

Algoid – ALGEA
www.algoid.net

miX-TesN

Présenté par Yann Caron
aka CyaNn

Rappel :: Les objets

- Une voiture est un objet
- Elle possède des propriétés (des variables)
 - couleur
 - nombre de portes
- Qu'est ce qu'une voiture ?



Rappel :: Les méthodes

- Une voiture a des méthodes (fonctions)
 - conduire
 - tourner le volant
- Que peut-on faire avec une voiture ?



Rappel :: Les évènements

- Et des évènements
 - caler
 - crever
- Qu'est-ce qu'il peut se passer avec une voiture ?



ALGEA :: Actor

- Dans les jeux vidéos, les acteurs sont des objets graphiques
- Ils ont donc des propriétés
 - image
 - position
 - rotation
 - hitBox
 - interaction



ALGEA :: Actor

- Des méthodes
 - move
 - Rotate
 - say
 - applyForce
 - impulse
- Des évènements
 - onContact
 - onPulse

Notre jeu :: CuteJet

- Jetpack Joyride like



Notre héros

```
set hero = gameLayer.createActor(  
    "PlanetCute/beetleship");  
  
hero.move(200, 400);
```



Parler

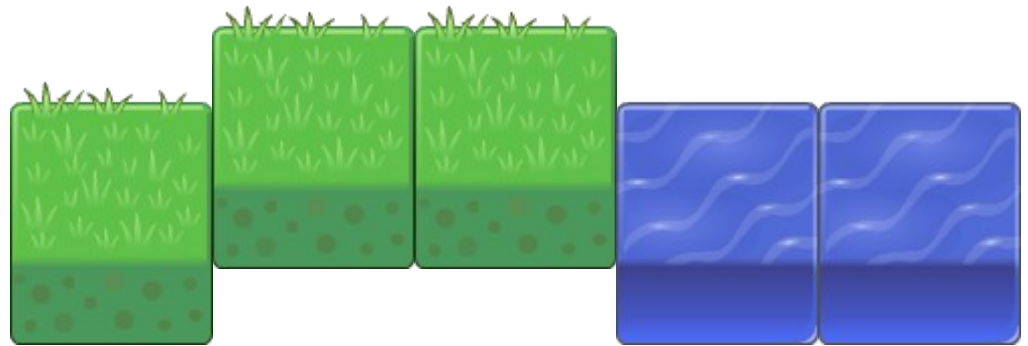
```
hero.say(font, "Allons y !");
```



Le Niveau

```
set x = 0;  
loop(50) {  
    set ground = gameLayer.createActor("PlanetCute/Grass Block");  
    ground.move(x, 50);  
  
    x = x + ground.getWidth();  
}
```

- Une boucle
- Une variable
- On incrémente
- D'autres blocks ?



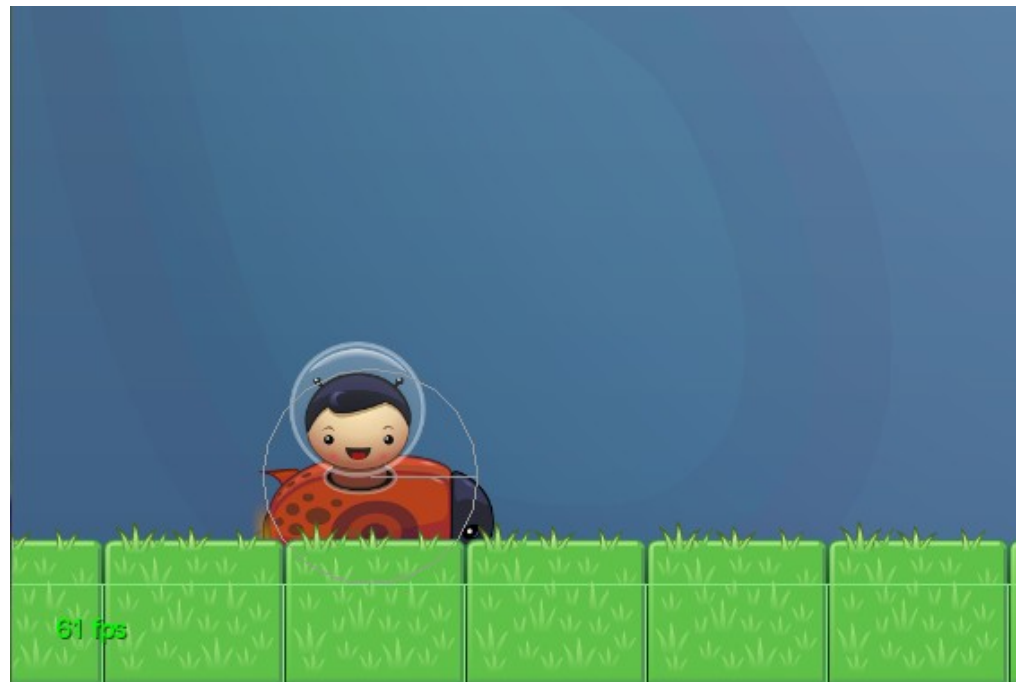
Physique

- Les acteurs interagissent avec le monde
- Type d'acteur
 - `typeStatic()`, `typeDynamique()`, `typeSensor()`
- Static
 - Les murs, le sol, le décor
- Dynamique
 - Notre héros
- Sensor
 - Les objets non physique (bonus)

Physique :: exemple

- Notre héros :: `hero.typeDynamic().hitCircle(15);`
- Le sol :: `ground.typeStatic().hitSquare(75);`

On appuie sur
le bouton Debug



Animation

- Maintenant on veut que notre héros avance tout seul !
- Et ça marche !!!

```
// le heros avance
hero.applyForceX(25);

// mais ne tourne pas
hero.fixeRotation();

// on limite sa vitesse
hero.setMaxVelocity(20);

// et la caméra le suit
gameLayer.follow(hero);
```

Fonction et évènement

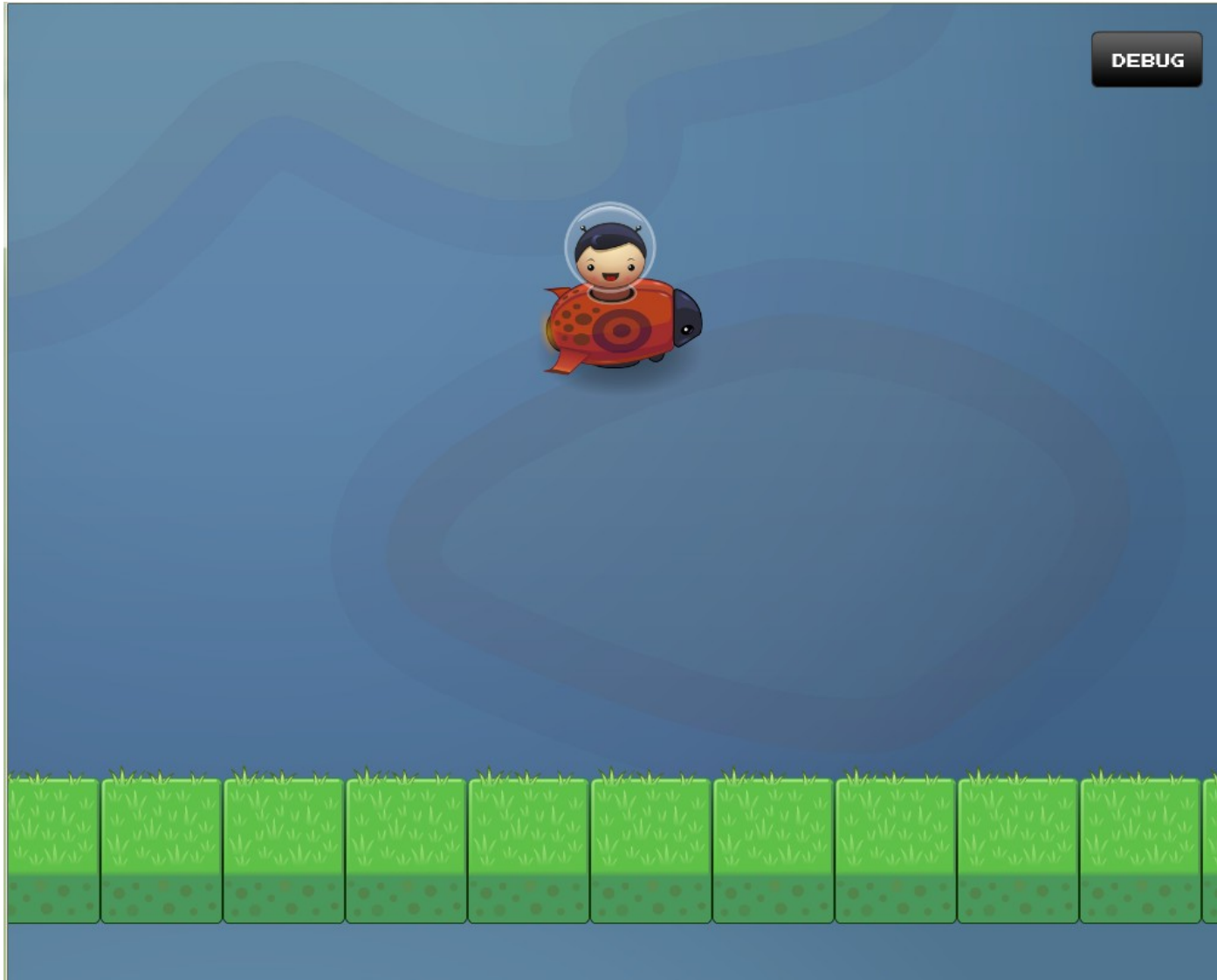
- Quand on clique, on veut que notre héros saute

```
set jump = function() {  
    hero.impulseY(15);  
}
```

```
stage.onTap(jump);
```

- Il est vivant ! Il vole !

Et voilà le travail !



Une petite pause ?



Random

- La fonction `math.random(10)`
 - Choisit un nombre aléatoirement entre 0 et 10. Ou tout autre nombre.
- On veut générer aléatoirement des pierres précieuses
- Comment faire ???



Pierres précieuses

```
loop(25) {  
    set x = math.random(gameLayer.getWidth())  
    set y = math.random(gameLayer.getHeight())  
  
    set gem = gameLayer.createActor(  
        "PlanetCute/Gem Orange")  
    gem.move(x, y).scale(0.5)  
    gem.typeSensor().hitCircle()  
}
```



Bonus

- Et on veut gagner des points quand on touche une pierre

```
loop(25) {  
  // .... Génération des pierres ..  
  set winGem = function () {  
    score = score + 10;  
    gameLayer.remove(gem);  
  };  
  gem.onContactWith(hero, winGem);  
}
```

Dangers

- Maintenant il faut éviter des cailloux

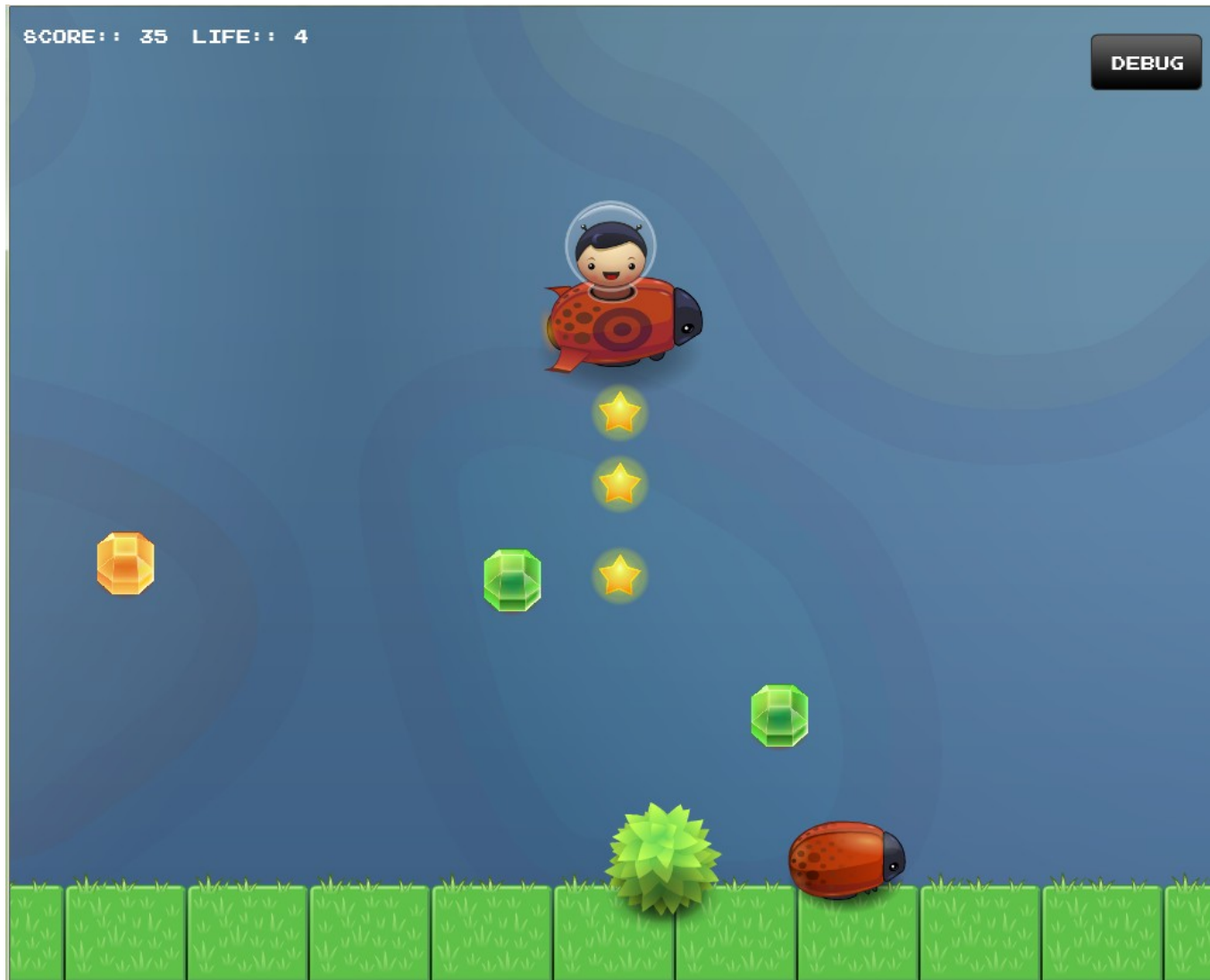
```
set throwRock = function () {  
    // Comme les pierres  
  
    rock.typeSensor().hitCircle();  
    rock.setXVelocity(-5);  
  
    rock.onContactWith(hero, gameOver);  
}  
hero.onPulse(throwRock, 1500);
```



Ca y est, on sait faire un jeu !

- Imaginons qu'il faille éviter des pierres
- Ou des projectiles
- Qu'on veuille tirer sur des ennemis
- Qu'on fasse marcher les ennemis

Et voilà le travail



Le son

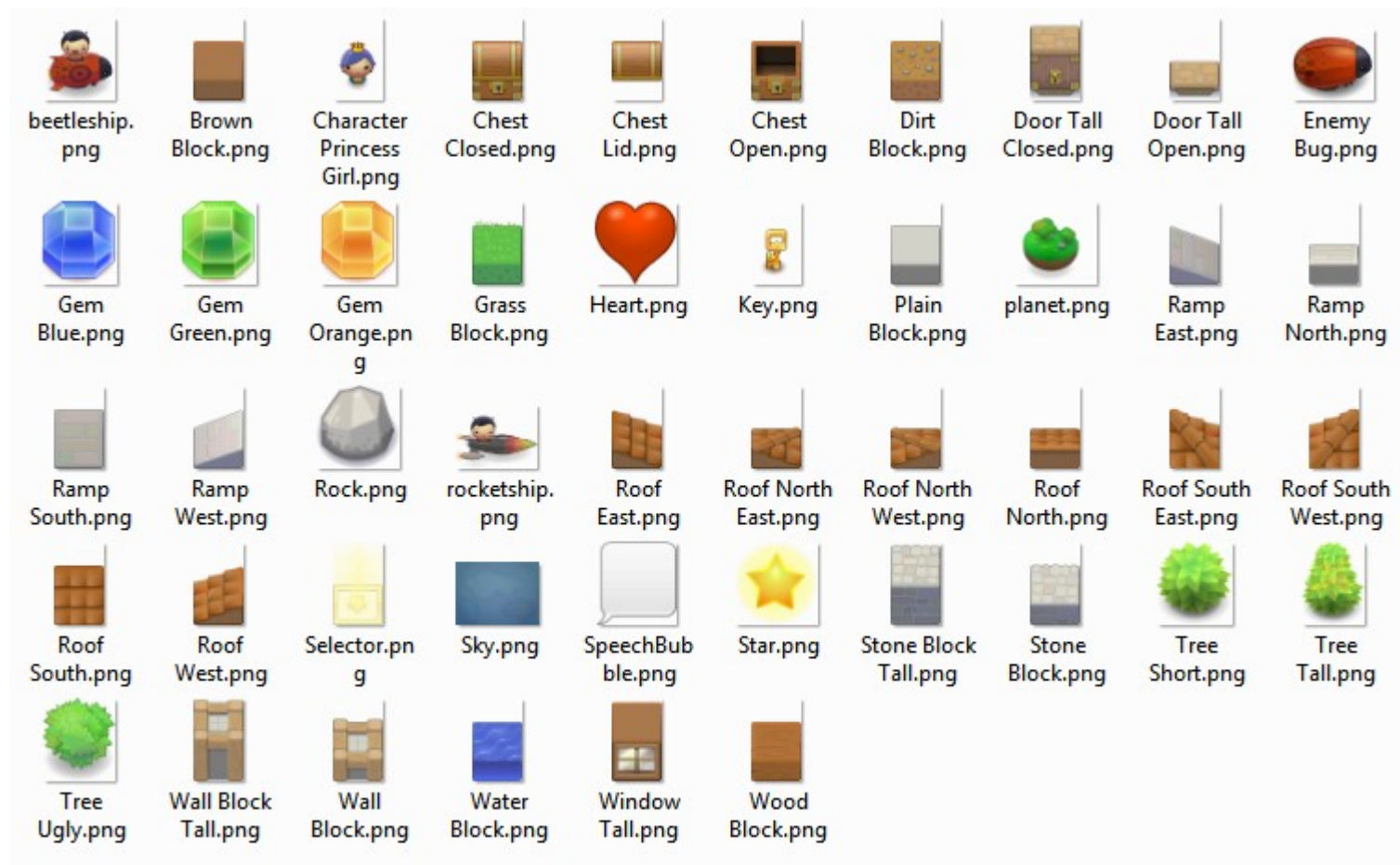
- On crée un son
- Et on le joue en boucle

```
set music = game.createSound("Loop/Vibraphone")  
music.playLoop();
```

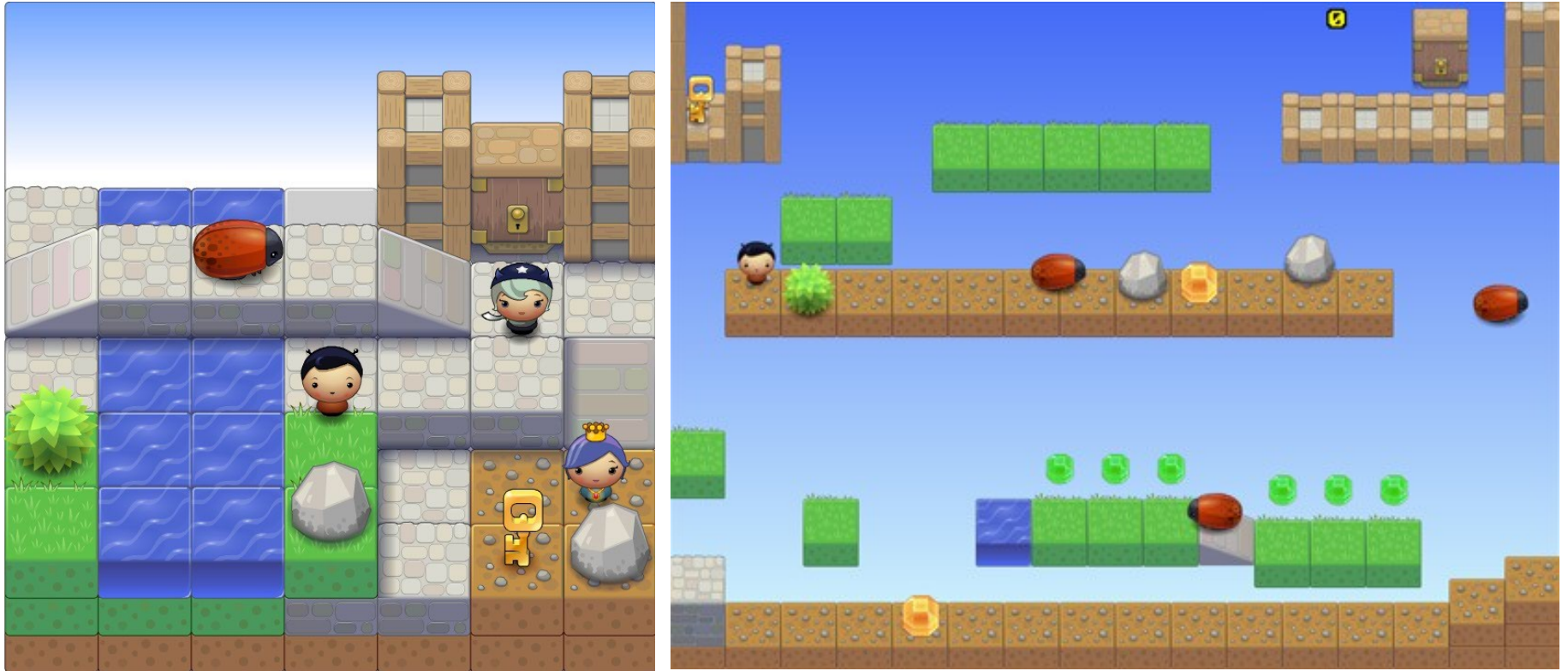
- Ou une seule fois:: play
- On le contrôle:: pause, resume, stop

De quoi disposons-nous ?

- Dans le répertoire ALGEA/assets img ou sound



PlanetCut



Merci à Daniel Cook pour son game-design

<http://www.lostgarden.com/2007/05/>

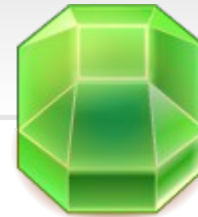
Pour aller plus loin

Les pierres précieuses au hasard

- On veut choisir parmi 3 couleurs de pierres (**tableaux, concaténation**)

```
set gems = array {"Gem Blue", "Gem Green", "Gem Orange"};
loop(25) {
    // ....
    set g = math.random(gems.length());

    set gem = gameLayer.createActor("PlanetCute/" .. gems[g]);
    // ....
}
```



Collision selon l'acteur ou son nom

- Gérer la collision selon ce que l'on touche

```
set contact = function (enemy, other) {  
  if (other.getName() == "PlanetCute/Star") {  
    // Si c'est une étoile  
  } elseif (other.getName() != "PlanetCute/Grass Block") {  
    // Les autres sauf l'herbe  
  }  
}  
  
enemy.onContact(contact);
```