

Algoid - ALGEA
www.algoid.net

DEVOXXTM
4KIDS

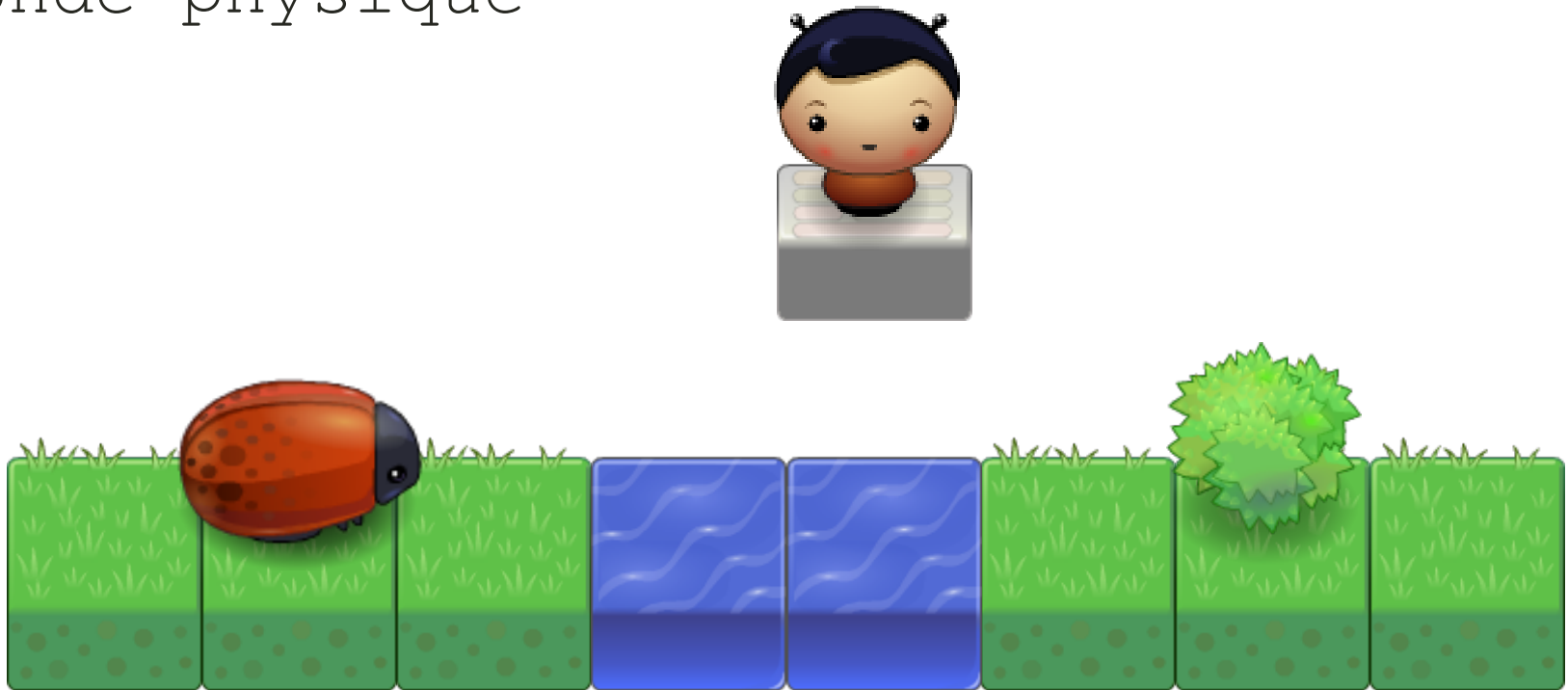


FRANCE

Présenté par Yann Caron
aka CyaNn

ALGEA :: Actor

- Dans les jeux vidéos, les acteurs sont des objets graphiques
- Qui interagissent entre eux dans un monde physique



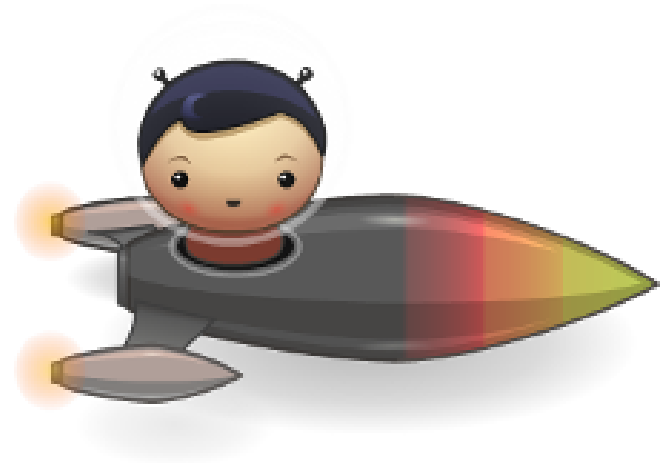
Rappel :: Les objets

- Un acteur (ex : le hero) est un objet
- Il possède des propriétés (des variables)
 - Image
 - Position
 - Hitbox
 - Type physique
- **Qu'est ce qu'un acteur ?**



Rappel :: Les méthodes

- Un acteur a des méthodes (fonctions)
 - Bouger (move)
 - Tourner (rotate)
 - Dire (say)
 - Pousser (applyForce)
 - Lancer (impulse)



- **Qu'est-ce qu'un acteur peu faire ?**

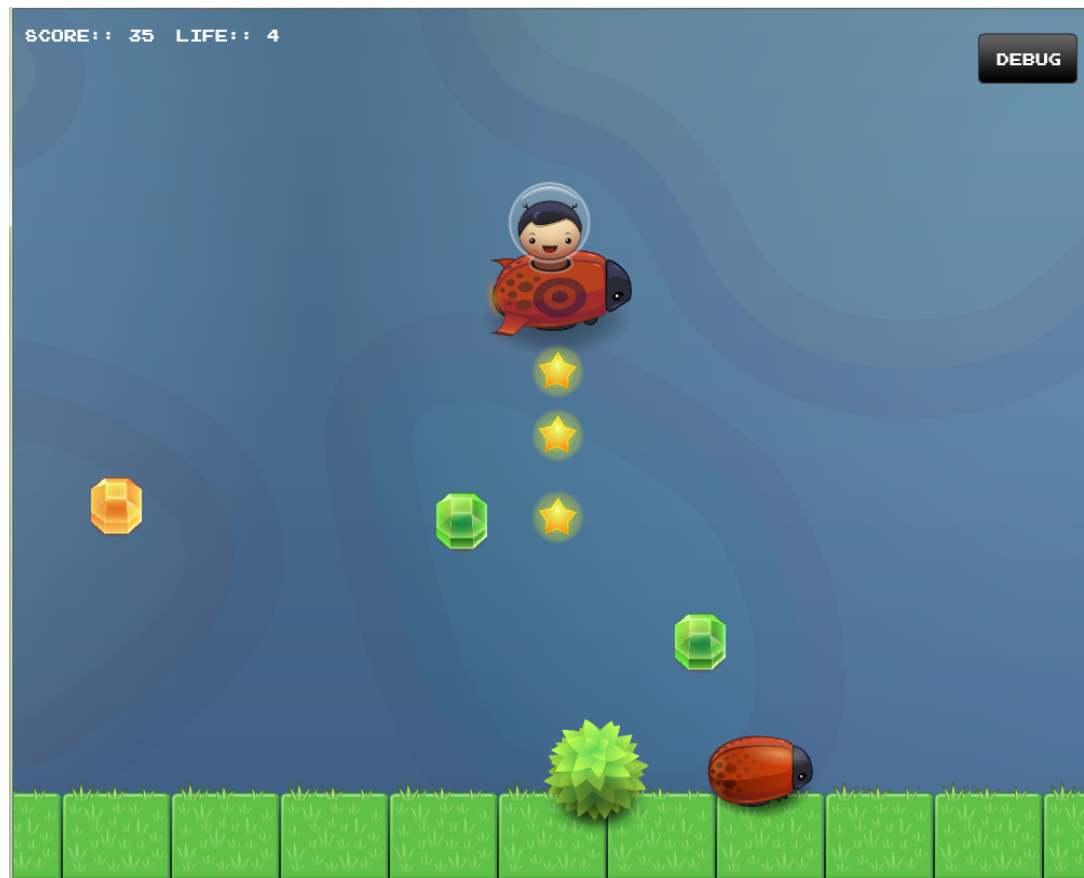
Rappel :: Les évènements

- Et des évènements
 - Toucher (`onContact`)
 - Cliquer (`onTouch`)
- Qu'est-ce qu'il peut se passer avec un acteur ?

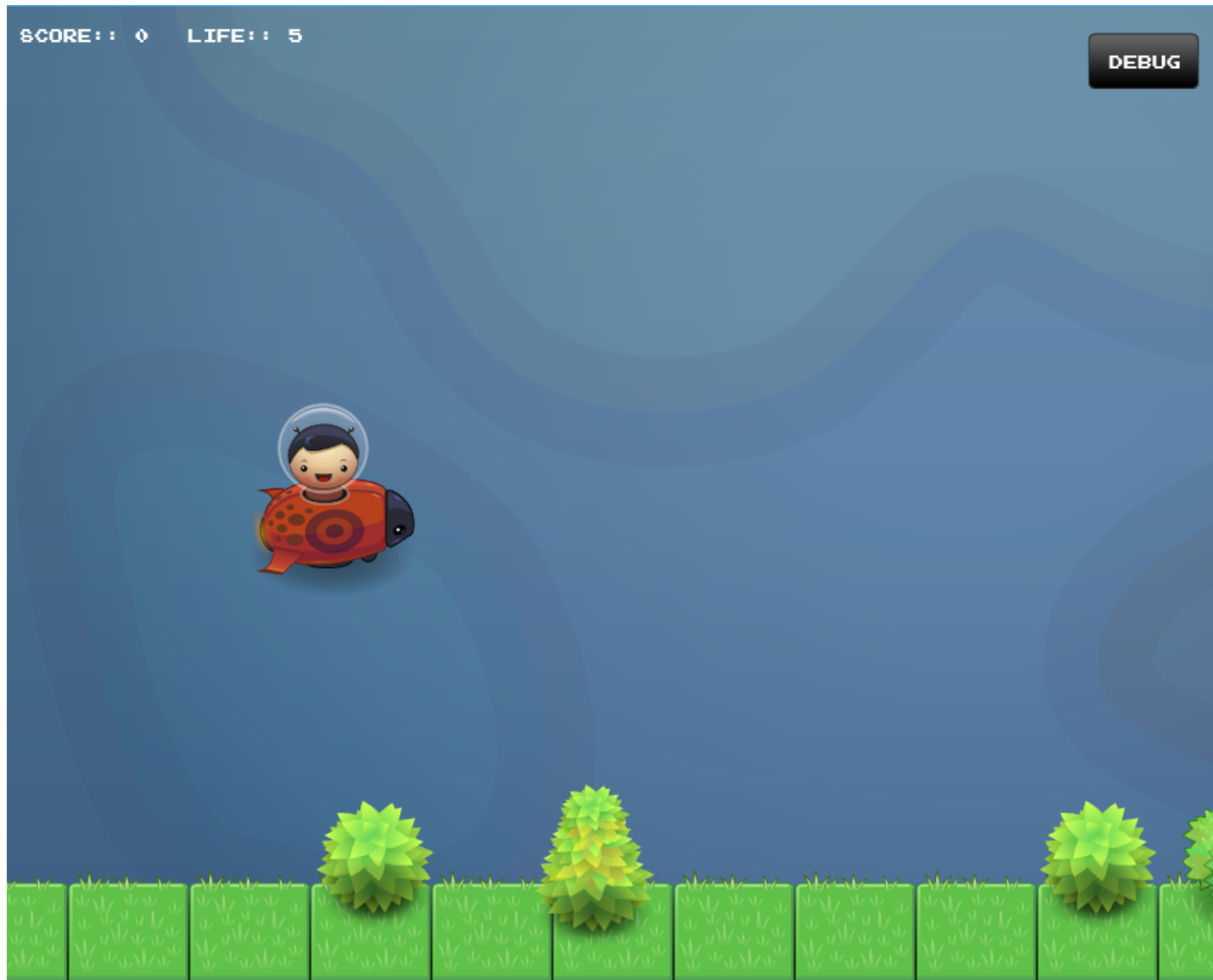


Notre jeu :: CuteJet

- Jetpack Joyride like - 2em partie



Au départ



Les gemmes

```
set gem = gameLayer.createActor(  
    "PlanetCute/Gem Green");
```

```
gem.scale(0.5)
```

```
gem.move(200, 400)
```

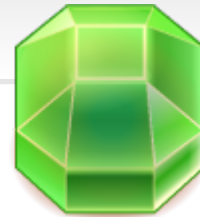


Jeu de hasard

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

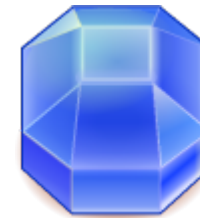
Jeu de hasard

```
set x = math.random(gameLayer.getWidth())  
set y = math.random(gameLayer.getHeight())  
  
gem.move(x, y)
```



Plusieurs gemmes

- On veut créer 25 gemmes dans notre niveau
- Comment faire ???



Plusieurs gemmes

```
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()  
}
```

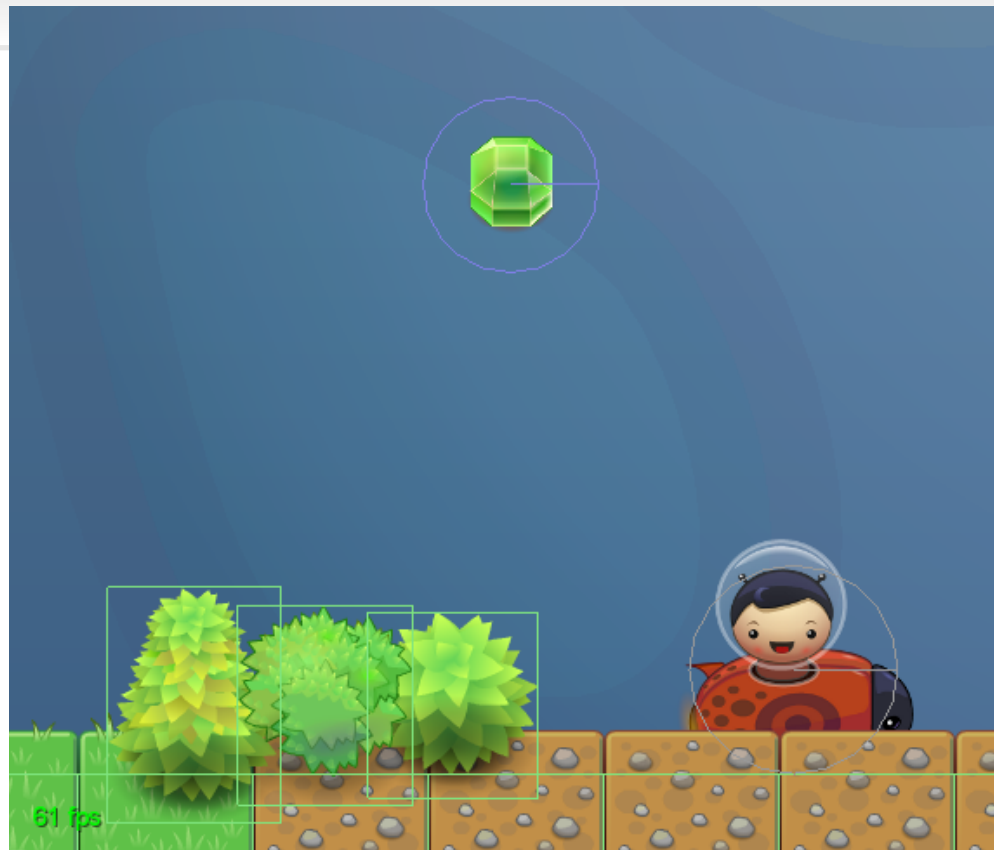


Physique

- Les acteurs interagissent avec le monde
- Type d'acteur
 - `typeStatic()`, `typeDynamique()`, `typeSensor()`
- Static
 - Les murs, le sol, le décor
- Dynamique
 - Notre héros, les ennemies
- Sensor
 - Les gems

Physique des gemmes

```
gem.scale(0.5);  
gem.typeSensor().hitCircle();
```



Fonction et évènement

- Quand on touche le hero, on veut qu'il gagne des points

```
set winGem = function() {  
    score = score + 10;  
    gameLayer.remove(gem);  
}  
  
gem.onContactWith(hero, winGem);
```

- Et ça marche !

Dangers

- Maintenant il faut éviter des cailloux

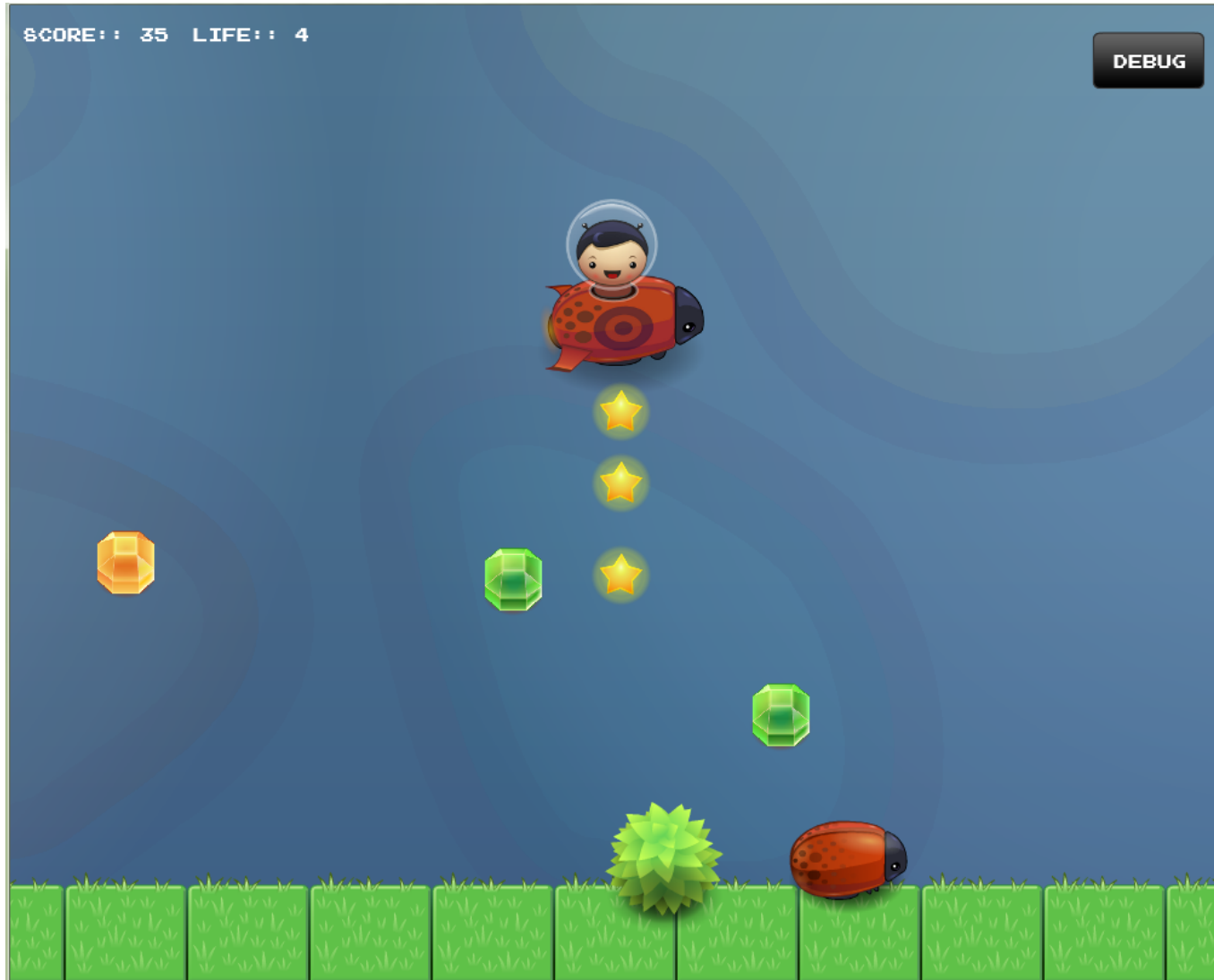
```
set throwRock = function () {  
    // Comme les gemmes  
  
    rock.typeSensor().hitCircle();  
    rock.setXVelocity(-5);  
  
    rock.onContactWith(hero, gameOver);  
}  
hero.onPulse(throwRock, 1500);
```



Ca y est, on sait faire un jeu !

- Pour aller plus loin, imaginons :
 - Que l'on crée des ennemis
 - Qu'on veuille tirer sur ces ennemis
 - Qu'on fasse marcher les ennemis

Et voilà le travail



Pour aller encore plus loin

Parler

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



Le son

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

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

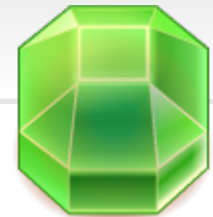
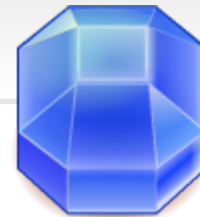
- Ou une seule fois:: play
- On le contrôle:: pause, resume, stop

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]);
  // ....
}
```



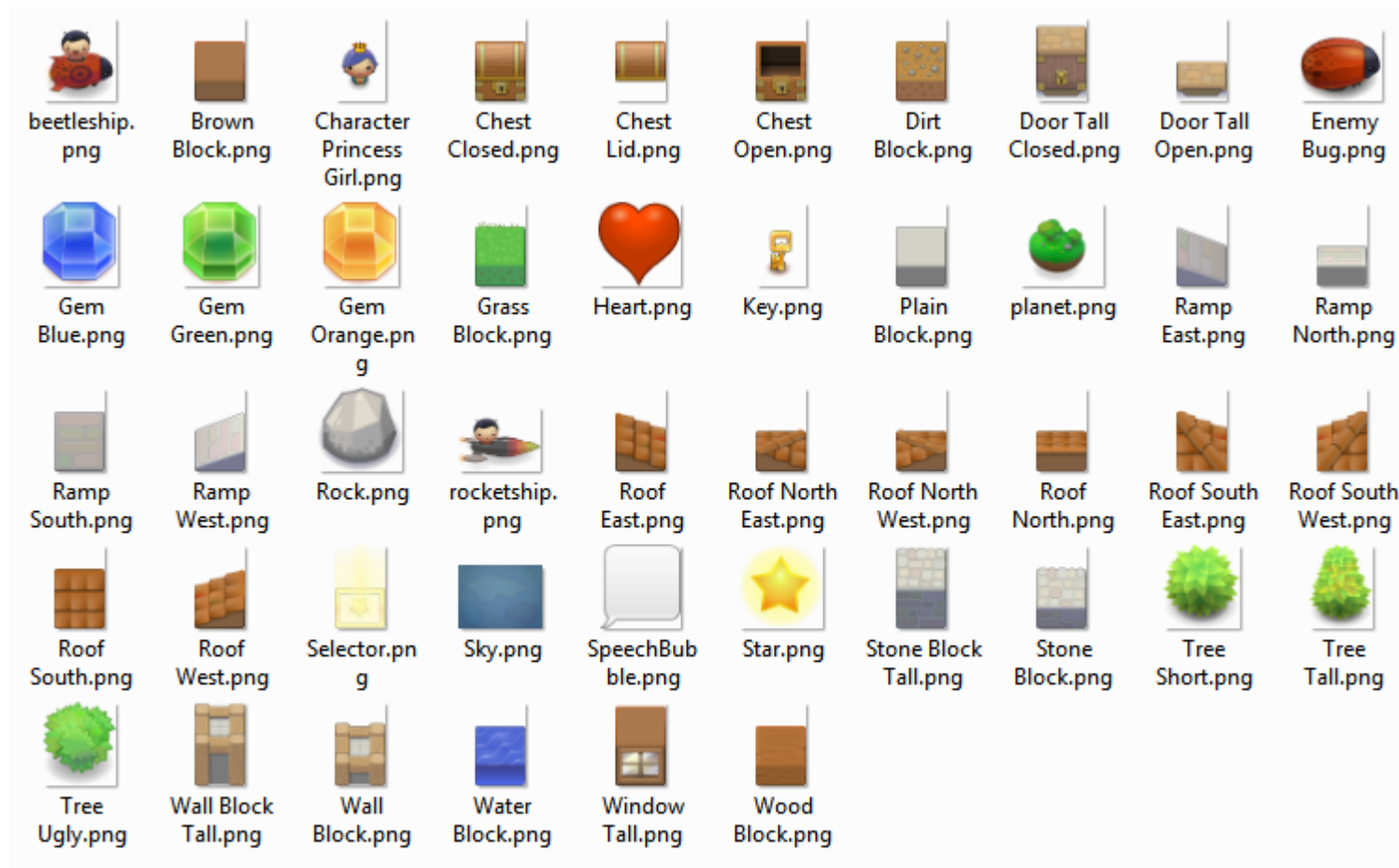
Collision selon l'acteur ou son nom

- 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);
```

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/>