PSE Molekulardynamik

Team D - Sheet 2 : Collision of two bodies

GoogleTest and ParticleContainer

- Very similar to JUnit
- Added to project through “FetchContent()”

- ParticleContainer has two very important functionalities
    - applyToAll
    - Pairs::iterator
- Both were tested through GoogleTest

Continuous integration

- on every push and pull request to main:
  - set up environment
  - check if our program compiles
  - we use address-sanitizer
  - we run our tests with ctest

Making force calculation more efficient

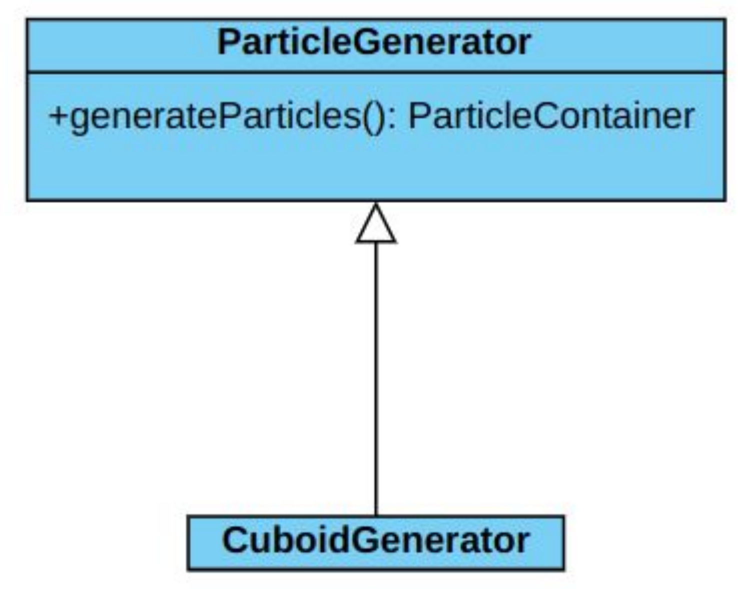
All pairs → *Unique pairs*

1 calculation / particle → *0.5 calculations / particle*

- unfortunately, the changed made the force calculation around ~30% slower

Generating a cuboid of particles

- ParticleGenerator interface to allow for future extensions
- ***output:*** ParticleContainer containing the cuboid



ParticleGenerator

+generateParticles(): ParticleContainer

CuboidGenerator

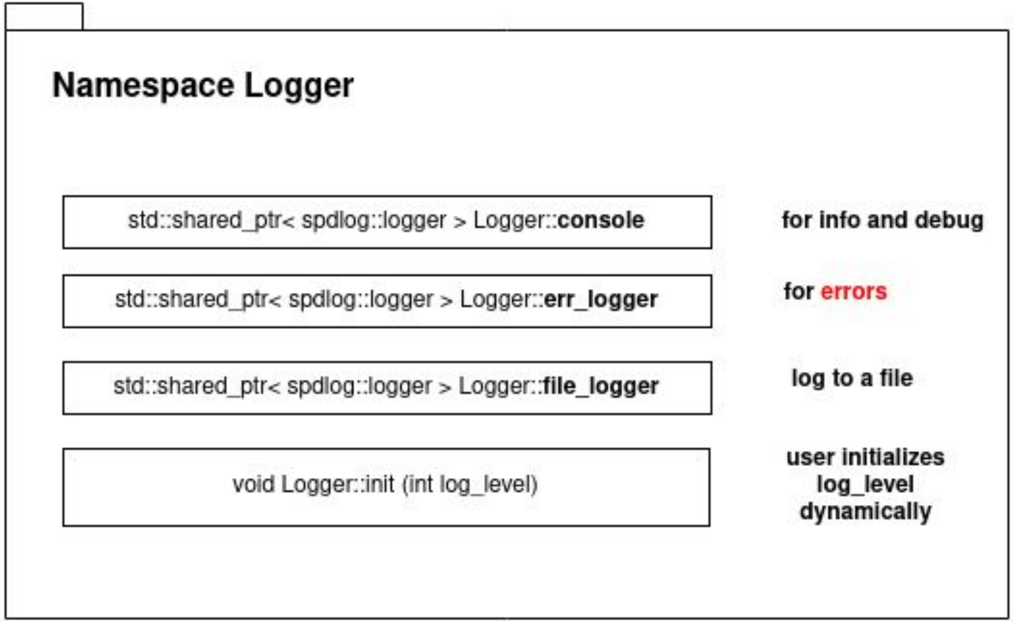
Reading in a Cuboid

- Implemented ParticleReader
- This time around, a stronger focus on input validation

- Unit tests were also essential in developing this class, through a “TDD lite” approach

Logging Made Easy: spdlog Essentials

- header-only C++ logging library →Easy integration
- choice of log functions for more flexibility and dynamic log levels
- Implementation within a namespace



Namespace Logger

std::shared_ptr< spdlog::logger > Logger::**console**

for info and debug

std::shared_ptr< spdlog::logger > Logger::**err_logger**

for errors

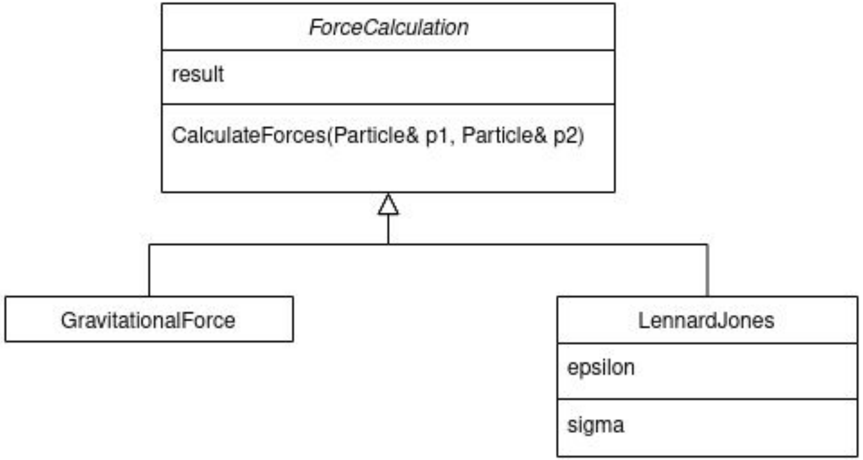
std::shared_ptr< spdlog::logger > Logger::**file_logger**

log to a file

void Logger::init (int log_level)

user initializes
log_level
dynamically

Intermolecular Interactions: Lennard-Jones Forces



Implementation of the formula: Cache repeated computations for more effeciency

Impact in Motion: Visualizing Body Collision

https://github.com/klaramozna/PSEMolDyn_GroupD/assets/101558922/7ae15dff-bda3-4fdf-904e-31e34f2324f5

[1]

Performance

- Currently, performance is not that great

At the end, we measured: 58885 milliseconds or ~58,9 seconds

- Our hypothesis for reasons are
  - Bad implementation of iterators for ParticleContainer
  - Unoptimized handling of Particle objects

But we still need to further investigate…

References

[1] https://seeklogo.com/vector-logo/297768/movie-time-cinema