



# **Komponentní systém pro herní grafický engine**

Bakalářská práce

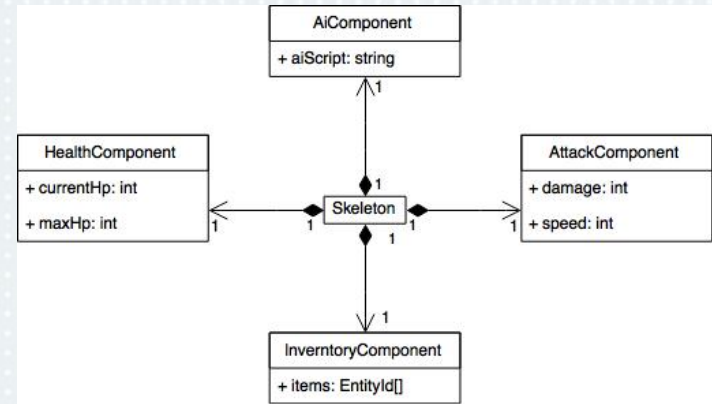
Vedoucí: Ing. Michal Španěl, Ph.D.

Autor: Tomáš Polášek

30. leden 2017

# Cíl

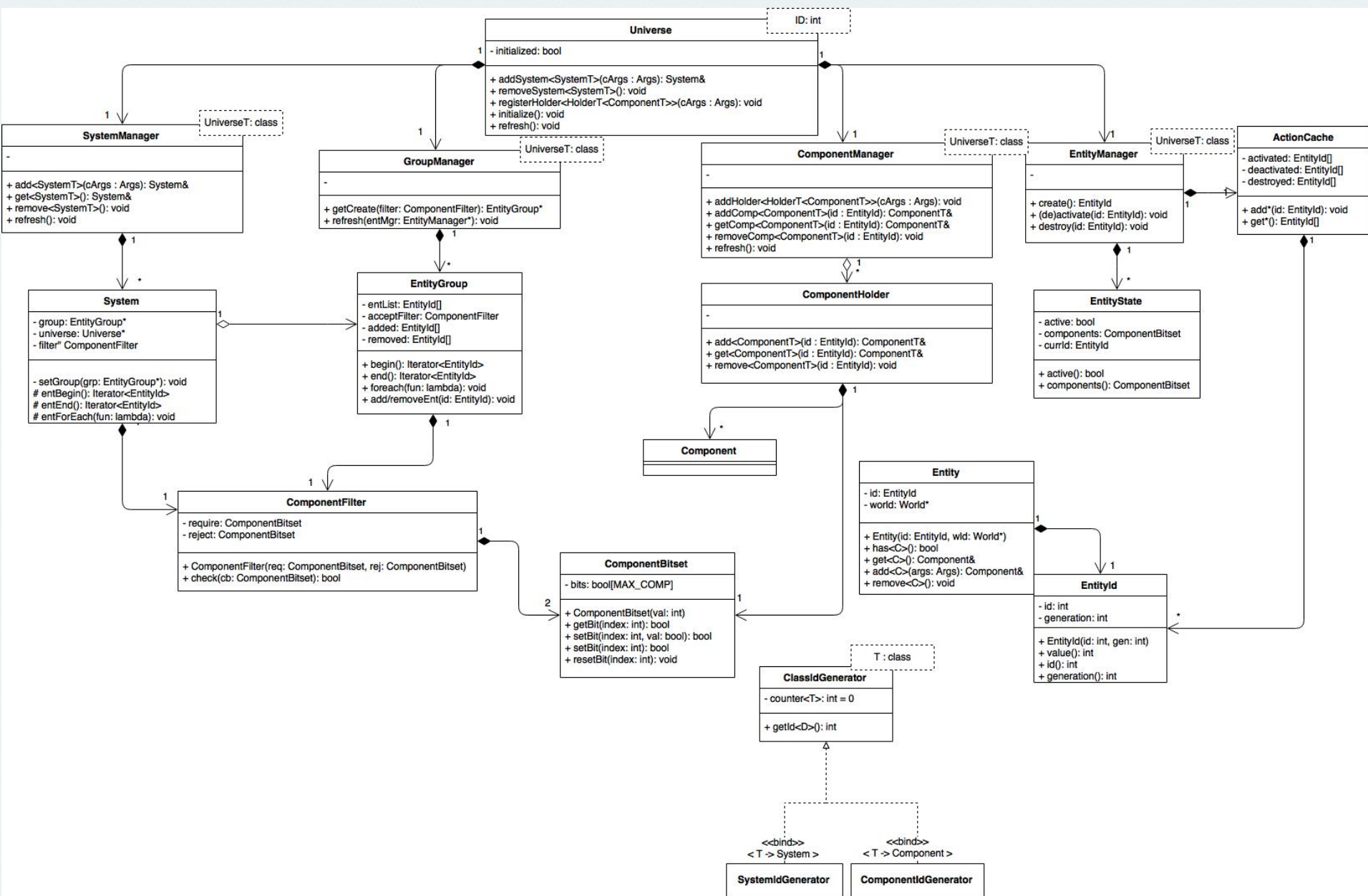
- Co to je ECS?
- Knihovna Entropy (C++).
- Unit testing
- Dokumentace
- Příklady použití



# Návrh řešení

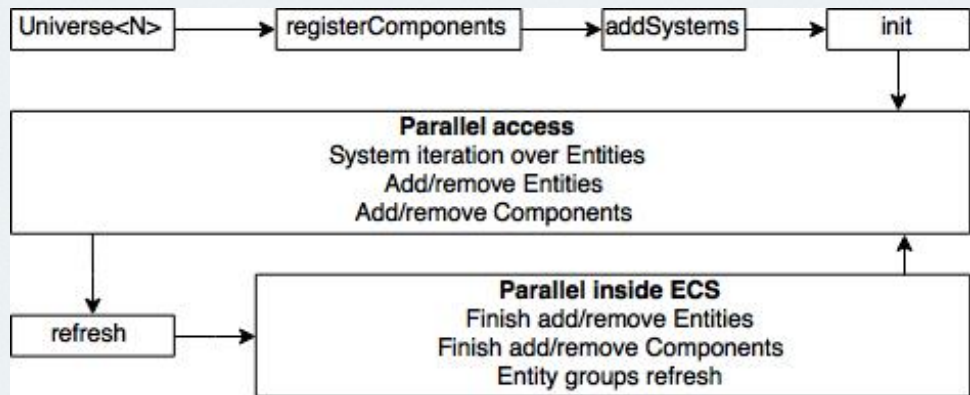
- Universe<N> – rozhraní uživatele knihovny
- Manager třídy
- Entity, Component, System
- Řídící struktura
- Paralelizmus

# Návrh řešení



# Návrh řešení

- Universe<N> – rozhraní uživatele knihovny
- Manager třídy
- Entity, Component, System
- Řídící struktura
- Paralelizmus



# Aktuální stav

- Prototyp
- Unit testing
- Profilování
- Implementace podle návrhu



# Plán řešení

- Dokončení implementace podle návrhu
- Implementace paralelizmu
- Příklady použití
- Optimalizace

```

#include <Entropy/Entropy.h>
using namespace ent;

struct PosComponent {
    float x, y;
}

struct VelocityComponent {
    float xVel, yVel;
}

struct AirResistanceSystem : public System<Require<VelocityComponent>> {
    static constexpr float AIR_RESISTANCE{0.1};
    void run (float delta) {
        foreach([&](Entity &e){
            e.get<VelocityComponent>().xVel *= AIR_RESISTANCE * delta;
            e.get<VelocityComponent>().yVel *= AIR_RESISTANCE * delta;
        })
    }
}

/* MovementSystem ... */
int main() {
    auto u{Universe<0>::instance()};
    u.registerHolder<ComponentHolder<PosComponent>>();
    u.registerHolder<ComponentHolder<VelComponent>>();
    auto resSys{u.addSystem<AirResistanceSystem>()};
    auto movSys{u.addSystem<MovementSystem>()};
    constexpr float END{100.0};
    float time{0.0}, delta{0.1};
    while (time < END) {
        movSys.run(delta);
        resSys.run(delta);
        time += delta;
    }
}

```

<https://www.draw.io/>

<http://pygments.org/>