

Investigating the Strong Nuclear Force with the OSG

Connor Natzke

July 9, 2022

There are four fundamental forces in nature

Gravity

Binds the Solar System together

Electromagnetic

Binds atoms together

Strong

Binds the atomic nucleus together

Weak

Radioactive decay

There are four fundamental forces in nature

Gravity

Binds the Solar System together

Electromagnetic

Binds atoms together

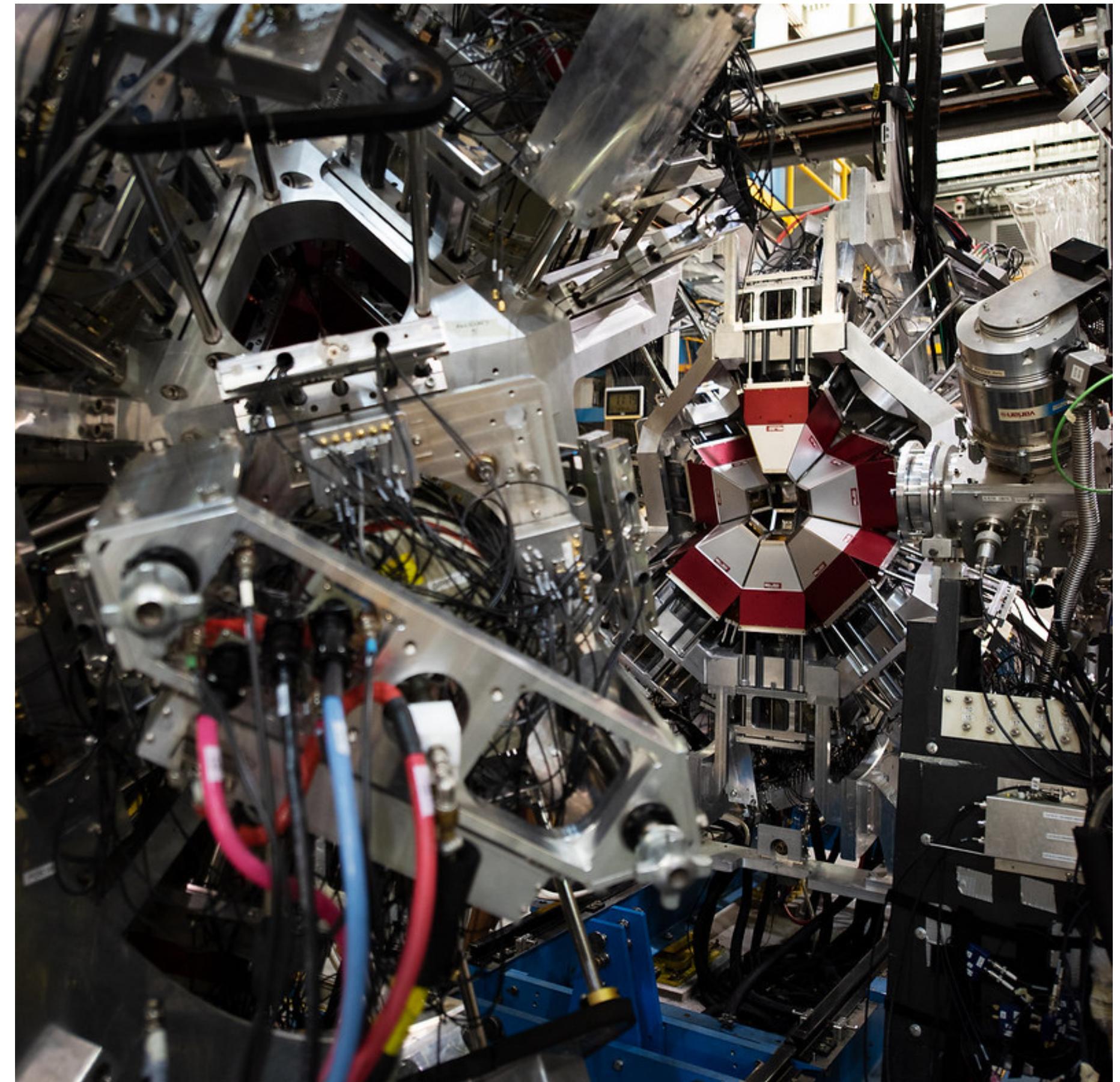
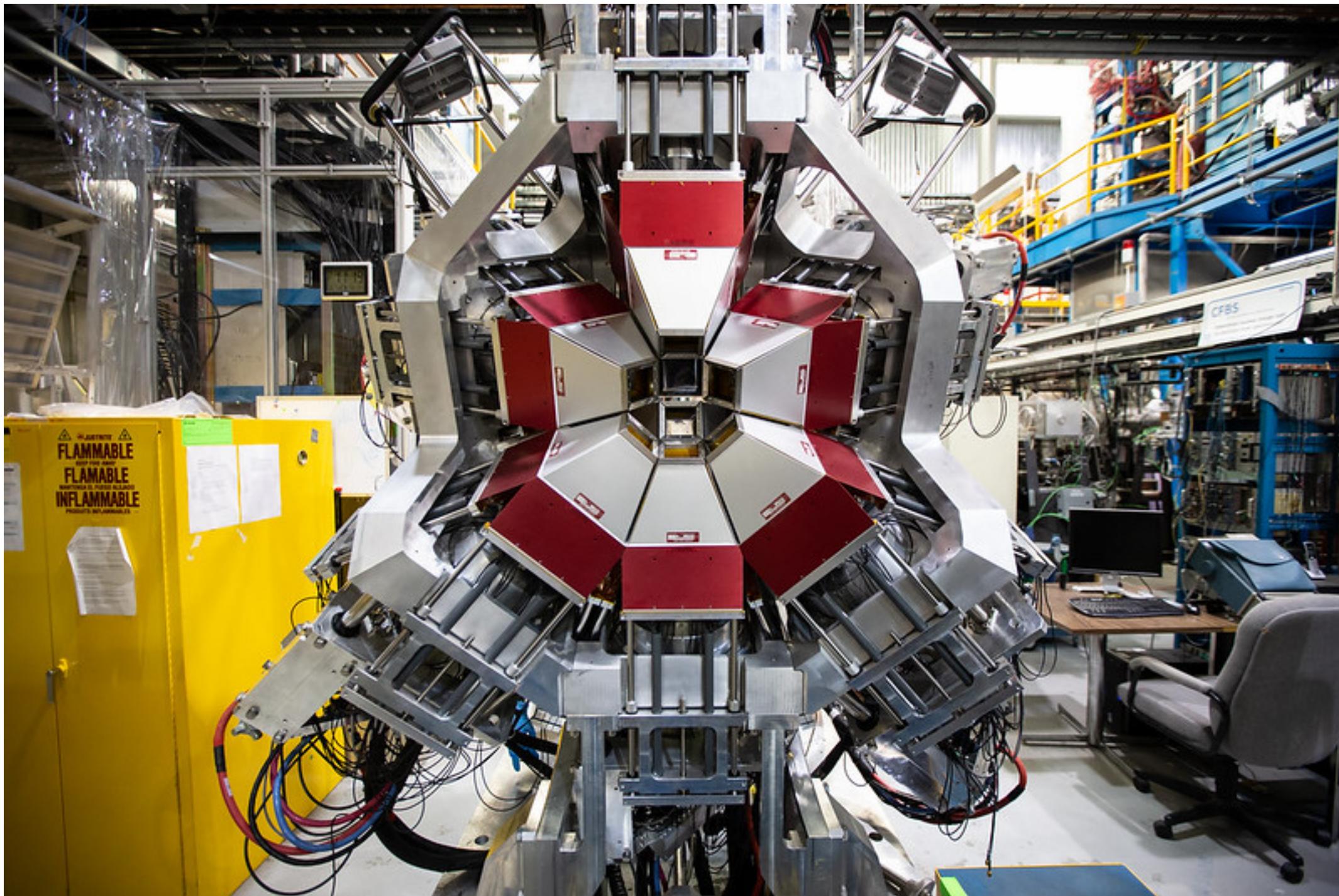
Strong

Binds the atomic nucleus together

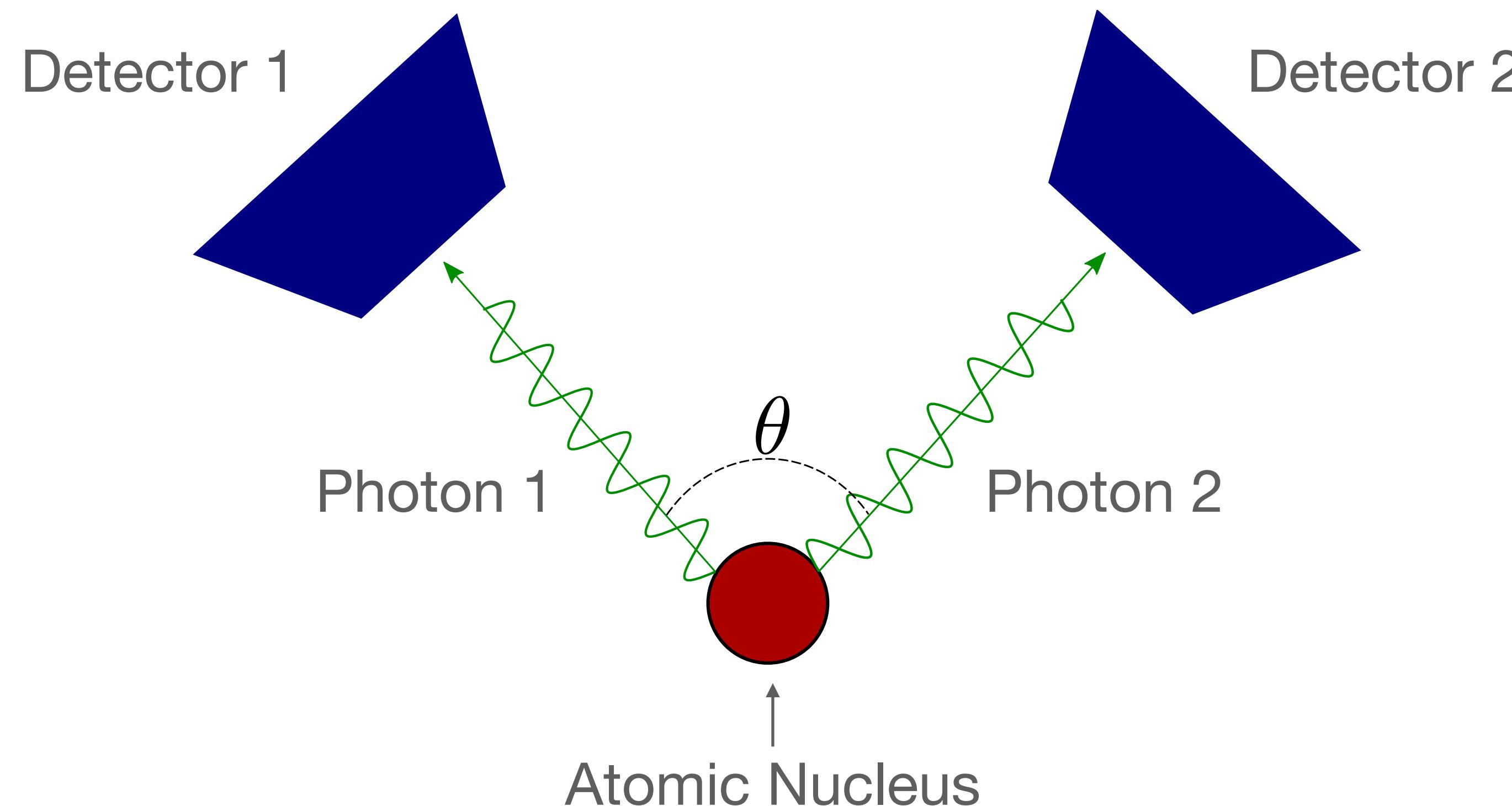
Weak

Radioactive decay

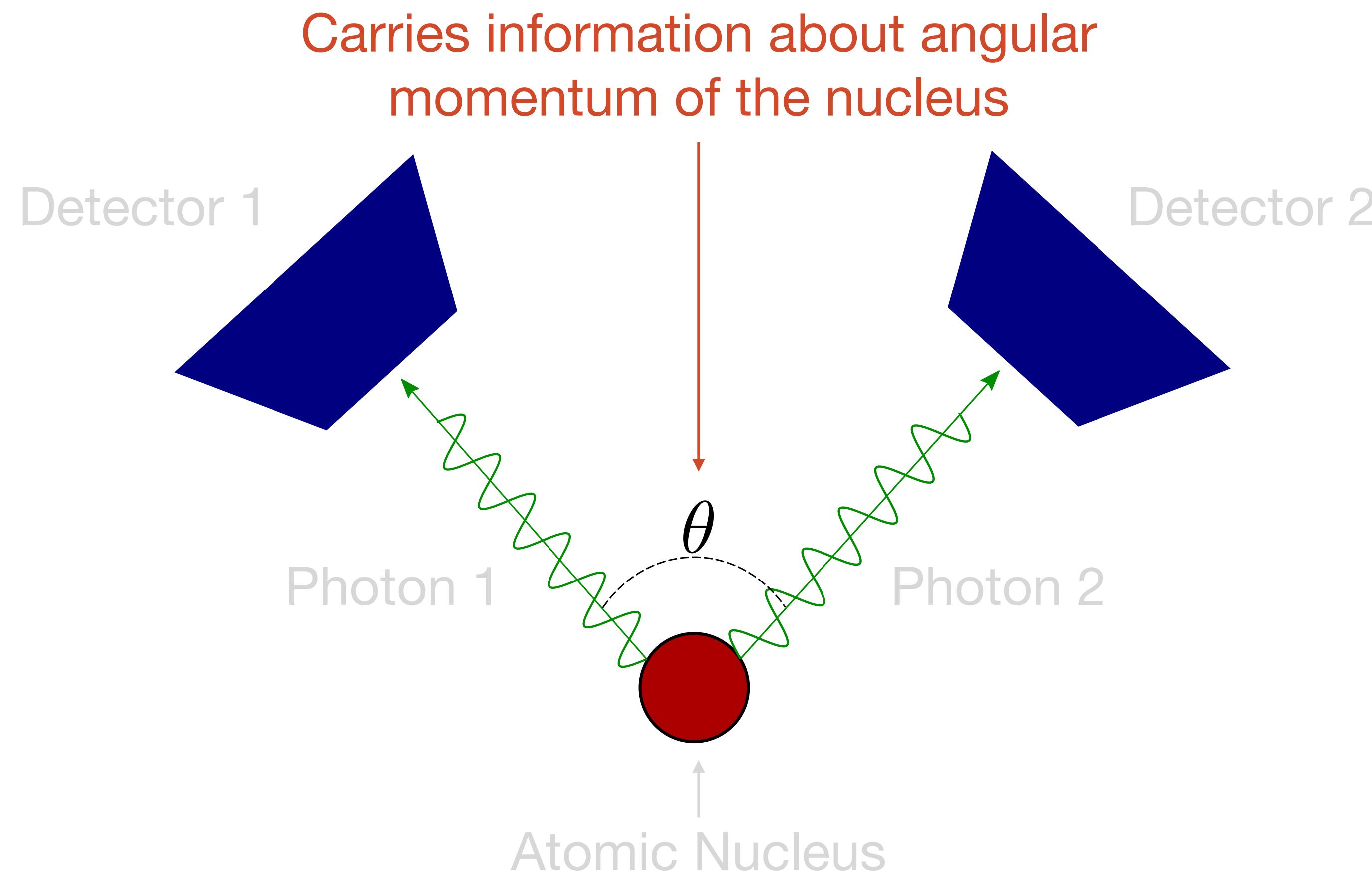
The smaller something is the larger the microscope needs to be



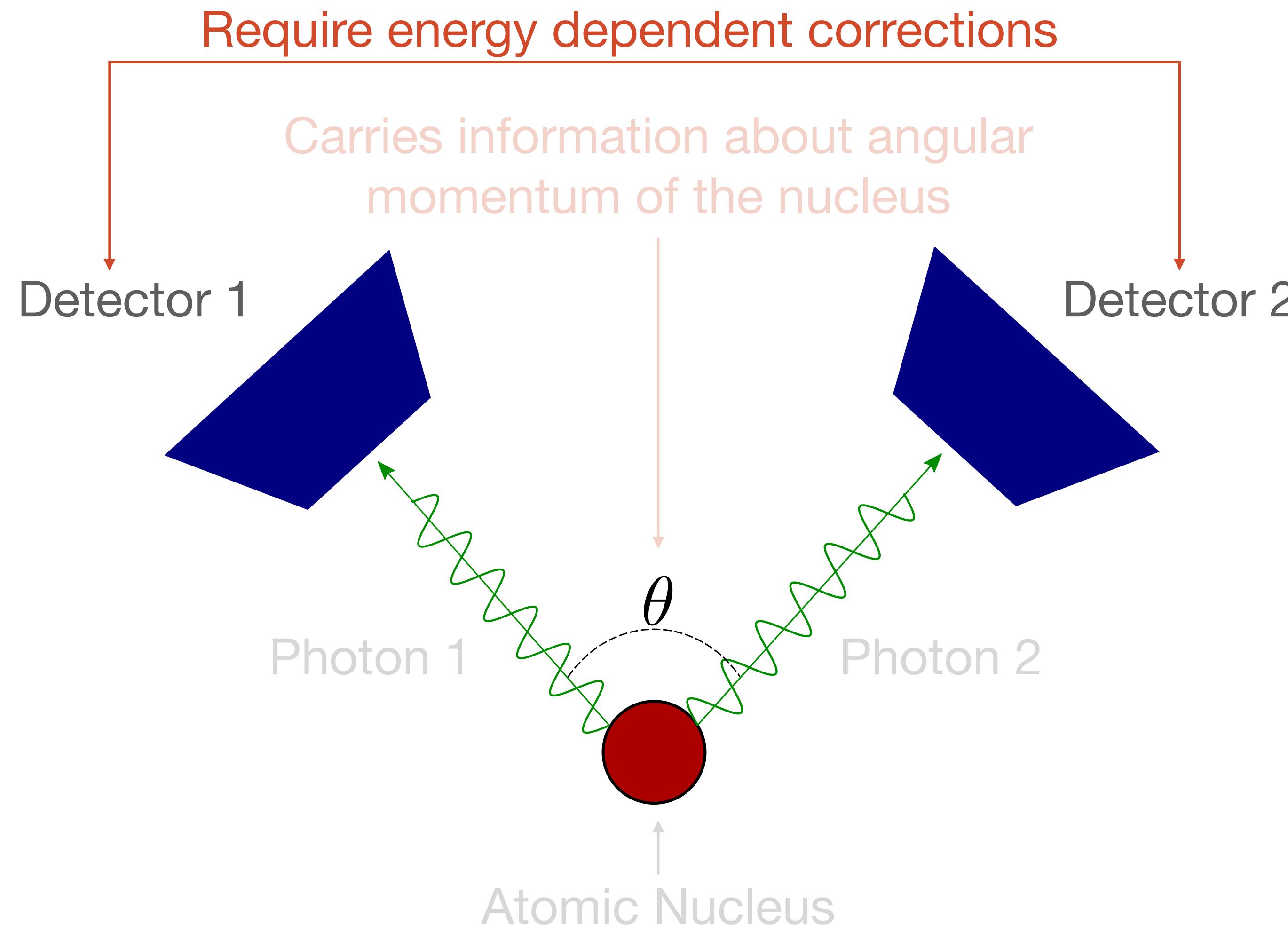
Radiation emitted from atomic nuclei carries information about the structure



Radiation emitted from atomic nuclei carries information about the structure



Radiation emitted from atomic nuclei carries information about the structure

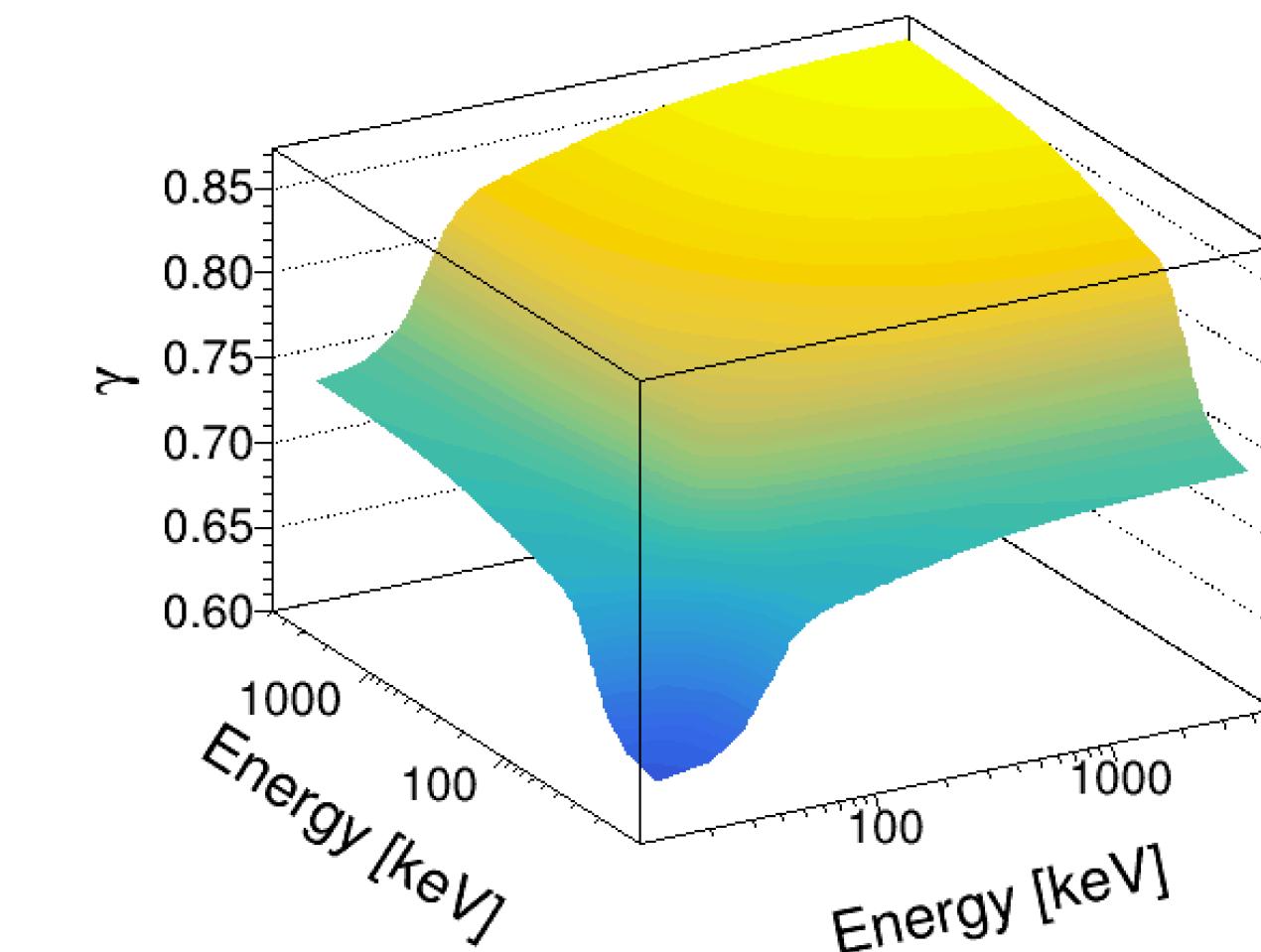
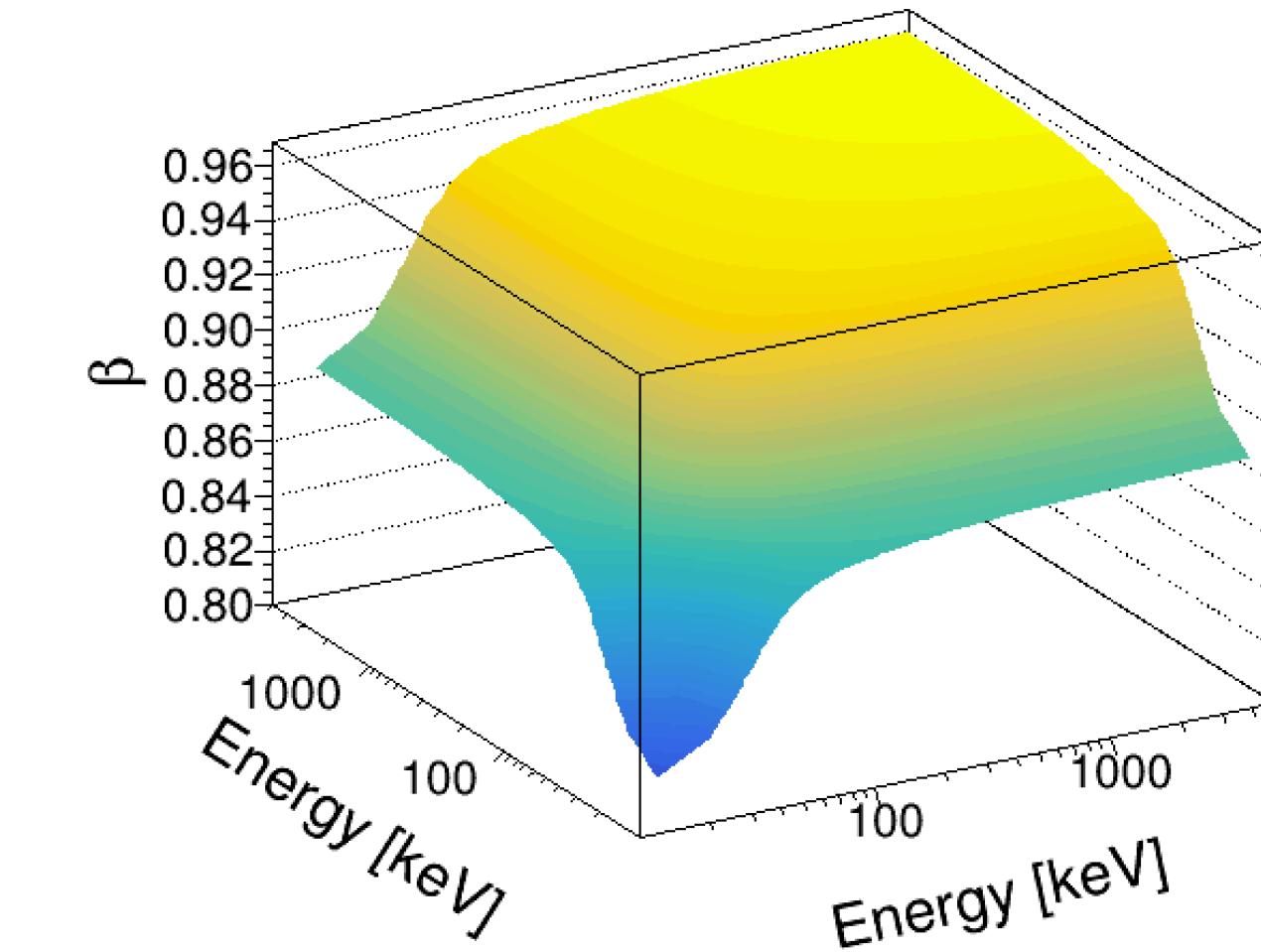


Detectors require correction factors found by mapping an energy surface

Map surface via Monte Carlo simulation

41 points required to map surface

- 3 simulations per point
- 1e9 events per simulation
- ~400 CPU hrs per simulation



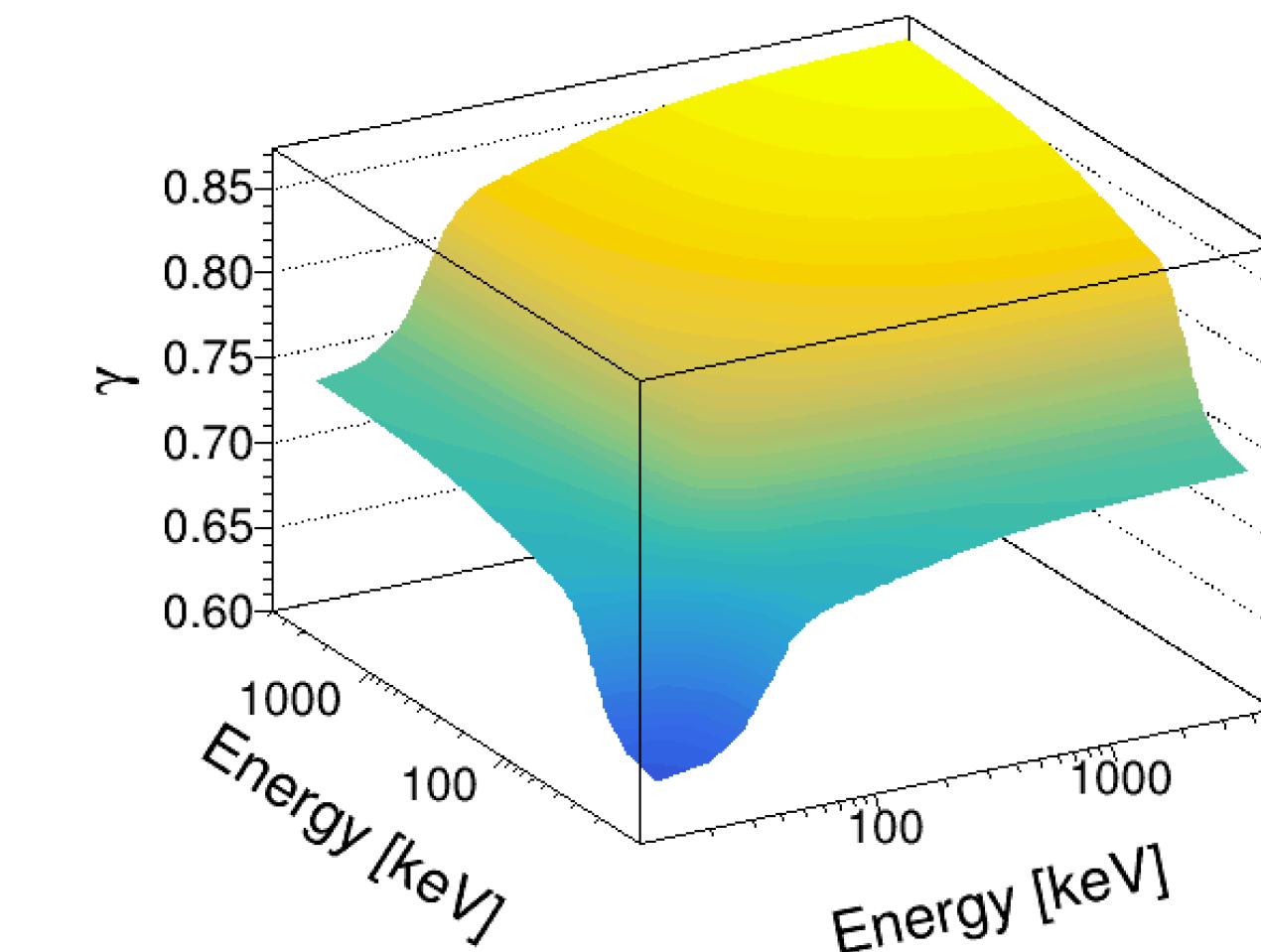
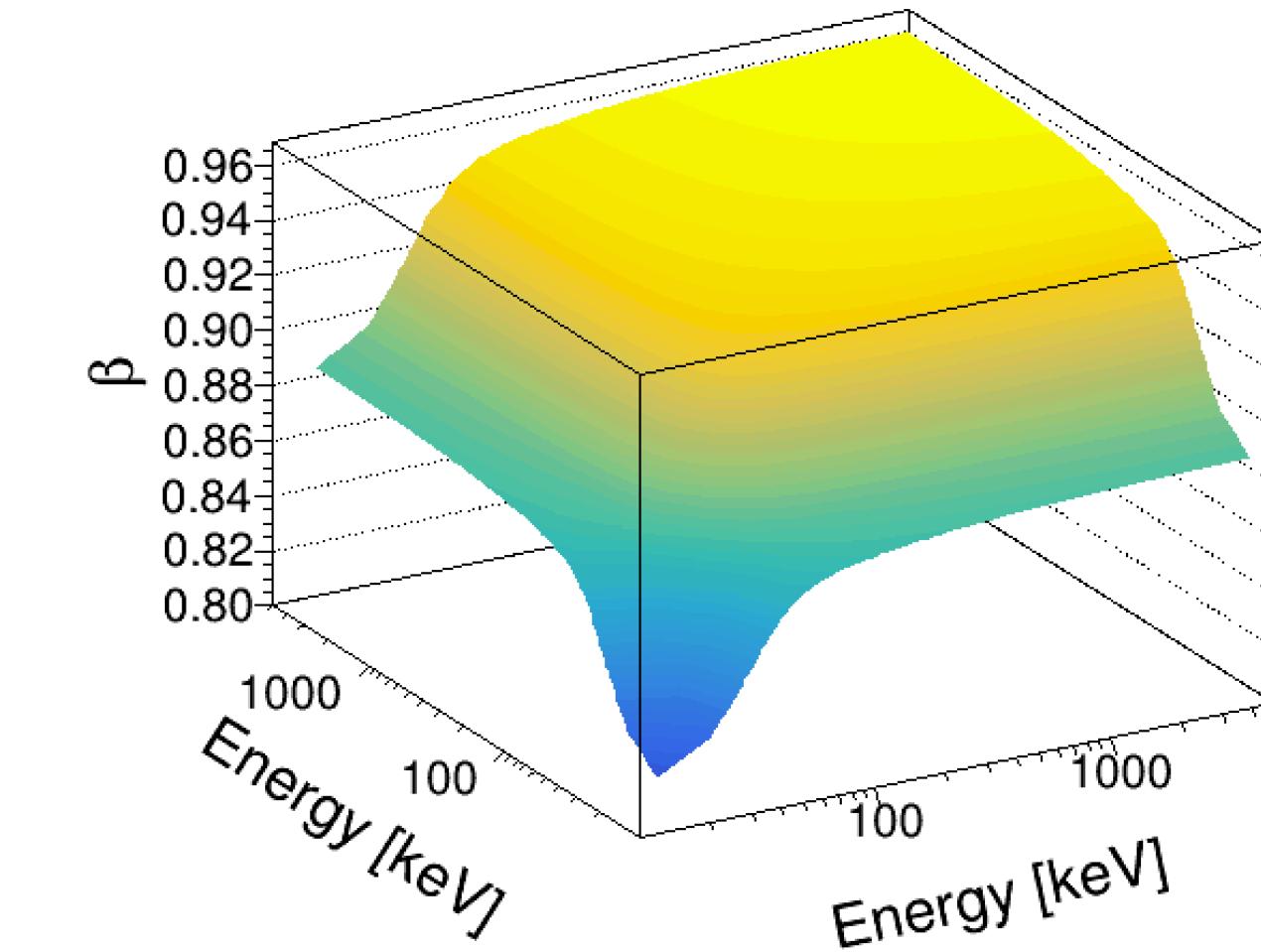
Detectors require correction factors found by mapping an energy surface

Map surface via Monte Carlo simulation

41 points required to map surface

- 3 simulations per point
- 1e9 events per simulation
- ~400 CPU hrs per simulation

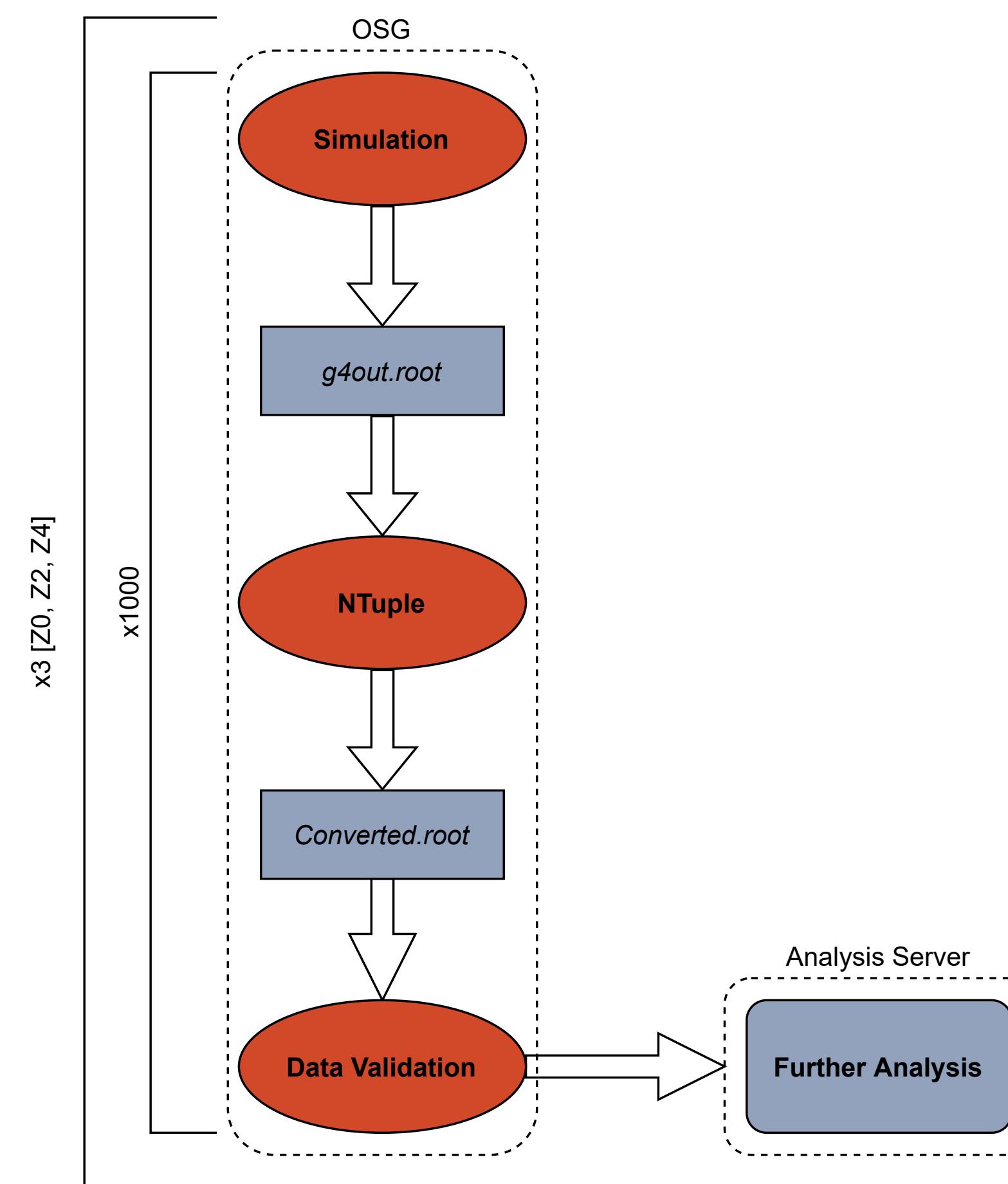
~50,000 CPU hours = 5.7 years!



OSG User School 2019



One simulation of $1e9$ events broken up into 1000 simulations of $1e6$ events



Building an OSG workflow is an iterative process, and it doesn't need to be perfect!

Submission File

Gotta start somewhere

DAGMan

Automation is key

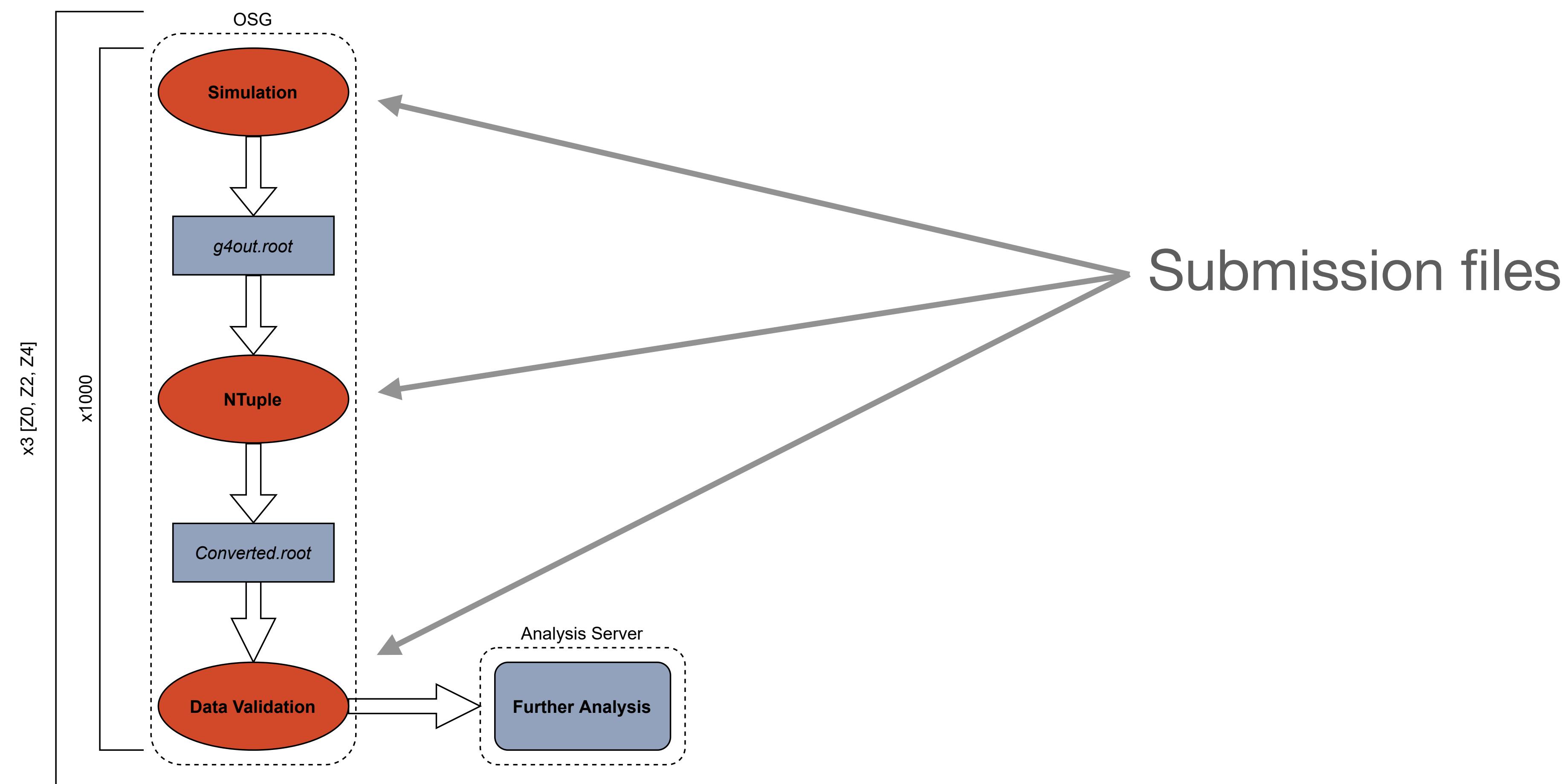
Pegasus

Proper programming makes life easy

Queued Pegasus

Embrace the laziness

One simulation of $1e9$ events broken up into 1000 simulations of $1e6$ events



Building an OSG workflow is an iterative process, and it doesn't need to be perfect!

Submission File

Gotta start somewhere

DAGMan

Automation is key

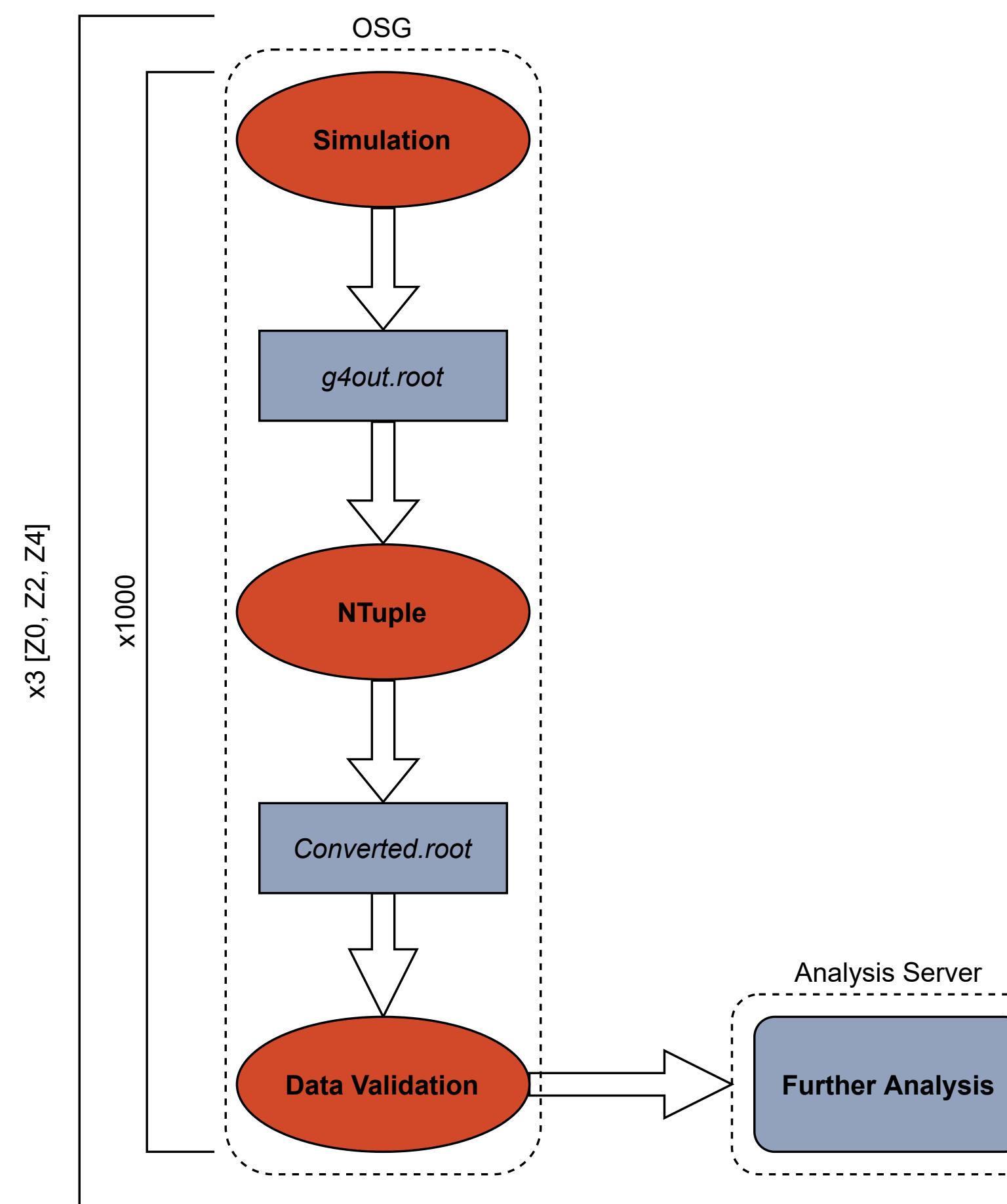
Pegasus

Proper programming makes life easy

Queued Pegasus

Embrace the laziness

One simulation of $1e9$ events broken up into 1000 simulations of $1e6$ events

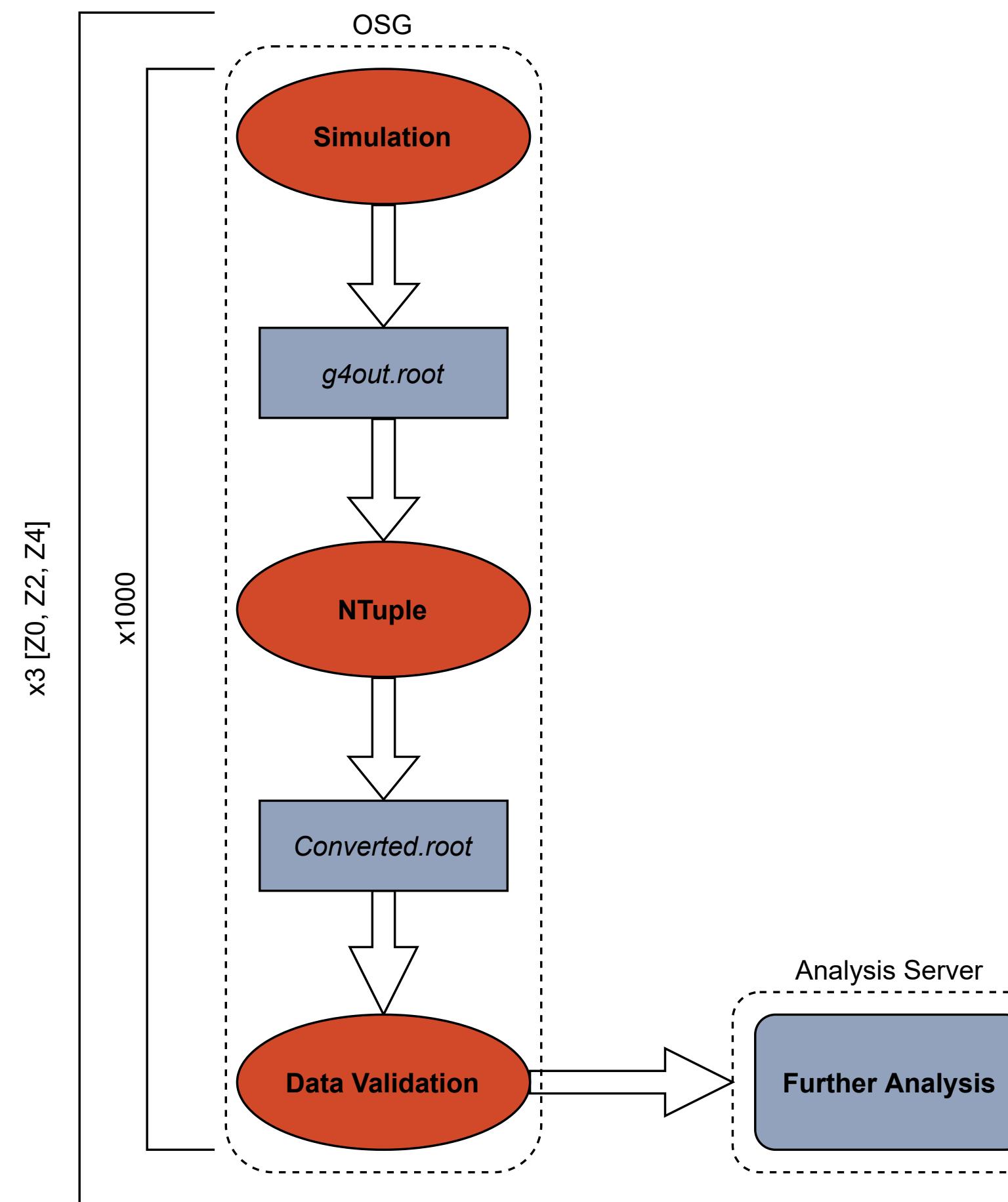


DAGMan handled workflow

Python script created DAG file

Total workflow took ~24 hours

DAGMan was good, but not perfect

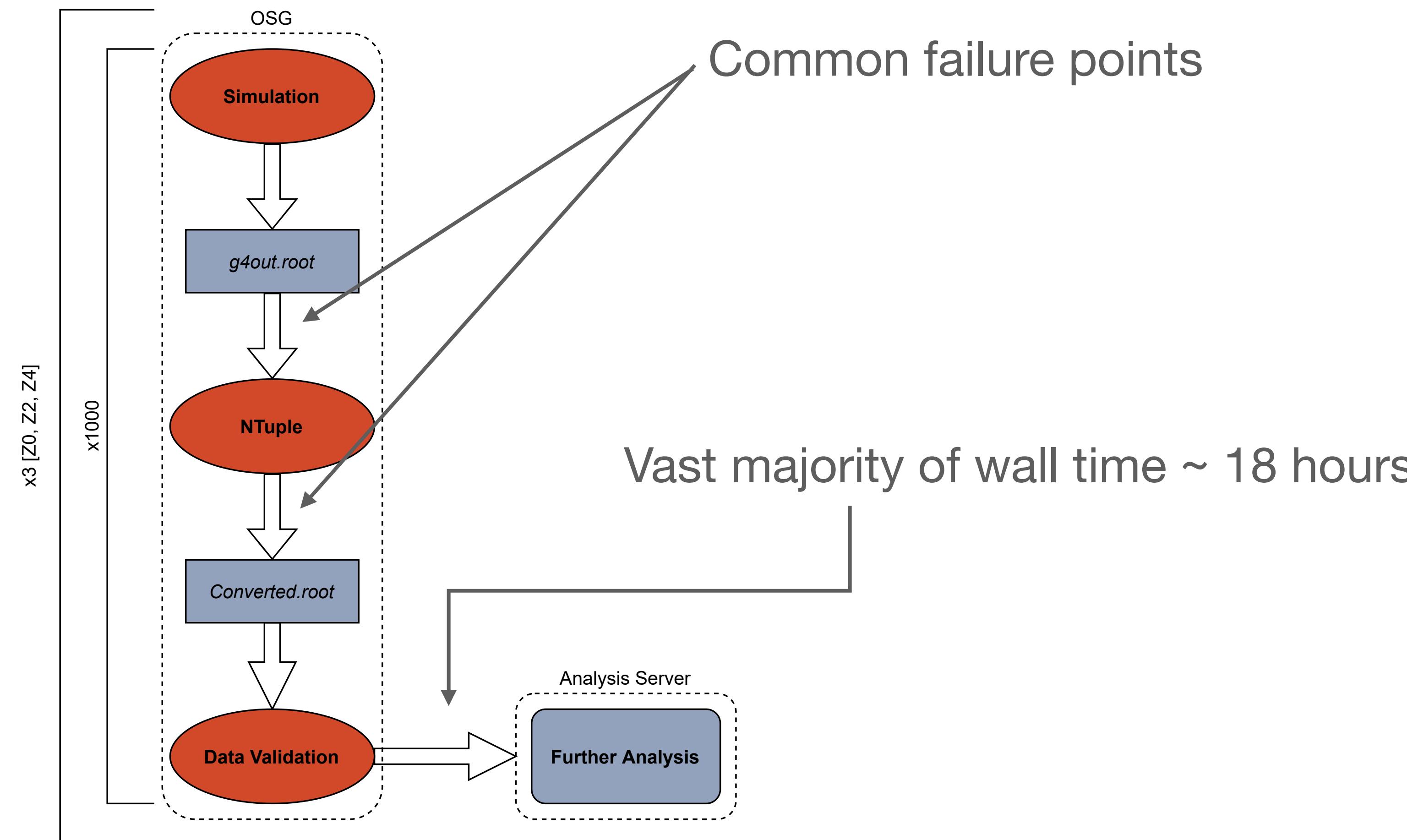


Jobs would fail randomly
File transfers, bad simulations, etc

Automation reliant on my
ability as a programmer

Large memory footprint on submit node
> 400 GB

DAGMan was good, but not perfect



Building an OSG workflow is an iterative process, and it doesn't need to be perfect!

Submission File

Gotta start somewhere

DAGMan

Automation is key

Pegasus

Proper programming makes life easy

Queued Pegasus

Embrace the laziness

Converted workflow to Pegasus for file management, transfers, and error handling

Jobs would fail randomly

Retried automatically!

Automation built in

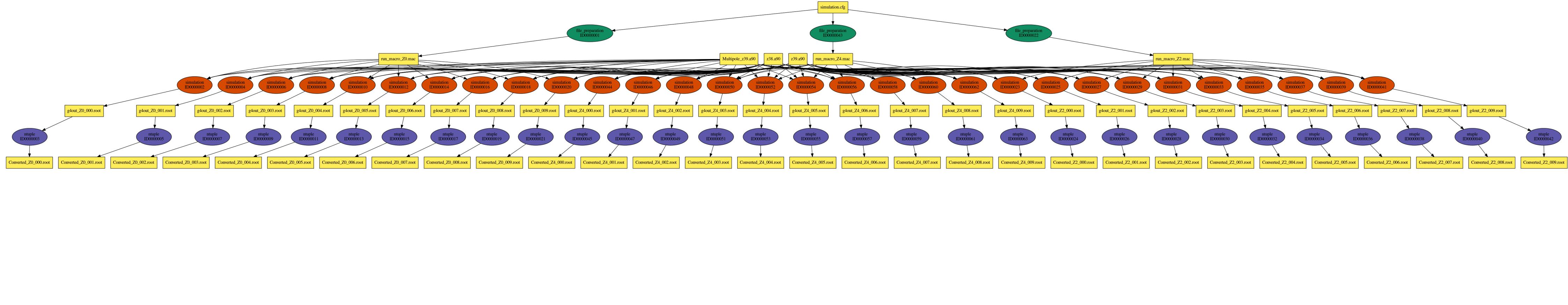
File transfers, clean up, simpler inputs, etc

Smaller memory footprint on submit node

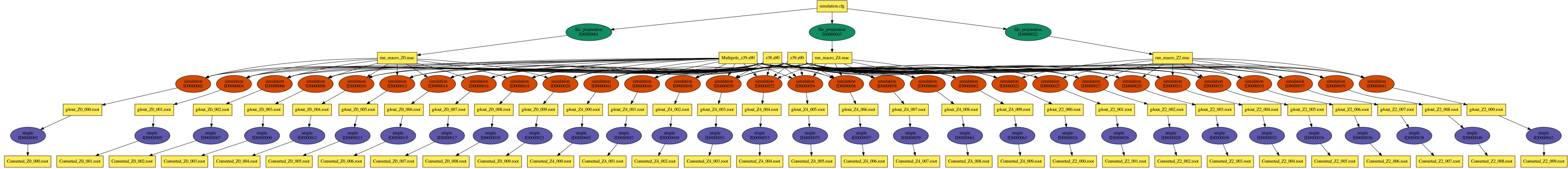
< 200 GB



Pegasus allowed for a faster and more robust workflow



Pegasus allowed for a faster and more robust workflow



Total workflow takes ~4 hours!

Using the workflow only takes 3 command line calls

vim simulation.ini

./make_input_files.sh

./ggac_surface.py

```
1 [simulation]
2 z=13
3 a=34
4 g1=1193
5 g2=2588
6 r=145
```

Using the workflow only takes 3 command line calls

vim simulation.ini

./make_input_files.sh

./ggac_surface.py



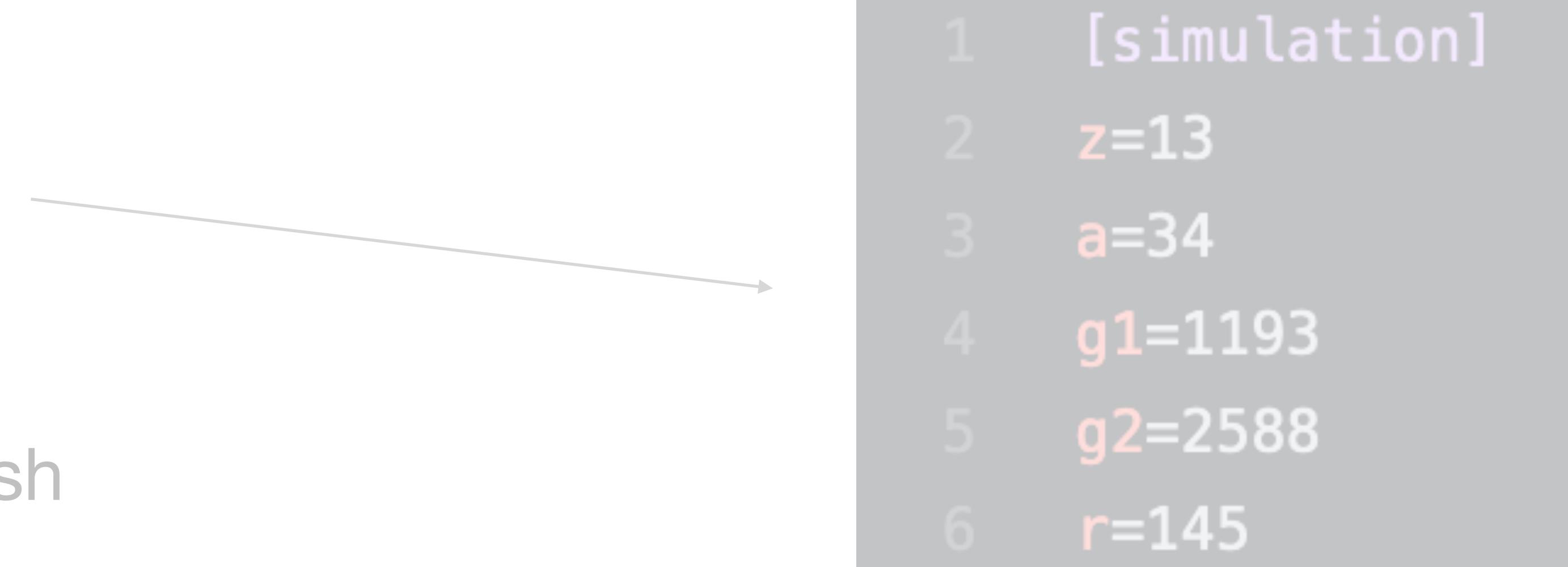
```
1 [simulation]
2 z=13
3 a=34
4 g1=1193
5 g2=2588
6 r=145
```

Using the workflow only takes 3 command line calls

vim simulation.ini

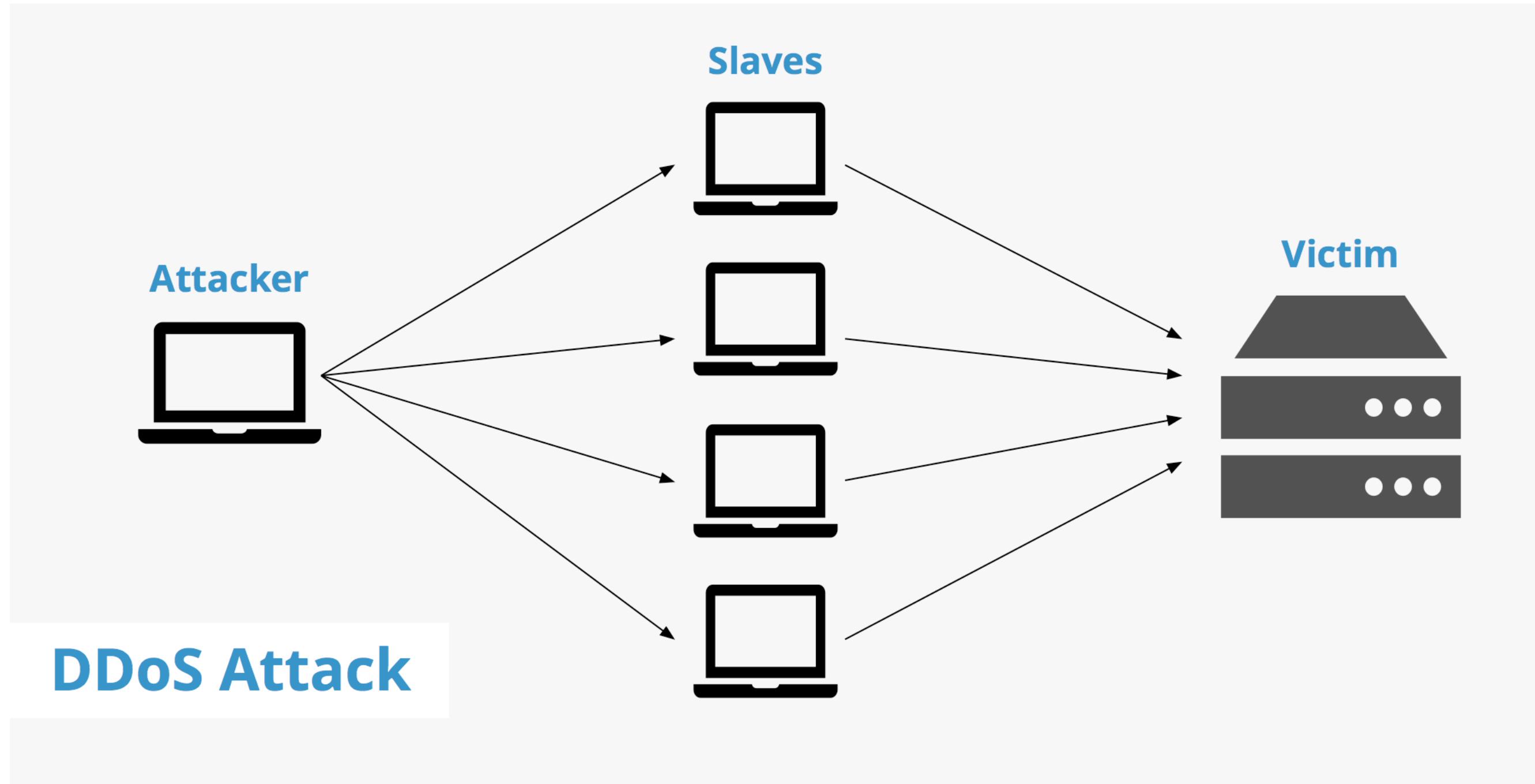
./make_input_files.sh

./ggac_surface.py



```
1 [simulation]
2 z=13
3 a=34
4 g1=1193
5 g2=2588
6 r=145
```

In large workflows Pegasus can mimic a cyberattack



Distributed Denial-of-Service
Attack (DDoS)

Flood server network interfaces
and (potentially) cause crash

The OSG has provided a more than 40x increase in simulation speed

	<u>Standard Computation</u>	<u>OSG Workflow</u>
Surface points	19	61
Wall time	168 hrs / pt	4 hrs / pt
File management	<i>Manual</i>	<i>Automatic</i>

Building an OSG workflow is an iterative process, and it doesn't need to be perfect!

Submission File

Gotta start somewhere

DAGMan

Automation is key

Pegasus

Proper programming makes life easy

Queued Pegasus

Embrace the laziness

If some automation is good, more must be better!

Automatically submit multiple workflows

Even simpler input file



```
1 z, a, total_events, events_per_sim
2 27, 60, 10e6, 2e5
3 63, 152, 10e6, 2e5
```

More can go wrong without warning

Elements != Isotopes

Words of wisdom from a graduate student

(Use at your own peril)

Don't be afraid to break things

**If you have to do it more than twice,
automate it**

The answer is *always* in the error logs

**The Job Failures
Will Continue**



Until Morale Improves

Thank you to everyone who helped develop my workflow!



Tim Cartwright
OSG



Lauren Michael
DAGMan



Mats Rynge
Pegasus



COLORADO SCHOOL OF MINES
EARTH • ENERGY • ENVIRONMENT

Connor Natzke
cnatzke@mines.edu



TRIUMF

Access to the OSG has changed how I approach expensive computational problems

Total Wall Hours:
135k hours
~15 years!

Total Jobs:
590,000

