\_functions()

- takes spectrum data and initial fit parameter for two voigt model functions and a linear background and optimizes the fit with least-square method

→ all changes regarding initial fitting parameter or method of fitting in here !

Returns: fit plots (e.g. see left), fit reports and fit save files , such that lmfit could load the exact fit in the future

## TODO after 1. fitting:

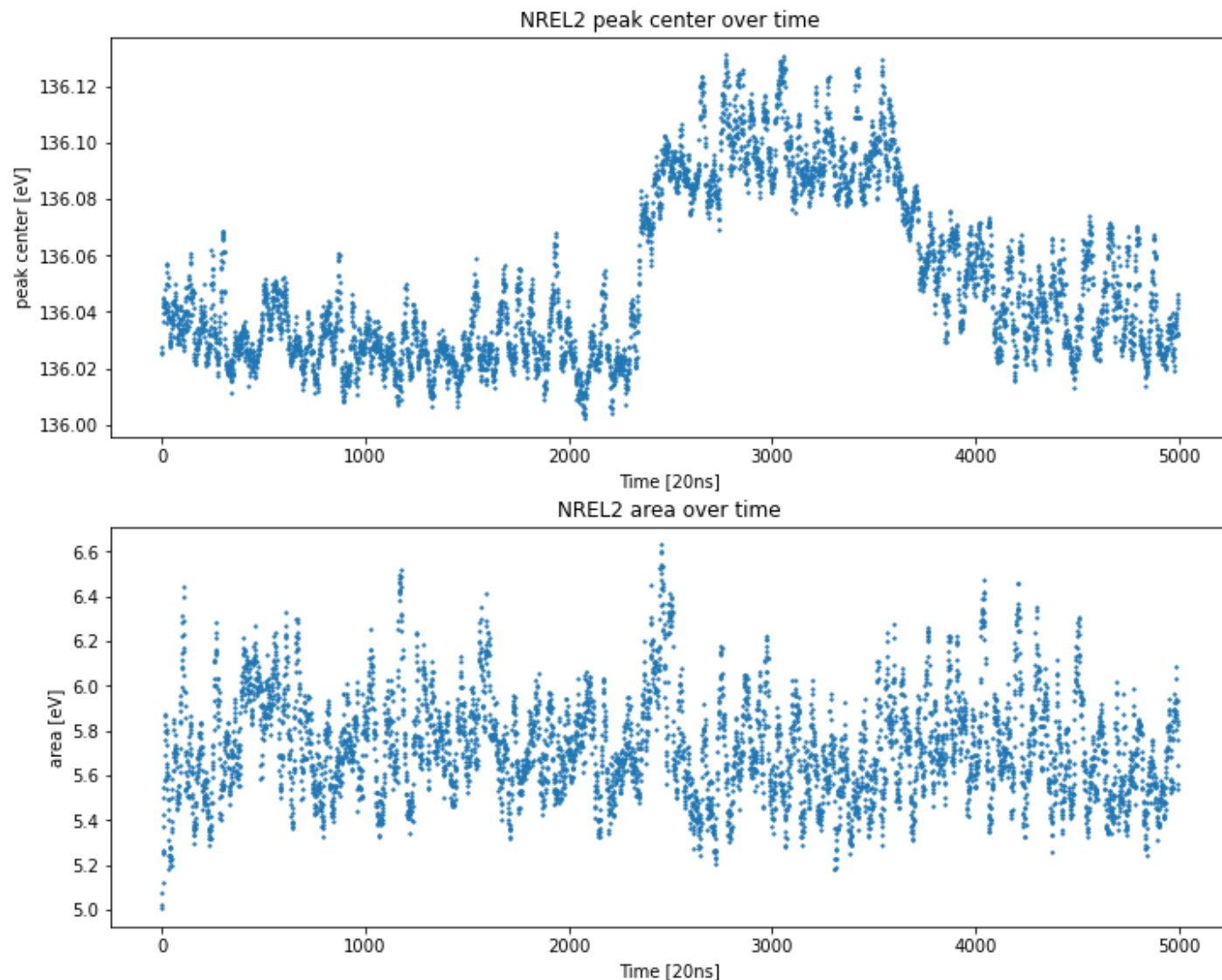
**1. Check for zero fits and refit if necessary**

**2. Extract FWHM ratio constant**

→ **Redo all fits with FWHM ratio constant**

**=> Yields set of analysable data**

## 4. Summarize the fit results



### **TRPESAnalysis.read\_fit\_parameter()**

- reads in all the fit reports and extracts the fit parameter
  - saves a fit\_summary.csv file which contains the extracted results for up to 4 voigt functions and background choice

### **TRPESAnalysis.split\_Pb4f\_fit\_data()**

- splits the fit\_summary.csv into fixed arrays representing the Pb4f signatures

- all changes regarding which fit parameter should be recorded in here and the dependent functions

**Multiple TRPESAnalysis.plot\_X functions, all meant for visualizing the fit results. Some are functional, most are not, this is said in the docstrings**

**e.g. :**

### **TRPESAnalysis.plot\_figure\_grid()**

- plots four plots showing a single attribute of the two Pb2+ and two Pb0 peaks (e.g. peak center shifts of all four peaks)

### **TRPESAnalysis.plot\_correlation\_matrix\_seaborn()**<sup>16</sup>

- plots the correlation matrix of two peaks and two attributes in seaborn layout style



- What do you expect from the last week ?
- How do you want the code to be handed over ?