DPD: A Java implementation

Jo Hoffmann

SSP 2001

DPD: Overview

- Why is this project interesting?
- What is DPD about?
- A DPD model for OO
- Problems with Java
- Done and to be done

DPD: Background

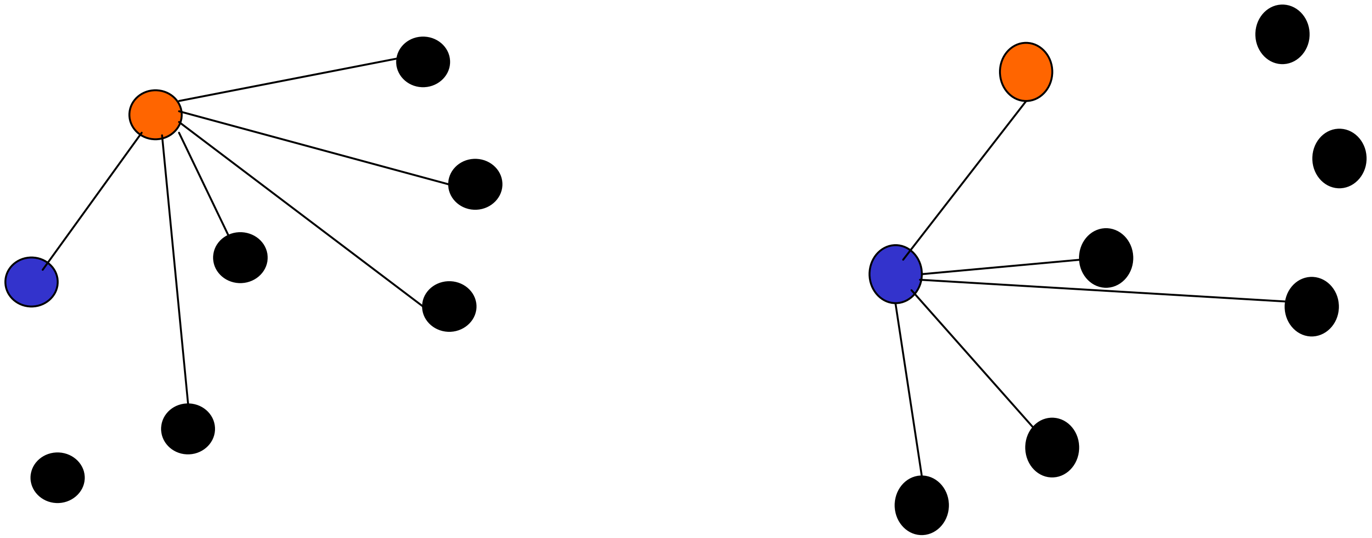
- Original Fortran code developed by Mike Cates, parallelised with MPI
- Original code difficult to modify
- Interest in a GUI
- Interest in a Shared Memory System
- so rewrite in Java with OpenMP or Threads

DPD: The physics

- Dissipative Particle Dynamics (DPD)
  - like Molecular Dynamics (MD)
  - System defined by
    - position of all particles and their
    - momenta
  - Simulation involves 2 steps
    - force calculation
    - velocity and position update

DPD: The Physics

- Force calculation
  - cut-off-distance
- overall n(n-1) steps



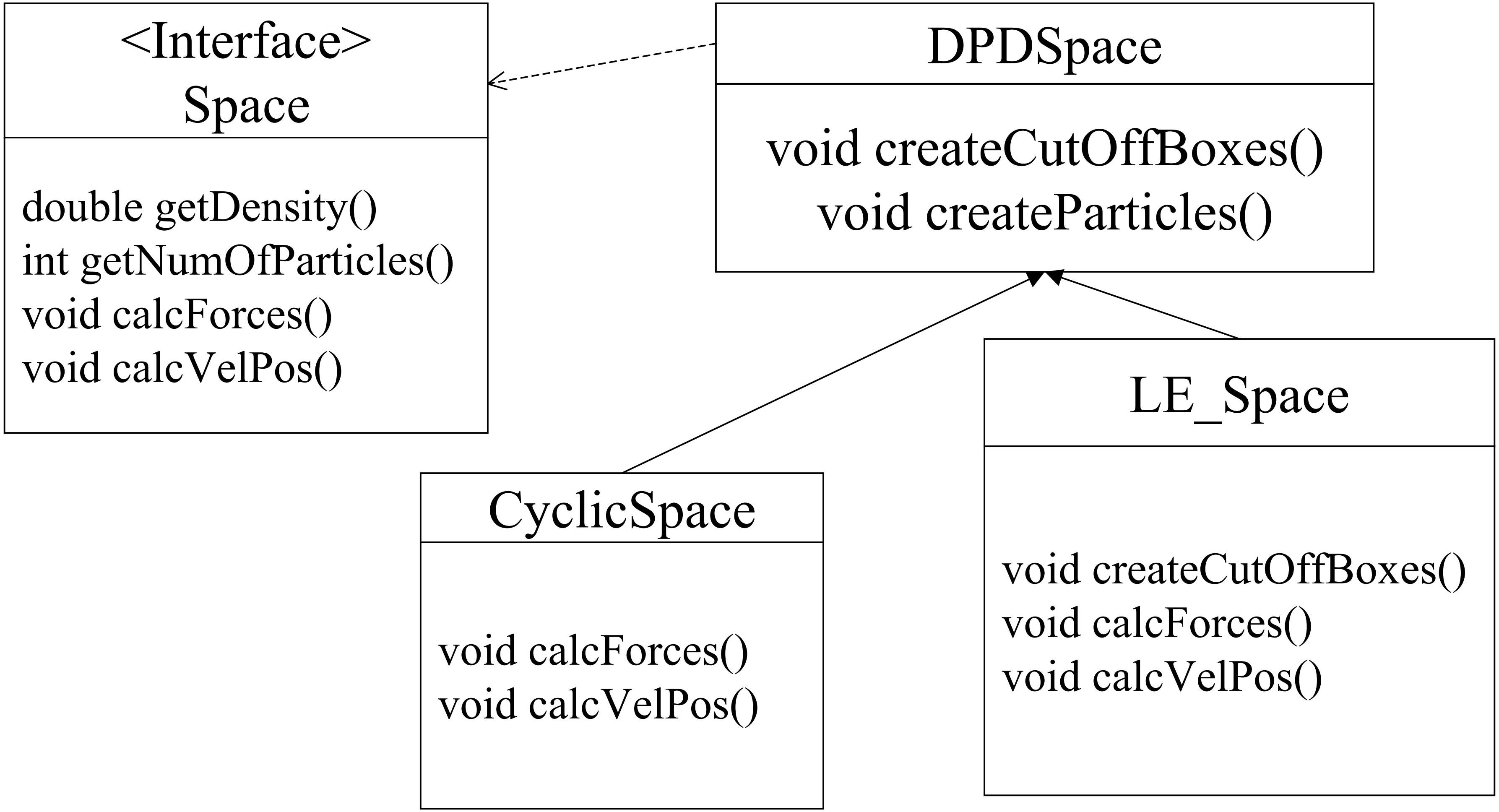
DPD: The development process

- Identifying objects
- Modelling the classes with UML
- Do a bit of coding
- Bumping into trouble
- Do more modelling
- Eventually get it working

DPD: Identifying objects

- Particles: Monomers, Dimers, Colloids
  - Type
- Space: cyclic boundary, Lees Edwards boundary
- CutOffBoxes
- Input, Statistics and other helper classes

DPD: The Space hierarchy



<Interface>
Space

double getDensity()
int getNumOfParticles()
void calcForces()
void calcVelPos()

DPDSpace

void createCutOffBoxes()
void createParticles()

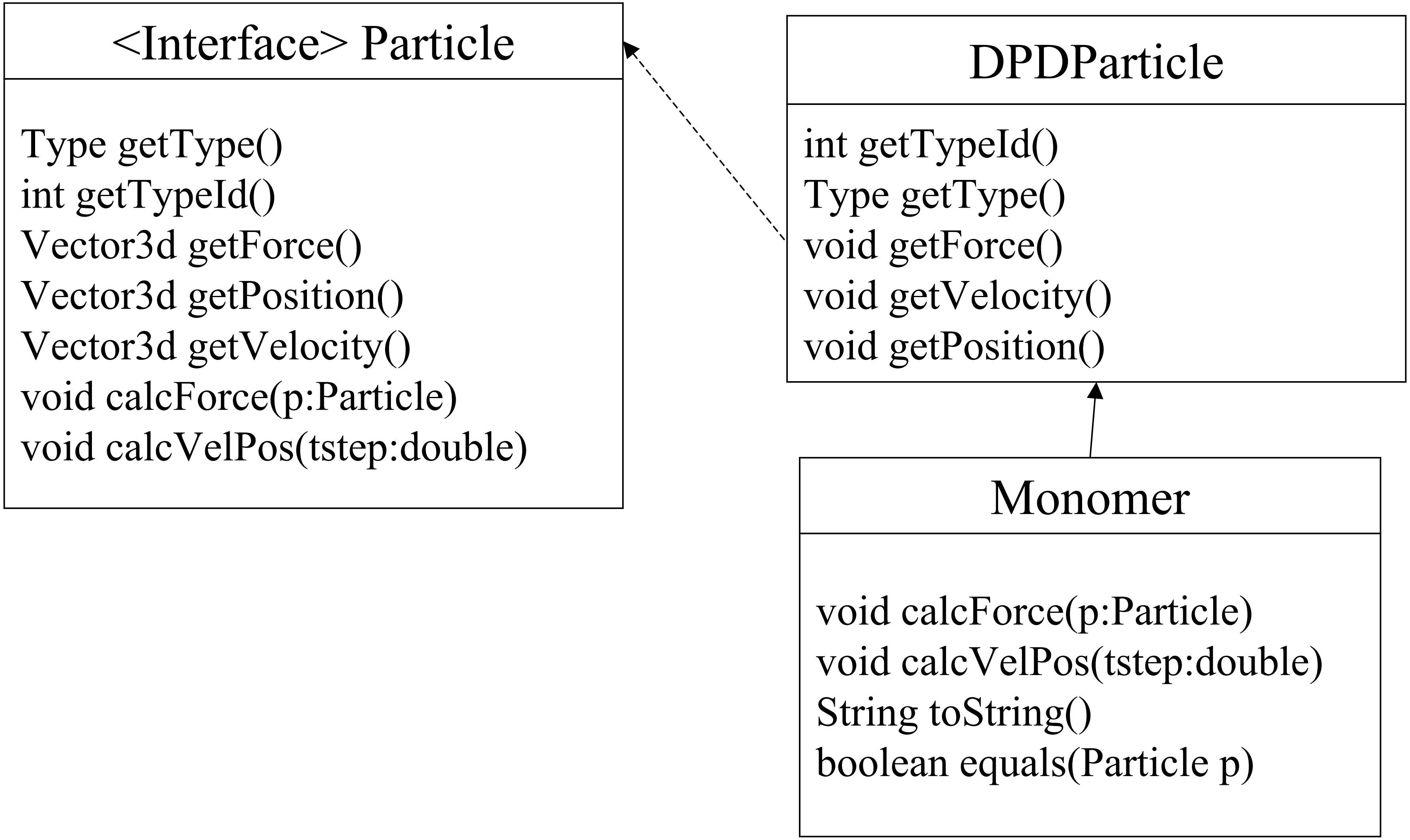
CyclicSpace

void calcForces()
void calcVelPos()

LE_Space

void createCutOffBoxes()
void calcForces()
void calcVelPos()

DPD: The particle hierarchy 1



<Interface> Particle

Type getType()
int getTypeId()
Vector3d getForce()
Vector3d getPosition()
Vector3d getVelocity()
void calcForce(p:Particle)
void calcVelPos(tstep:double)

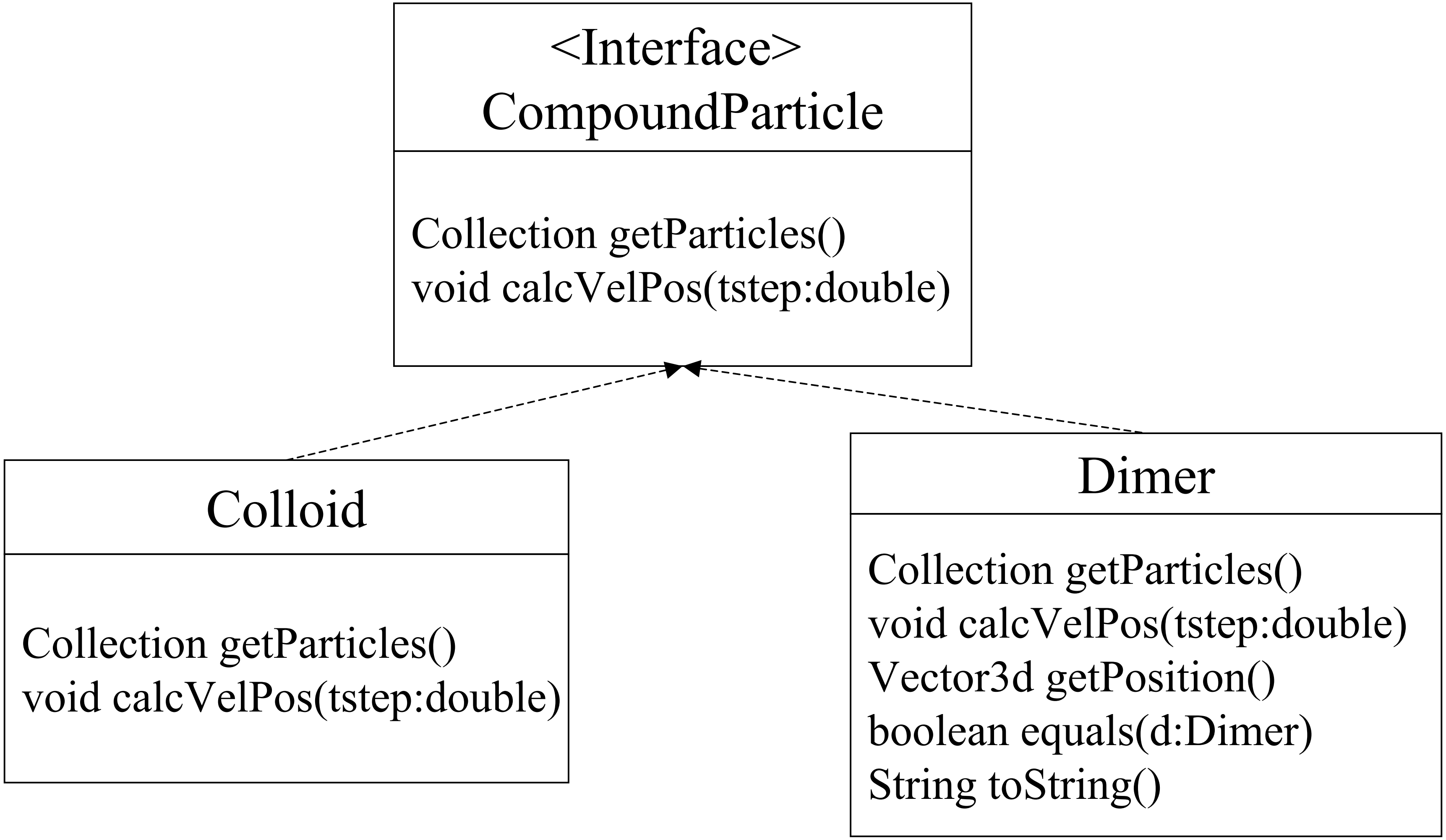
DPDParticle

int getTypeId()
Type getType()
void getForce()
void getVelocity()
void getPosition()

Monomer

void calcForce(p:Particle)
void calcVelPos(tstep:double)
String toString()
boolean equals(Particle p)

DPD: The particle hierarchy 2



<Interface>
CompoundParticle

Collection getParticles()
void calcVelPos(tstep:double)

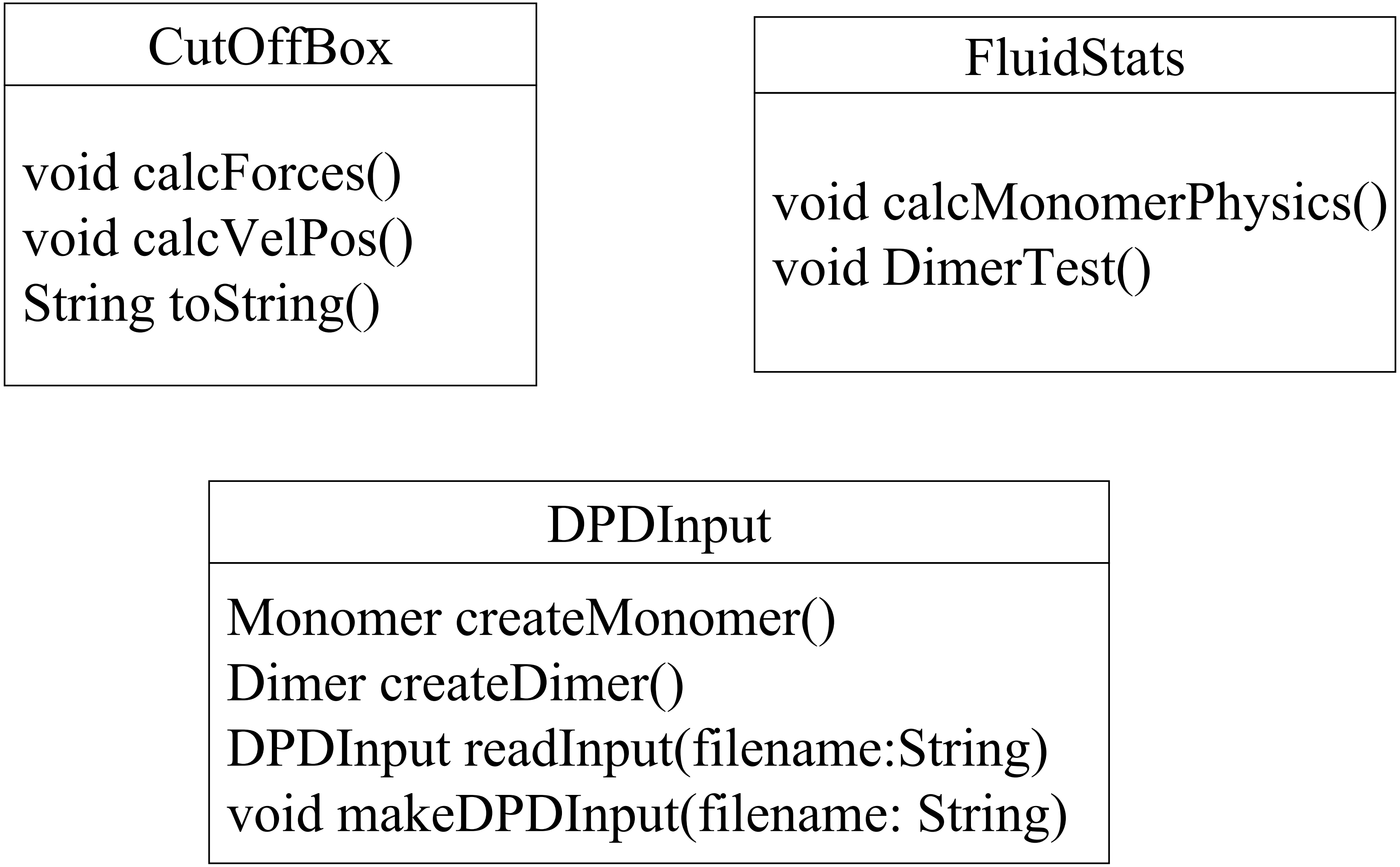
Colloid

Collection getParticles()
void calcVelPos(tstep:double)

Dimer

Collection getParticles()
void calcVelPos(tstep:double)
Vector3d getPosition()
boolean equals(d:Dimer)
String toString()

DPD: Other classes



CutOffBox

void calcForces()
void calcVelPos()
String toString()

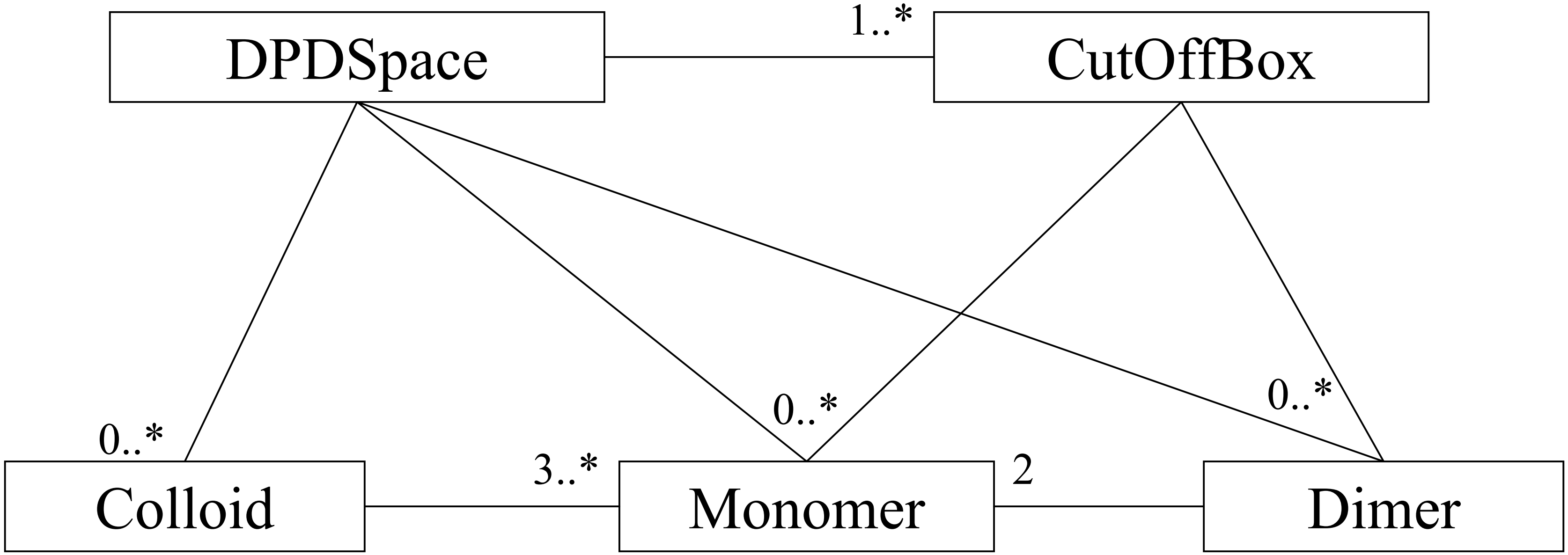
FluidStats

void calcMonomerPhysics()
void DimerTest()

DPDInput

Monomer createMonomer()
Dimer createDimer()
DPDInput readInput(filename:String)
void makeDPDInput(filename: String)

DPD: Relationships



DPD: Testing the code

- Individual method tests
- Integration tests
- Loss and creation of particles during simulation
- Energy conservation
- Dimer constraints
- Dimer allignment

DPD: Problematic Java

- Algorithms spread out
- Global variables
- Performance issues
  - Object creation; destruction; garbage collection
  - Iterators
  - Casting

DPD: Status

- Done so far
  - CyclicSpace, LE_Space
  - Monomers, Dimers, Input, Statistics
- To be done
  - make the Input class and Simulation Space setup more general
  - GUI
  - parallelisation with Open/MP or Java Threads