

Objects as compounds of constrained particles: advantages



- Interesting/rich/useful set of “emerging behaviors” (i.e. effects with “just automatically happens”) :
 - **rigid, deformable, jointed objects**
 - made of particles + hard constraints
 - their **angular velocities**
 - rotation around proper **axis**
 - their **barycenter**
 - their **momentum of inertia**
 - angular velocity is maintained
 - somewhat believable **bounces on “impacts”**
 - but, out of designer control: **impact impulses** can be added

don't need to be computed (or stored)

consequence of constraints disallowing penetration

Compounds of particles disguised as rigid bodies



Particles compounds or rigid bodies?



- Rigid-body based systems:
 - explicitly compute dynamics for rigid bodies
 - updating their rotation, angular speed,...
- Particles-based systems:
 - only compute dynamics for particles
 - rigid (or deformable, or jointed) bodies as an emerging behavior
- Mixed:
 - use both
 - bonus: dynamically swap between the two representations for rigid bodies

Particle-based systems: Challenges



- Approximations introduced
 - e.g.: mass concentrated in a few locations
- Scalability issues
 - many constraints to enforce, many particles to track
- Some of the data which is kept implicit is needed by the rest of the engine
 - and therefore must be extracted ☹
 - e.g.: (rotation) of a the “rigid body”
 - (needed for rendering!)
 - and: its angular speed, barycenter pos, velocity...

