BOIDS + PHYSICS + VECTORS + PYGAME = AWESOME!!

References:

- Wikipedia ☺

PYGAME + VECTORS

- Examples:
  - [Draw a circle at a math3d.VectorN as position]
  - [Make a velocity vector – use dt!]
  - [Make character track mouse]
  - [“Oriented points”]

WORLD'S SHORTEST EXPLANATION OF ACCELERATION

- Position update (from the example)

$$\overrightarrow{newPos} = \overrightarrow{oldPos} + \overrightarrow{vel} * \Delta t$$

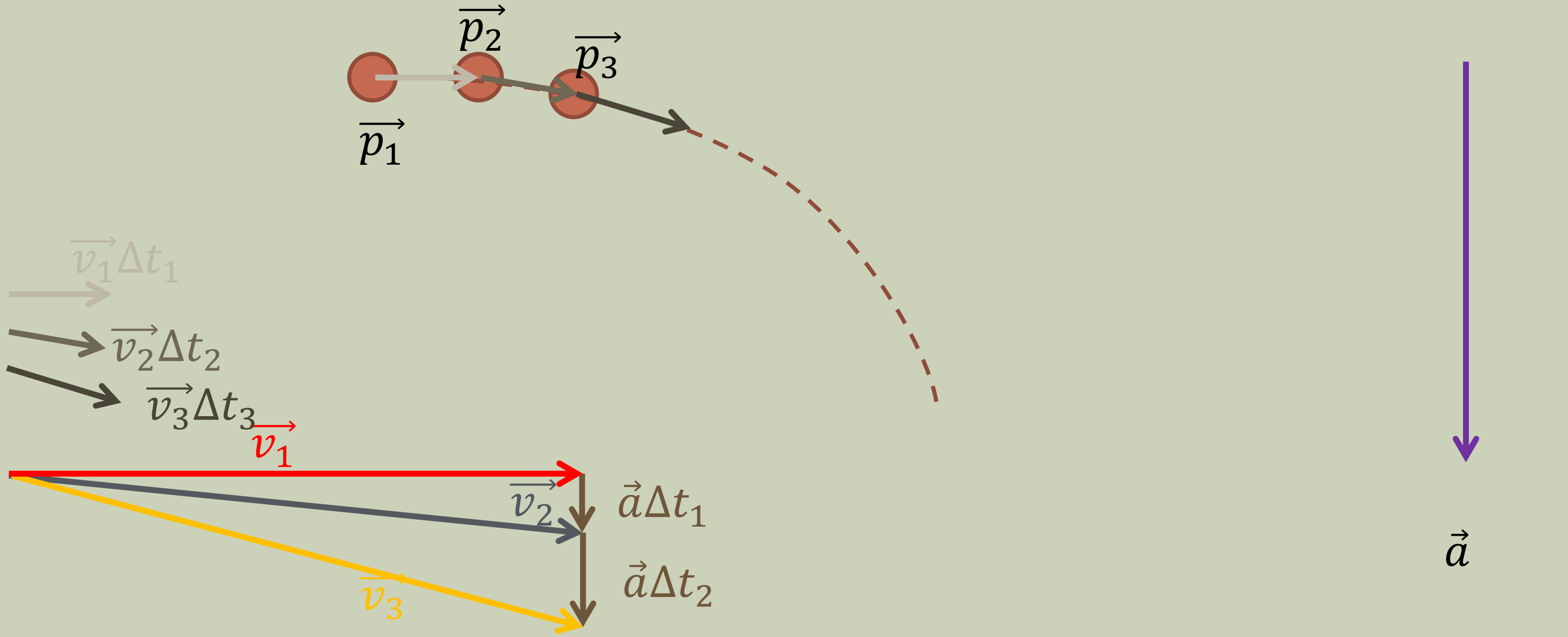
- Acceleration
  - Gradually increasing velocity
  - Mimics real-life objects (can't start / stop on a dime)
  - Velocity update:

$$\overrightarrow{newVel} = \overrightarrow{oldVel} + \overrightarrow{acceleration} * \Delta t$$

  - Just do this (and the position update) once per frame.
- Side-note:
  - This update method is called Newton-Euler1
    - It's actually the worst!
    - Doesn't handle variable frame rates well
  - Better solution (but more complex)
    - Runga-Kutta
    - …

GRAVITY

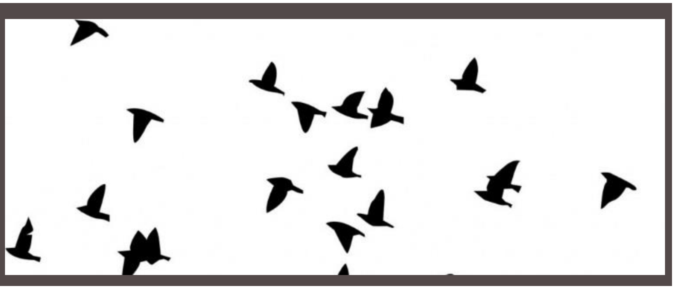
- Gravity produces an acceleration on objects
- Assuming we're on the earth, it points downwards



TERMINAL VELOCITY

- In real life, there's a maximum speed objects can go
  - Theoretically the speed of light.
  - On earth, there's wind resistance.
- In games, we often impose a **terminal velocity**
  - More numerically stable.
  - More consistent gameplay experience.
  - A little more like real life.
- [Develop the formula]

BOIDS



- Developed by Craig Reynolds (Sony?) ca 1986
- The “Boston-y” way of saying birds☺
- Flocks of birds, fish, zombies, t-rex, etc.
  - Emergent Behavior
- We apply various accelerations to an individual boid
  - Seek-center
  - Align
  - Seek-mouse
  - …
- Weighting of acceleration vectors