LPC-DEV, LPC Caen, CNRS/IN2P3

4-8th November 2024

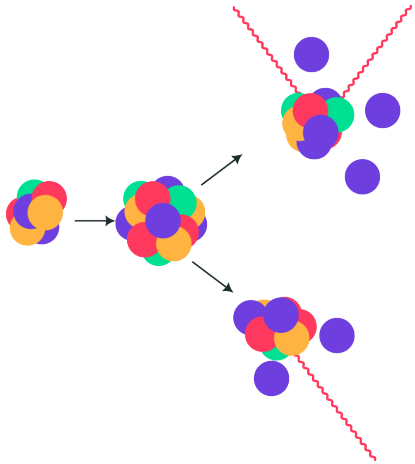
UNIVERSITÉ
CAEN
NORMANDIE

Normandie Université

Physics case

Fission induced reaction

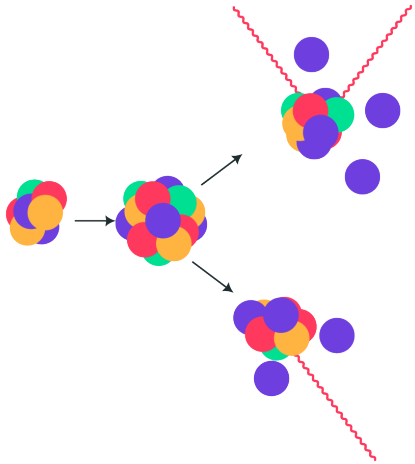
- Initial Reaction:
  $^{235}U + {}^{9}Be \rightarrow {}^{A1}X^{*} + {}^{A2}Y^{*}$



Physics case

Fission induced reaction

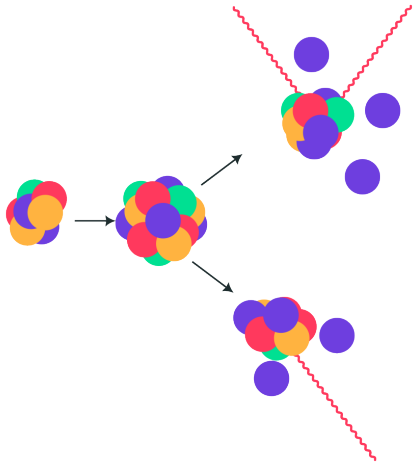
- Initial Reaction:
  $^{235}\mathrm{U} + {}^{9}\mathrm{Be} \rightarrow {}^{A1}\mathrm{X}^{*} + {}^{A2}\mathrm{Y}^{*}$
- Neutron Evaporation:
  ${}^{A1}\mathrm{X}^{*} \rightarrow {}^{A3}\mathrm{X}^{*} + \mathrm{V}n$
  ${}^{A2}\mathrm{Y}^{*} \rightarrow {}^{A4}\mathrm{Y}^{*} + \mathrm{W}n$



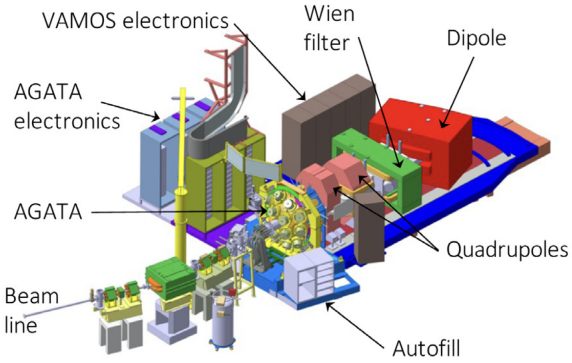
Physics case

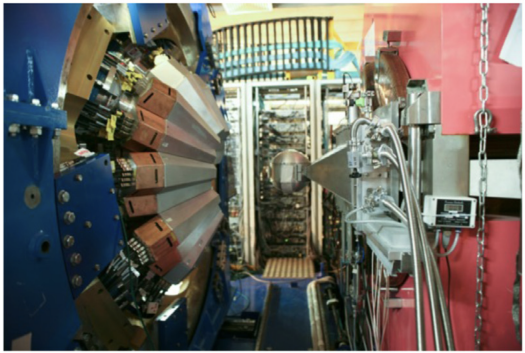
Fission induced reaction

- Initial Reaction:
  $^{235}\mathrm{U} + {}^{9}\mathrm{Be} \rightarrow {}^{A1}\mathrm{X}^{*} + {}^{A2}\mathrm{Y}^{*}$
- Neutron Evaporation:
  ${}^{A1}\mathrm{X}^{*} \rightarrow {}^{A3}\mathrm{X}^{*} + \mathrm{V}n$
  ${}^{A2}\mathrm{Y}^{*} \rightarrow {}^{A4}\mathrm{Y}^{*} + \mathrm{W}n$
- Deexcitation:
  ${}^{A3}\mathrm{X}^{*} \rightarrow {}^{A3}\mathrm{X} + \mathrm{P}\gamma$
  ${}^{A4}\mathrm{Y}^{*} \rightarrow {}^{A4}\mathrm{Y} + \mathrm{Q}\gamma$



Detection setup: AGATA-VAMOS





What we detect

Fragment:

- A and Z
- θ, ϕ,β

Gamma:

- γ energy
- First interaction position

Physics goals

- Produce Histogram for a given nuclei
- Check Data vs Known Gamma ray
- Found new Gamma in ^{90}Br,^{96}Kr,^{102}Sr,^{97}Kr

Skill goals

- Basics of collaborative programming
- Efficient workflow
- Reproducible analysis

Dataset

- Prepared by J. Dudouet (IP2I)
- GEF+TKEN+SToGS (Geant4)
- nndc decay + some extra physics :D

Cloning and forking project-zero

Goals

We will start from a base project. For this a basic CMake based project is provided. We will first clone and test the project, then fork it to create a new project we own.

Instruction

- Perform a fork of the project at gitlab.in2p3.fr/eurolabs-os-school/teaching/project-zero.git
- Clone your project
- Make sure you have access to the data (/data/bootcamp_2024/postprocess_agata/run)
- Follow the README instruction and execute the code.

Reading data

Goals

The provided sample code read the input data and copy it partially. In this exercice we will add the missing information so the code read and copy the data completly.

Instruction

- Student A : Add missing fragment data
- Student B : Add missing γ data
- Together: Commit and merge

Building a test suit

Goals

We would like to make sure our program works well. We will start to implement test within the project

Instruction

- Add a test folder and populate it with a CMakeLists.txt and a test source code
- Implement a test that check the input and output file are identical
- Add CI/CD to your repository
- Add the test to your CI/CD
- make a commit that break the test
- restore the code

Histogramming project-zero

Instruction

In this exercice we will produce our first histogram. The histogram will be gated on a given nuclei.

There is two functionnality to implement. The first one is event selection. The second one is to generate the histogram.

Instruction

- Create a new binary in your cmake project copying the original one
- Add new argument to the executable to select A and Z
- Create a function to select an event based on A and Z
- Instantiate an histogram and write it to the output file
- Fill the histogramm on selected event

Doppler correction

Goals

The obtained histogram in the previous example don't show much peaks. The gamma rays need to be doppler corrected.

Instruction

- Implement a doppler correction function
- Add a unit test for your doppler correction function
- Check that your spectrum looks correct
- Propose and implement a gobal test

Doppler correction formula

$$E_{DC} = E_{NDC}.\gamma(1-\beta cos(\theta))$$

Automation with snakemake

Goals

We will add a snakefile to the project to analyse the data in batch and produce a single histogram with all the data.

Instruction

- Create a snakefile
- Add a rule producing one histogram
- Add a rule adding all histograms (using hadd)

hadd command argument

hadd <outfile> <infile1> ... <infile2>

Automation with snakemake

Goals

When using all the statistics the peak dissappear. You will first implement a diagnostic showing the nature of the problem. Then you implement a solution

Instruction

- Create an histogram that show if the calibration is changing by event or by run.
- Implement the calibration in the code
- Implement the calibration dependency in snakemake
- Regenerate the histogram for the whole statistics

Calibration file

/data/bootcamp_2024/postprocess_agata/calibration