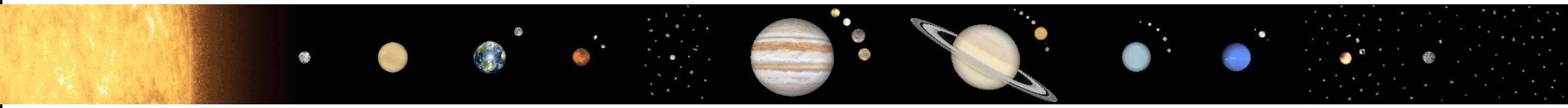
PSE Molekulardynamik
Sheet 1: First steps towards a molecular dynamics' simulation



Group A: Daniel Schade, Ashutosh Solanki, Robin Cleve

03.05.2024

Task 1 & 2: Set up

- ParaView (5.9)
- CMake (3.27.4)
- Doxygen (1.10.0)
- Clang (16.0.6)
- Make (4.3)
- Graphviz (2.42.2)
- Clang tidy

Task 3: Completion of program frame

- Implementation of **Force calculation** was straight forward
  - used L2 Norm from ArrayUtils.h
- Checked implementation by **running a simulation** in ParaView
  - adjusted output file format from .xyt to .vtu
- Boost for **argument parsing** in the command line

Task 3: Completion of program frame

$$x_i(t_{n+1}) = x_i(t_n) + \Delta t \cdot v_i(t_n) + (\Delta t)^2 \frac{F_i(t_n)}{2m_i}$$

```
void Simulator::calculateX() {
    for (auto &p : particles) {
        p.setX(p.getX() + deltaT * p.getV() + ((deltaT * deltaT) / (2.0 * p.getM())) * p.getOldF());
    }
}
```

$$v_i(t_{n+1}) = v_i(t_n) + \Delta t \frac{F_i(t_n) + F_i(t_{n+1})}{2m_i}$$

```
void Simulator::calculateV() {
    for (auto &p : particles) {
        p.setV(p.getV() + (deltaT / (2 * p.getM())) * (p.getOldF() + p.getF()));
    }
}
```

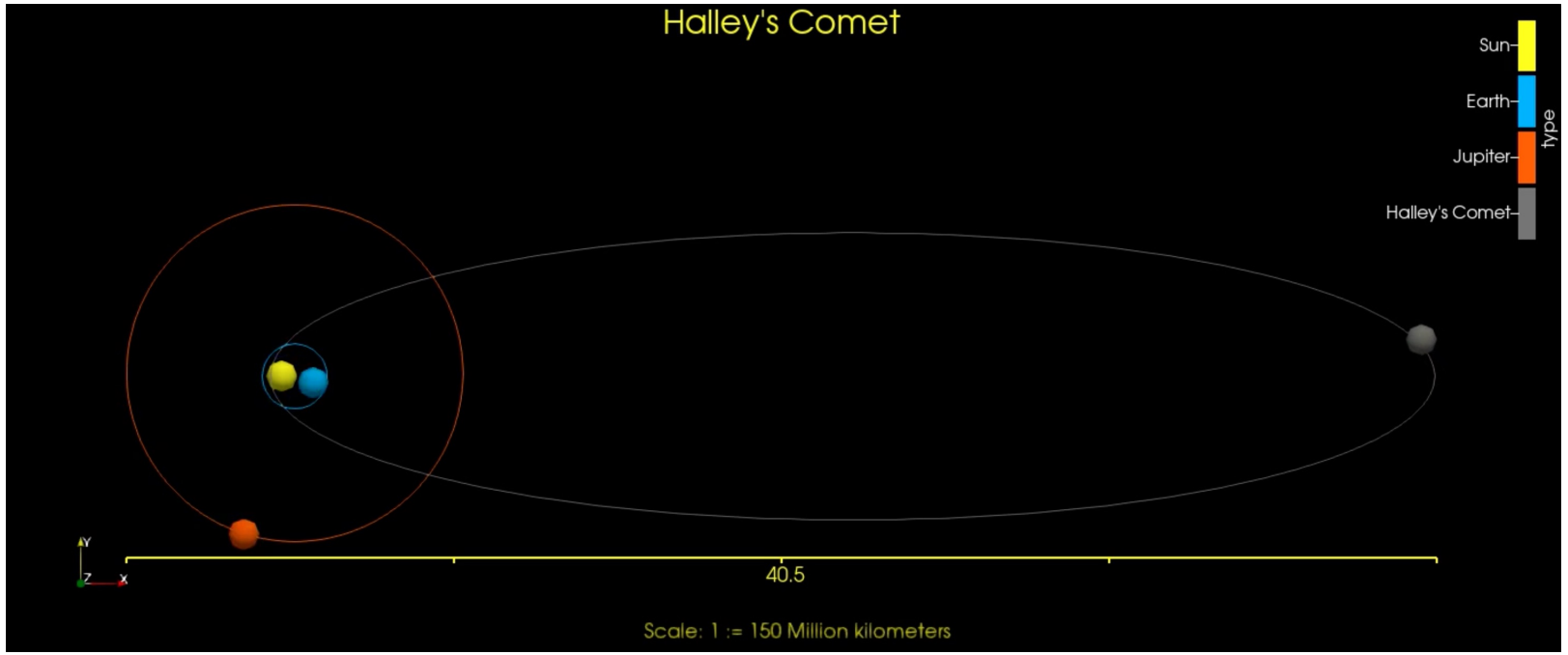
$$F_{ij} = \frac{m_i m_j}{\left(\|\mathbf{x}_i - \mathbf{x}_j\|_2\right)^3} (\mathbf{x}_j - \mathbf{x}_i)$$

```
std::array<double, 3> Gravity::compute(Particle &target, Particle &source) {
    return (target.getM() * source.getM()) / std::pow(ArrayUtils::L2Norm(target.getX() - source.getX()), 3.0) *
    (source.getX() - target.getX());
}
```

Task 4: Halley’s Comet

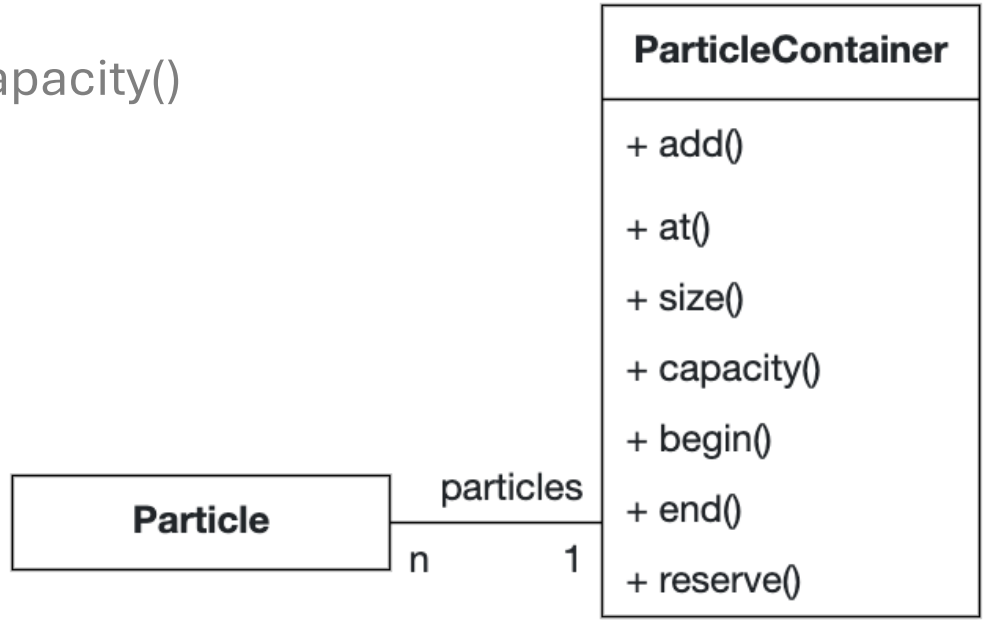
Identification of celestial bodies:

- **Sun:** most mass
- **Comet:** least mass, unconventional trajectory
- **Earth and Jupiter:** earth is closer to sun than Jupiter



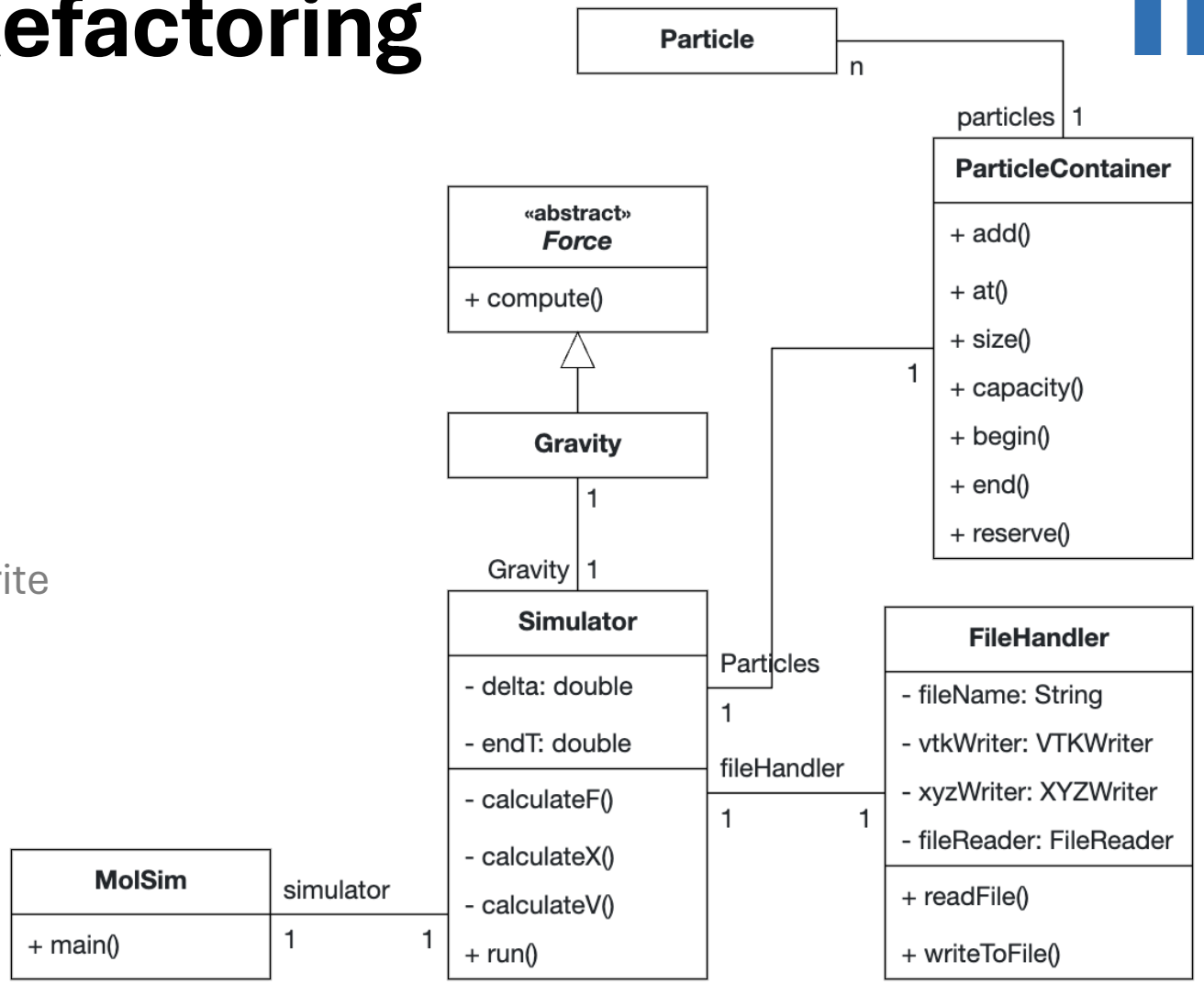
Task 5: Particle Container

- Store particles using **std::vector**
  - objects stored **consecutively** for better **performance**
  - but still **dynamic** on the other hand
  - Implementation of add(), at(), size(), capacity()
- **Iterator** pattern
  - Iterate over pairs with **nested for loop**
  - Implementation of begin(), end()



Task 5: Further Refactoring

- **Strategy pattern** for forces acting on Particles
  - abstract class Force.h as **parent** for all forces to come
  - **gravity** being the first (implementation of task 3)
- Interface for **file handling**
  - **FileHandler** encapsulates write and read classes
- **Simulator** class instead of comp-lex main method



Thank you for listening!