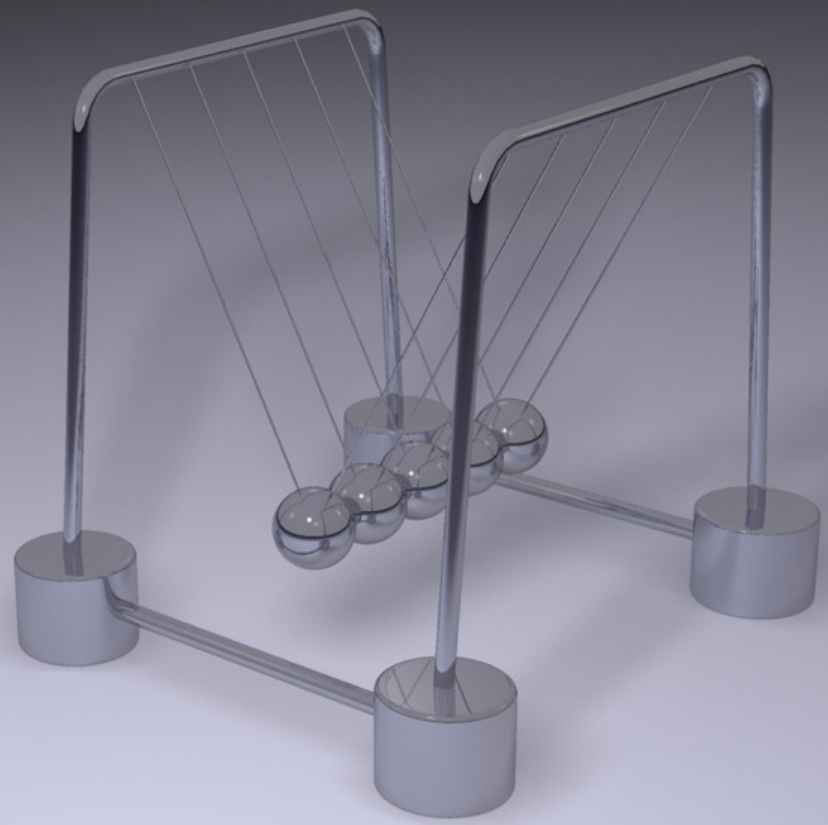
DM2212 Programming Physics

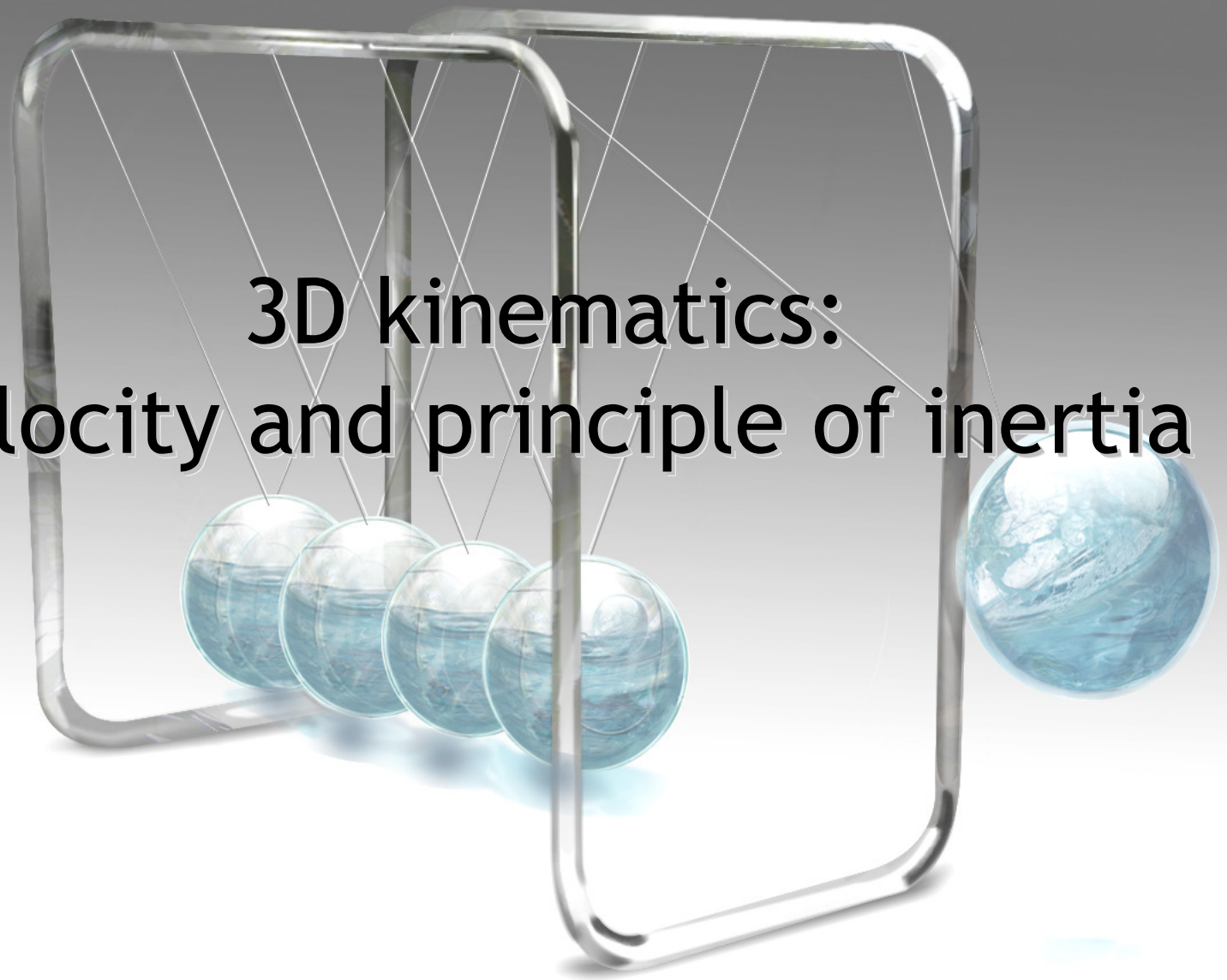


ML: Dioselin Gonzalez

2007 S1

Image by mindphase.deviantart.com

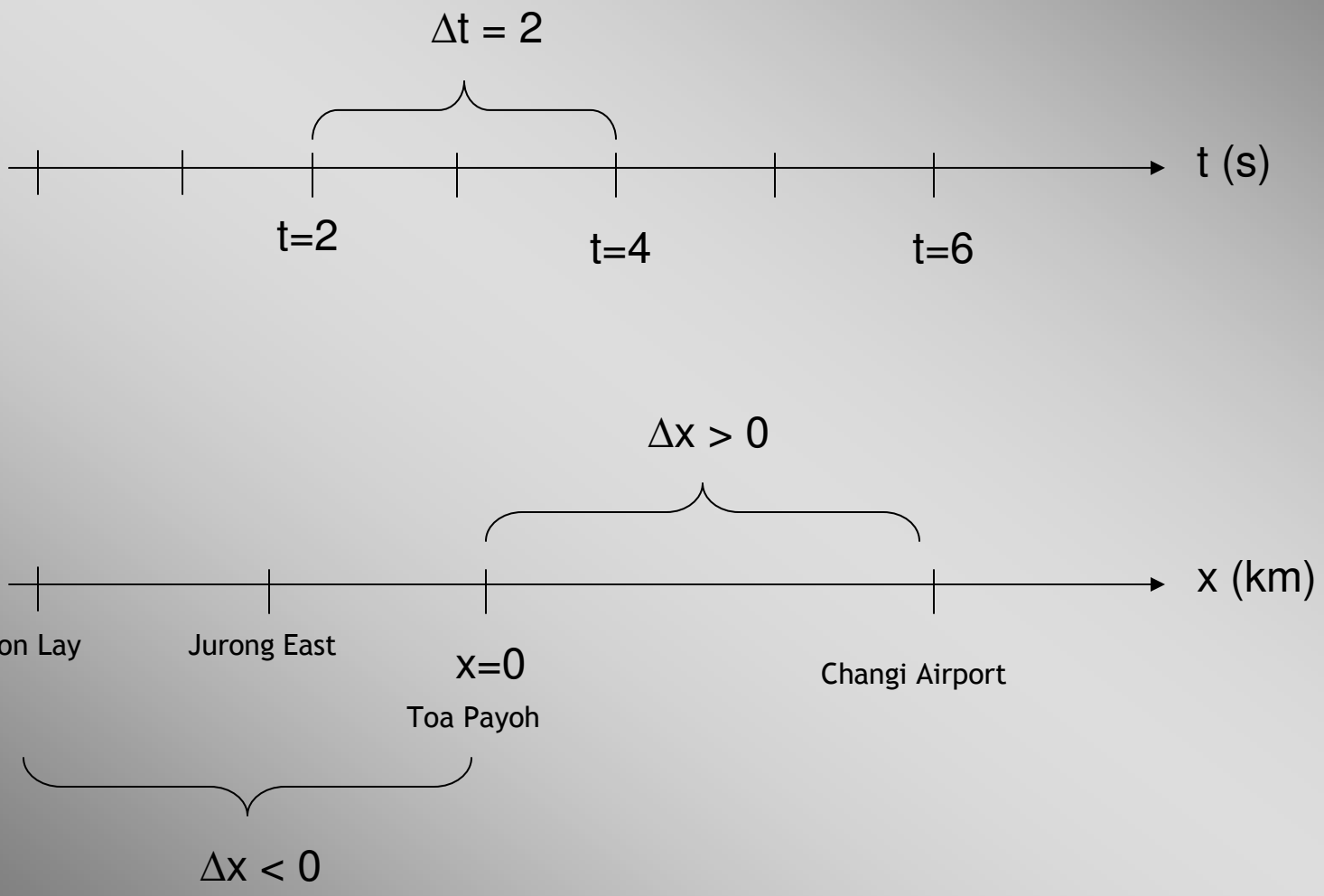
3D kinematics: velocity and principle of inertia



DM2212 - 2007S1 - Dioselin Gonzalez

Image by chocoloco.deviantart.com

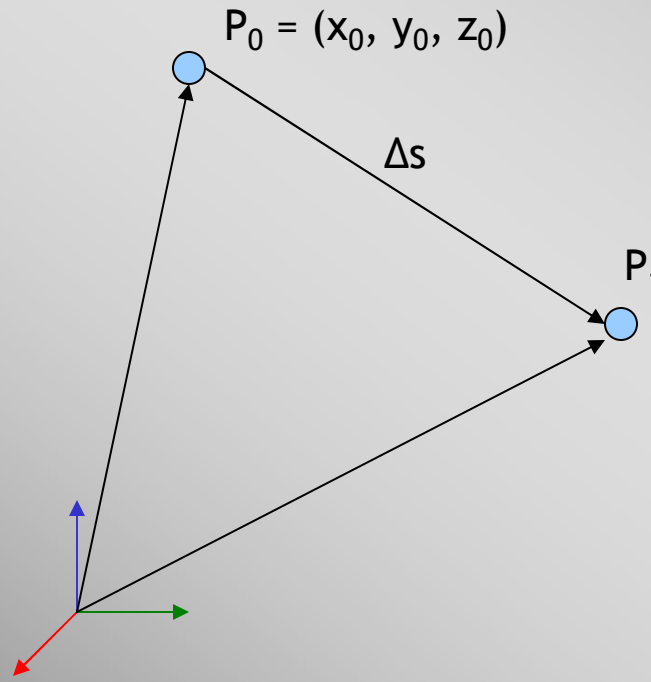
Time, position and change



Time, position and change

- A ball hits the floor, bounces to a height of one meter, falls, and hits the floor again. Is the Δx between the two impacts equal to zero, one, or two meters?

Displacement in 3D



$$\Delta s = P_1 - P_0$$

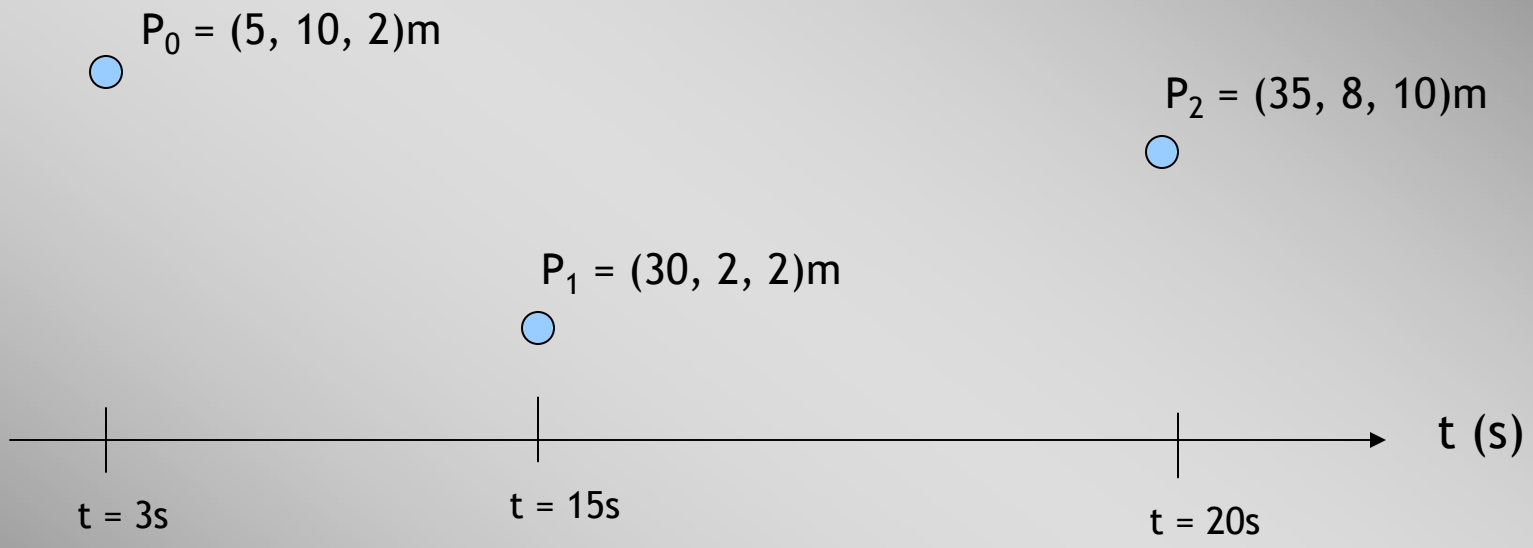
Velocity

$\mathbf{v} = \frac{\Delta s}{\Delta t}$ average velocity over time interval Δt

$|\mathbf{v}|$ = speed

$\mathbf{v} = (P_1 - P_0) \ / \ (t_1 - t_0)$

Velocity

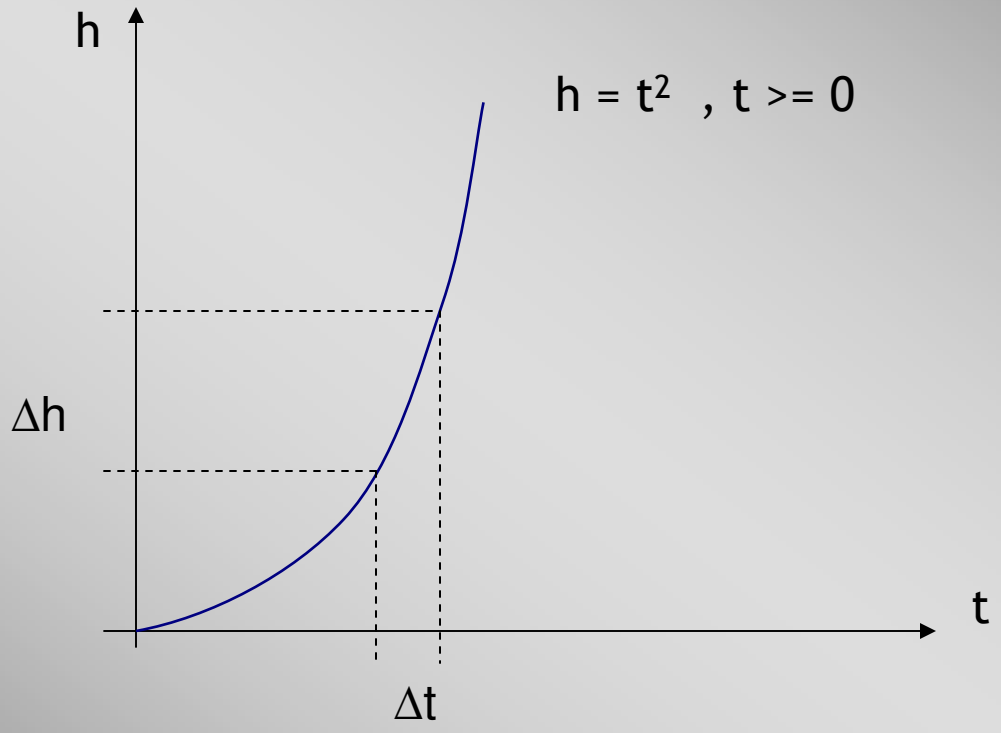


Average speed: from P_0 to P_1? From P_0 to P_2?

Total distance traveled From P_0 to P_2?

In general, displacement ≠ total distance traveled !!!

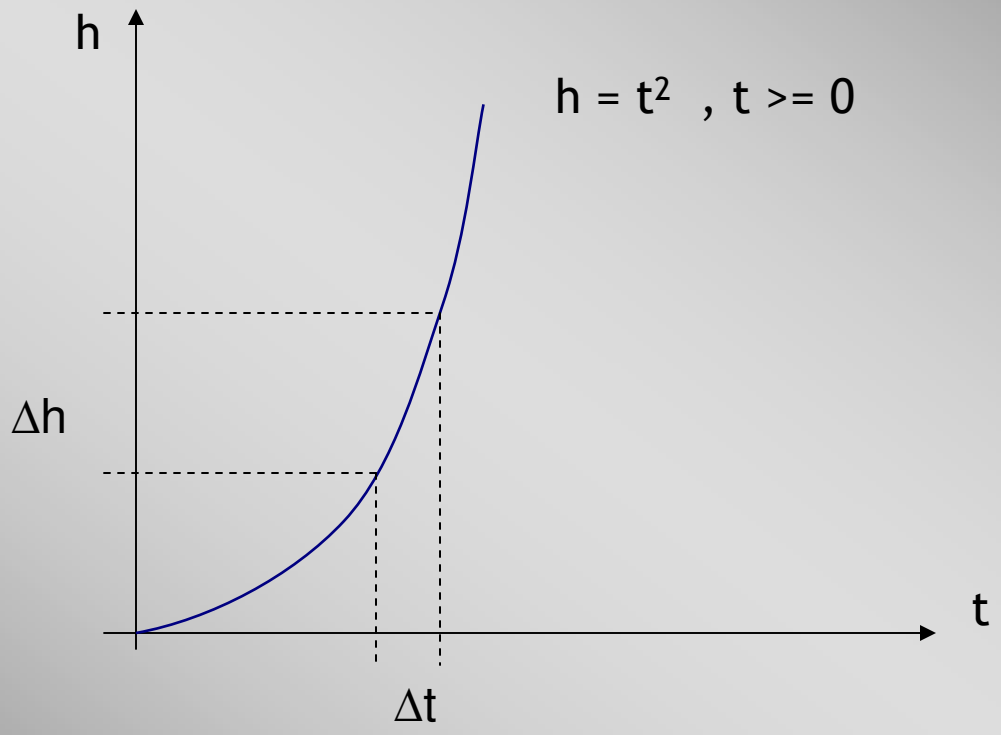
Motion graph (1D)



$$\mathbf{v} = \frac{\Delta h}{\Delta t} = \frac{h_1 - h_0}{t_1 - t_0}$$

average velocity is the slope

Motion graph (1D)

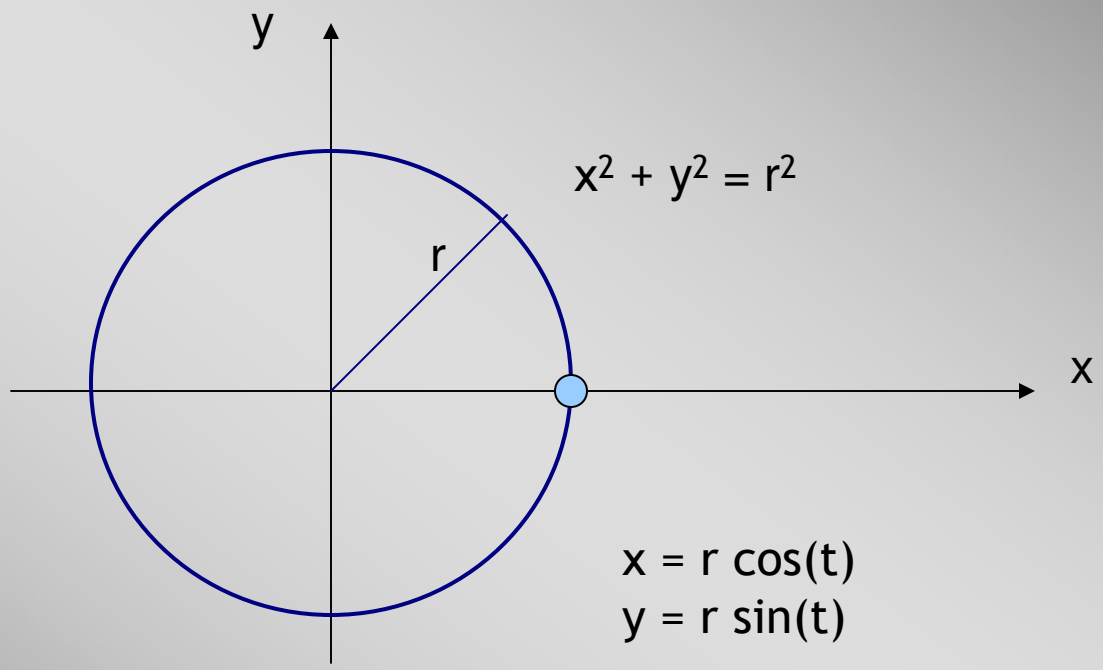


$\mathbf{v} = \lim_{\Delta t \to 0} (\Delta h / \Delta t)$

$\mathbf{v} = dh / dt$

instantaneous velocity

Motion graph (2D)



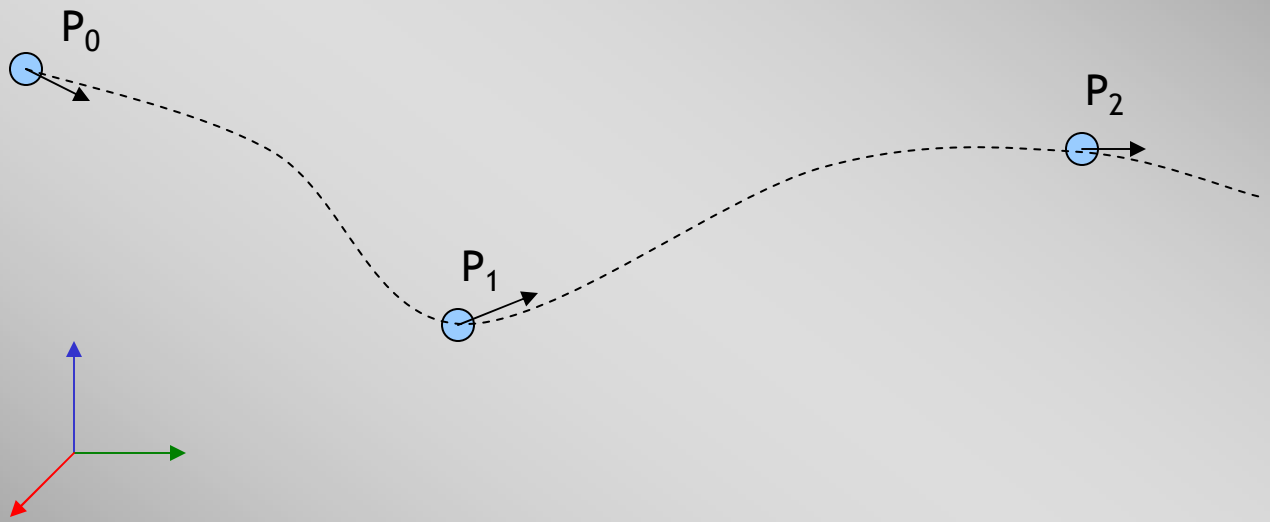
$x^2 + y^2 = r^2$

$x = r \cos(t)$

$y = r \sin(t)$

$$\mathbf{v} = ds / dt = \begin{bmatrix} dx/dt \\ dy/dt \end{bmatrix} = ?$$

Motion graph (3D)



$$\mathbf{v} = ds / dt = \begin{pmatrix} dx/dt \\ dy/dt \\ dz/dt \end{pmatrix}$$

Galileo's principle of inertia

A body moving on a level surface will continue in the same direction at constant speed unless disturbed.

http://musr.physics.ubc.ca/~jess/hr/skept/Kinem/node5.html

Programming constant velocity

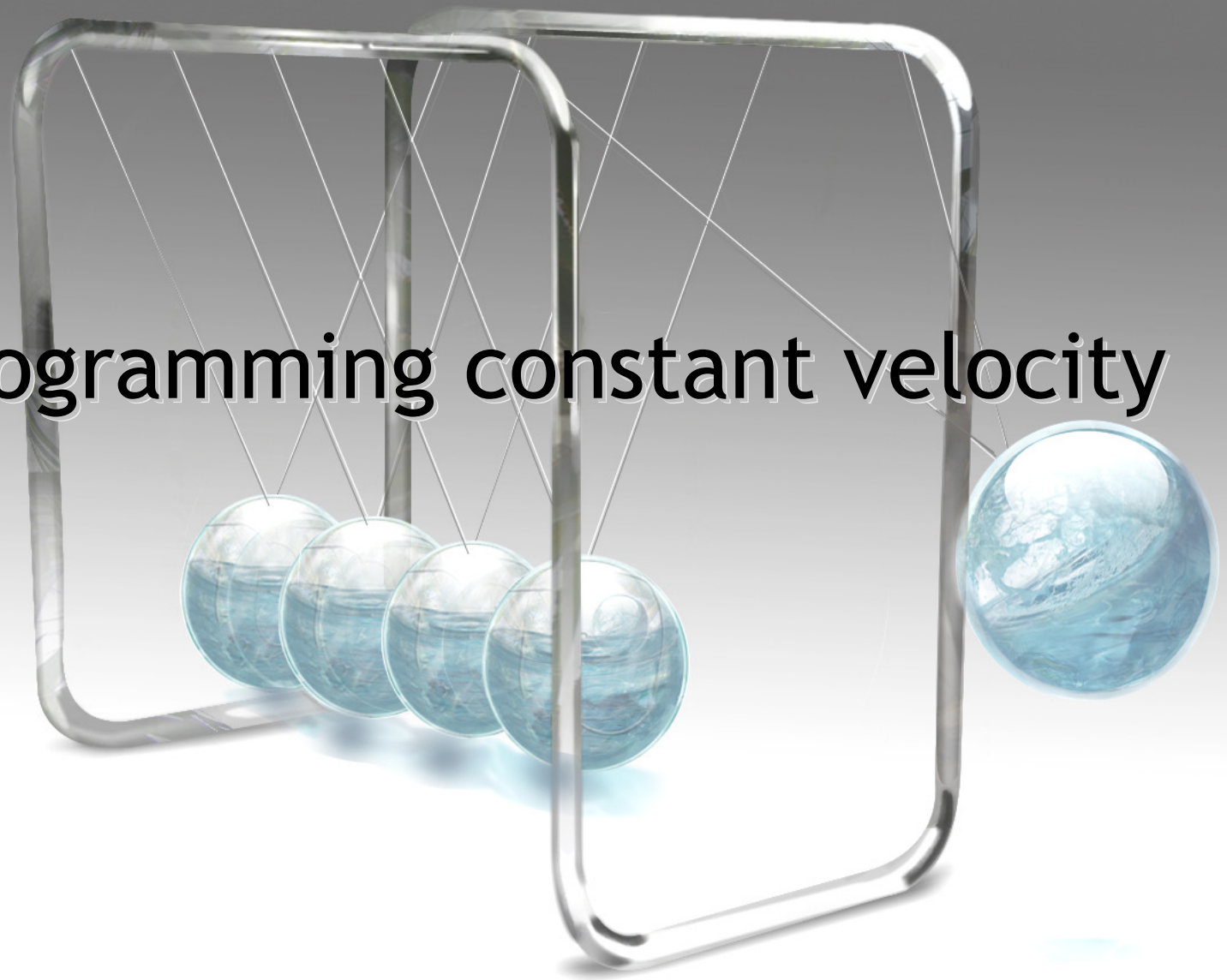
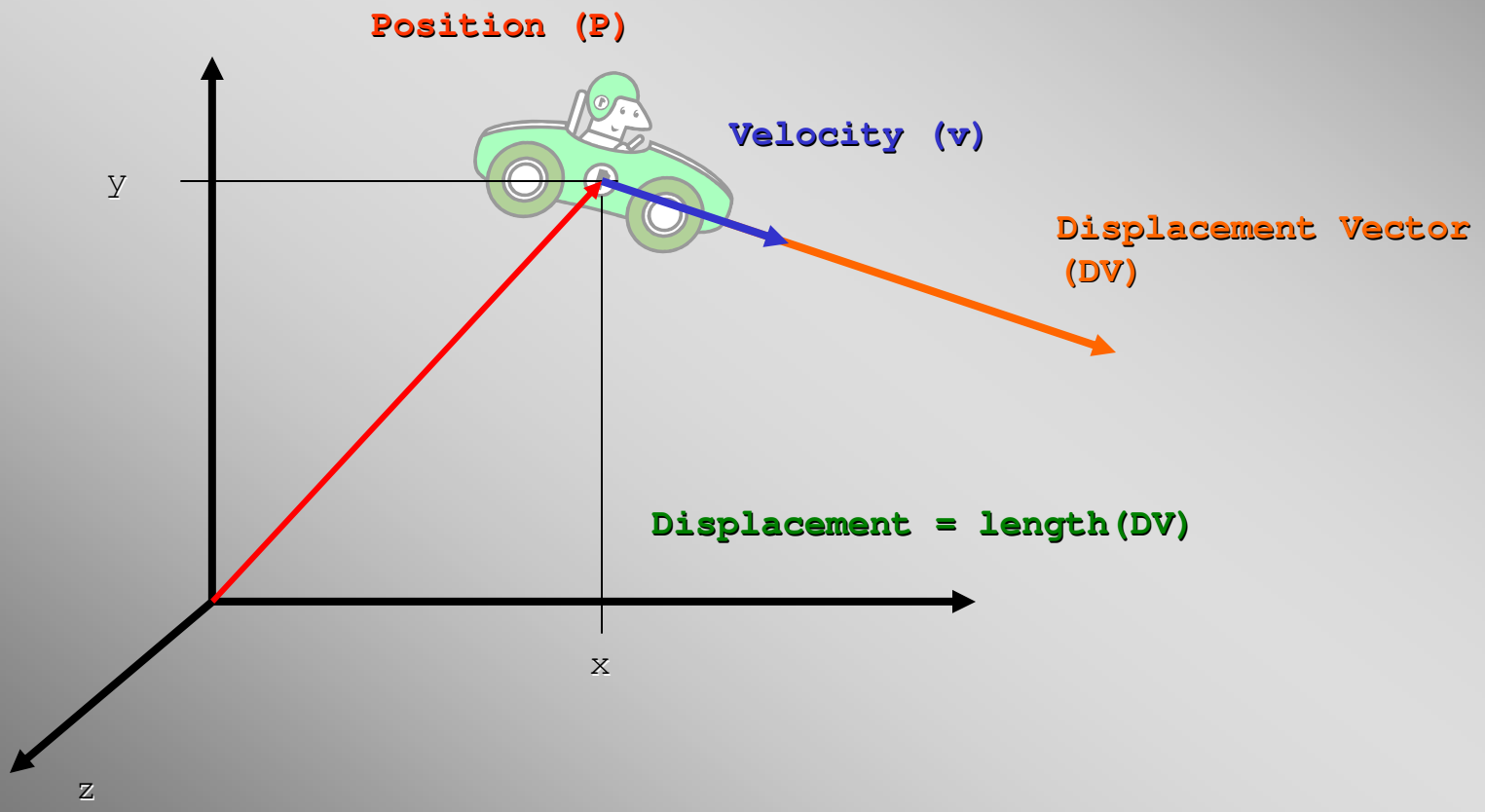


Image by chocoloco.deviantart.com

Movement


Position (P)
Velocity (v)
Displacement Vector
(DV)
Displacement = length(DV)
y
x
z

Naïve approach

```
myInit {
    vector3 vel(5.0f, -5.0f, 0.0f);
    vector3 mypos(0.0f,0.0f,0.0f);

}
```

```
myDisplay{
   mypos += vel;
   glTranslate3f(mypos[0],mypos[1],mypos[2]);
}
```

What's wrong with this?

Naïve approach

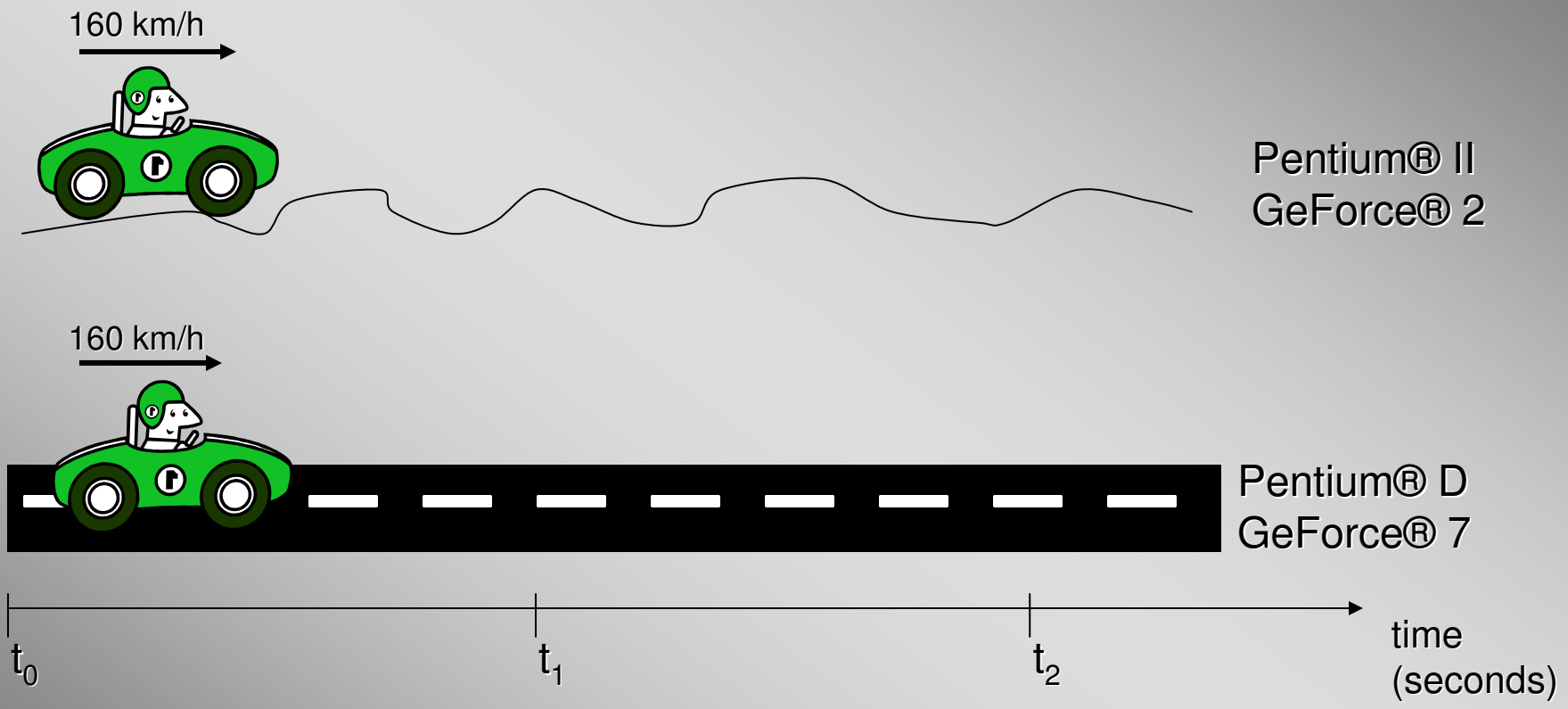
- A fixed displacement will result in same distance traveled each frame

  Different computers have different frame rates

  → Movement will be faster in some machines

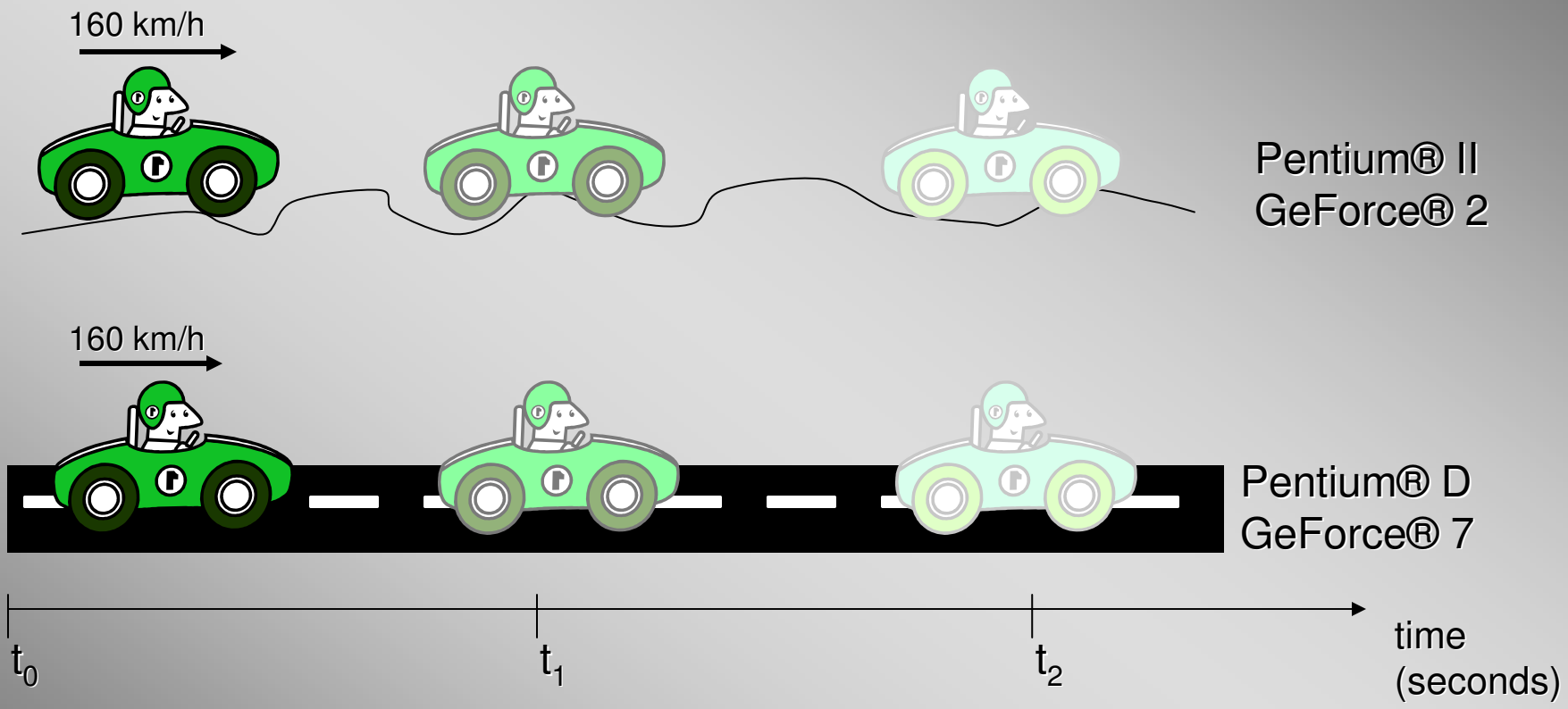
30fps → 30 x 5 = 150 units/second
60fps → 60 x 5 = 300 units/second

Setting your car in cruise control



See http://auto.howstuffworks.com/cruise-control.htm

Setting your car in cruise control



See http://auto.howstuffworks.com/cruise-control.htm

Better approach

```
myInit {
    vector3 vel(5.0f, -5.0f, 0.0f);
    vector3 mypos(0.0f,0.0f,0.0f);
    float start_time = current_time;

}
```

```
myDisplay{
    float delta_time = current_time-start_time;
    vector3 displacement = vel * delta_time;
    mypos += displacement;
   glTranslate3f(mypos[0],mypos[1],mypos[2]);
}
```

see: glutGet(GLUT_ELAPSED_TIME);

References

- http://colalg.math.csusb.edu/~devel/precalcdemo/param/src/param.html
- http://mathworld.wolfram.com/ParametricEquations.html
- http://www.mathwords.com/d/derivative_rules.htm