



# DEWAGNIER ALEXANDRE

## Rapport de Stage

Intégration et gestion d'animations Spine dans un jeu video de type « action/aventure » multiplateforme crée sur Unity 3D.

du 23/06/16 au 16/09/16



le **cnam**  
École nationale du jeu  
et des médias interactifs numériques **enjmin**

# REMERCIEMENTS

Je voudrais commencer par remercier Damien Chauveau et David Deckeur de m'avoir accueillis dans le studio Aurora pendant la durée de mon stage et de m'avoir de tout temps accordé un soutien et une attention constante, je souhaiterais aussi souligner le très bon accueil qui m'as été fait par les autres membres du studio.

J'aimerais également adresser un remerciement spécial aux membres du studio Bulwark qui m'ont très gentiment installés dans leurs locaux pendant plusieurs semaines me permettant de trouver chaque jour un espace de travail adapté.

Et pour finir merci encore aux membres du secrétariat de l'ENJMIN pour leur attention et pour leur efficacité.

# SOMMAIRE

REMERCIEMENTS	2
SOMMAIRE	3
1. AURORA STUDIO	4
▪ HISTOIRE DU STUDIO	4
▪ AURORA AUJOURD'HUI	4
2. MA MISSION	5
▪ ILIOS BETRAYAL OF GODS	5
▪ ILIOS BETRAYAL OF SPINE: TECHNIQUE D'ANIMATION	8
▪ LA SUITE DU STAGE	12
3. LA FIN DE L'AVENTURE	13
4. ANNEXES	14
▪ ILIOS	14
▪ OUTILS	17

# 1. AURORA STUDIO

## ▪ HISTOIRE DU STUDIO

L'idée d'Aurora Studio est née il y a de nombreuses années de la réunion de passionnés de jeux vidéos qui désiraient travailler ensemble. Leur premier projet a été Saint Seiya RPG, un projet amateur adaptant l'univers et l'histoire du mangas Saint Seiya<sup>1</sup>, mais n'ayant pas obtenu les droits par l'entreprise Bandai, à cause de leur manque d'expérience et d'un différent culturel, la licence étant japonaise, le projet fut annulé.

## ▪ AURORA AUJOURD'HUI

Désormais, forts de leurs travaux passés et de plusieurs expériences dans le monde du jeu vidéo, les membres fondateurs du studio travaillèrent lors de leur temps libre sur un nouveau projet gardant l'univers de la mythologie grecque de Saint Seiya RPG: Ilios Betrayal of Gods. Ayant réussi à mettre sur pied une démo jouable du jeu, un kickstarter fut mis en place pour financer la suite du projet et ainsi par la même créer Aurora Studio, entreprise de jeu vidéo.



<sup>1</sup> Les Chevaliers du Zodiaque en France

## 2. MA MISSION

### ▪ ILIOS BETRAYAL OF GODS

Première création du studio Aurora, Ilios Betrayal Of Gods est un jeu indépendant aventure/action en 2D qui nous mets dans la peau d'un héros inspiré de l'Illiade et l'Odyssée, Politès, ultime survivant de la guerre de Troie. Possédant un immense désir de vengeance, vous le suivrez dans ses aventures et vous confronterez aux Dieux qui ne sont pas ce qu'ils semblent être. Ilios Betrayal of Gods, s'inspire principalement de jeux tels que Castelvania, Odin Sphere, ou encore Bayonetta et Devil May Cry, de par leurs gameplay riches, nerveux, et de leurs histoires profondes.

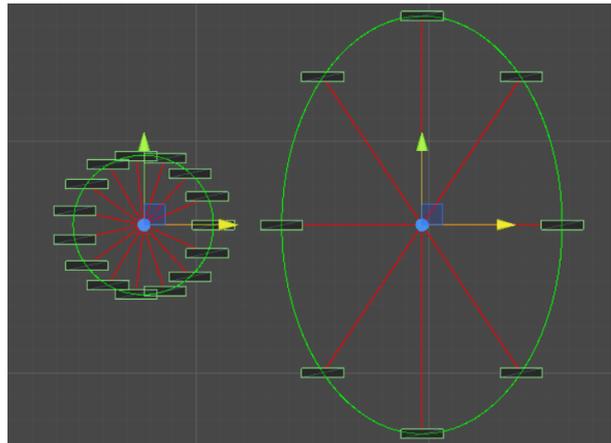
Pour le casting, Aurora a fait appel à des véritables « guest » au niveau du doublage, on pourra par exemple retrouver Le Joueur du Grenier, Benzaïe, Bob Lennon ou encore Sparadrap de la série Noob. Le studio a lancé un kickstarter afin de financer le projet, mais ce dernier n'a malheureusement pas abouti, le jeu ayant trouvé son public en France mais pas à l'étranger.



## ▪ DÉCOUVERTE DU PROJET

Lors de mon entrée dans la structure une de mes premières tâches était bien sûr de prendre en main le projet, j'ai donc passé plusieurs jours à découvrir la structure utilisée pour la création du prototype et à assimiler les conventions d'écriture du lead programmeur. Dû à la relative jeunesse du studio c'est quelque chose que j'ai dû faire en autonomie mon responsable travaillant très souvent sur des prestations à l'extérieur afin de financer de futurs développements. **Suite à cette prise en main, plusieurs missions «d'échauffements» m'ont été confiées:**

- La mise au point d'un **visualisateur d'animations** permettant de valider l'importation des assets graphiques créés par les graphistes à l'aide du logiciel Spine.
- L'intégration d'un outil de gestion de camera pour jeux 2D, **ProCamera 2D**, et test d'utilisabilité effectués de concert avec Damien Chauveau Game Designer du studio.
- Mise au point d'un **outil de level design** permettant d'ajouter simplement des plateformes tournant autour d'un axe dans les niveaux du jeu.
- Développement d'un **autre outil de level design** utilisé pour définir des zones d'actions spéciales modifiant le comportement du personnage principal.



Outil de level design

Ces missions simples m'ont permis de bien prendre en main les divers outils servant à la réalisation du jeu mais aussi et surtout de mieux comprendre la vision des membres de l'équipe.

## ▪ UN PROBLÈME D'ANIMATION

Je me suis vite rendu compte que l'USP<sup>2</sup> principale du jeu résidait surtout dans son aspect graphique très réaliste et dans la capacité du studio à créer des personnages à l'animation irréprochable. Une des inspirations moderne<sup>3</sup> de l'équipe pour le travail graphique est le studio Lab Zero<sup>4</sup> à l'origine du jeu de combat *Skull Girls* et du jeu en cours de développement *Indivisible*. Ces deux titres partagent en commun des animation très vives et précises donnant littéralement vie aux personnages. Malheureusement difficile pour un studio indépendant débutant d'atteindre une telle qualité sans de réels moyens financiers. Les graphistes d'Aurora ont donc dû se tourner vers d'autres techniques que l'animation traditionnelle pour atteindre l'excellence graphique visée. Après plusieurs essais et de longues recherches, la technique dite du cut-out utilisé sur des titres comme *Odin sphere* ou *Muramasa* à été retenue.



Sprites issus de « Indivisible »

---

<sup>2</sup> Unique Selling Point

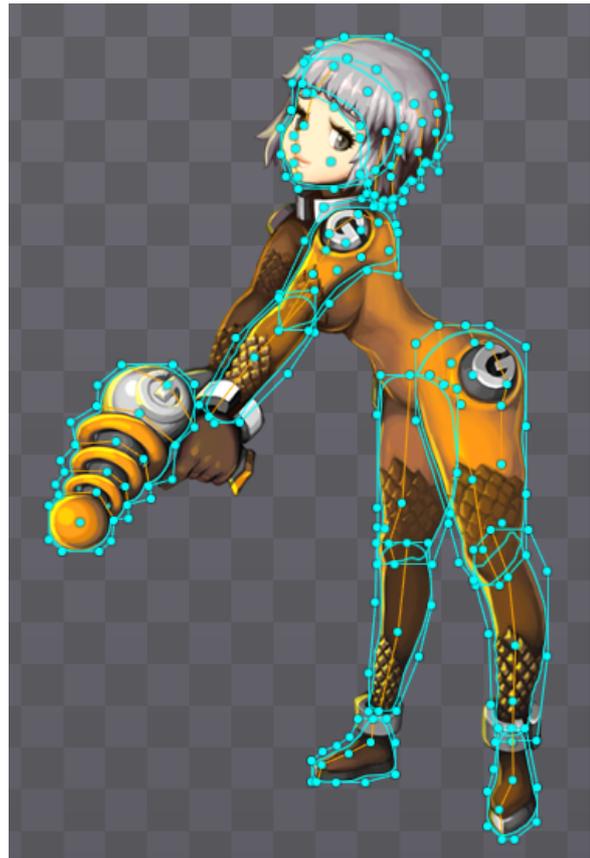
<sup>3</sup> Nous pouvons aussi noter dans les inspirations les jeux du studio Vanillaware

<sup>4</sup> <http://revergelabs.com>

## ▪ ILIOS BETRAYAL OF SPINE: TECHNIQUE D'ANIMATION

Le cut-out est une **technique d'animation** qui repose sur le **decoupage de l'objet à animer en différentes parties** qui seront par la suite posées sur un **squelette virtuel** permettant un travail d'animation très similaire à celui du monde de la 3D. Cette technique possède plusieurs avantages:

- Il n'est plus nécessaire de **redessiner l'intégralité du personnage** à chaque frames d'animation, cela permet de passer plus de temps sur le painting des sprites afin d'atteindre un rendu précis et réaliste.
- L'objet en jeu étant représenté par un squelette, les os le composant sont accessibles, **il est ainsi très facile de calculer les transitions entre différentes poses clefs** sans avoir à les animer à la main.
- Les animations n'étant que des mouvements des squelettes **la charge en mémoire des graphismes est amoindrie**, un seul atlas de textures peut être utilisé pour des centaines d'animations



Exemple de Setup Spine

Plusieurs outils professionnels permettent d'animer de cette manière, mais l'outil choisi par Aurora Studio est **Spine**, développé par Esoteric Software. Cet outil a pour avantage d'être **pensé pour le workflow du jeu vidéo**, il propose un export simplifié vers Unity couplé à une librairie complète permettant d'intégrer facilement une animation en jeu, de plus cet outil **représente les parties de l'objet animé avec des meshes**, il est donc **facile de les déformer** et d'ajouter des passes d'animations secondaires très facilement et ainsi créer du volume et de la vie.

## ▪ UN HÉROS TRÈS ANIMÉ

Lors de la confection de la demo d'Ilios betrayal of gods, une chose est apparue très clairement, si cette méthode d'animation est parfaite pour les ennemis du jeu<sup>5</sup> **elle trouve vite ses limites quand appliquée à Polités le personnage principal**. En effet de par son design et à cause du fait qu'il est le point central de l'action il est très difficile d'obtenir un mouvement pertinent tout en gardant les avantages de Spine.



**Des concessions ont dû être faites**, notamment ajouter beaucoup de sprites intermédiaires dans les animations ce qui a pour conséquence de se couper du système de transition automatique<sup>6</sup> et de ralentir le procédé de création graphique. De plus les mouvements de ce personnage ayant un fort effet sur le game feel, leurs précisions, leurs timings et leurs impacts se doivent d'être parfaits. Pour cela l'équipe graphique a défini **plusieurs poses clefs** dites rough réparties sur plusieurs squelettes représentant des familles d'actions. Ces animations sont ensuite testées dans la demo afin de déterminer le bon timing, et de voir si **le mouvement ne trahit pas la volonté du game designer** d'avoir un jeu très réactif et contrôlable. Une fois validées, les transitions entre ces poses clefs sont créées et intégrées, toujours dans un souci de cohérence avec l'ensemble du jeu.

---

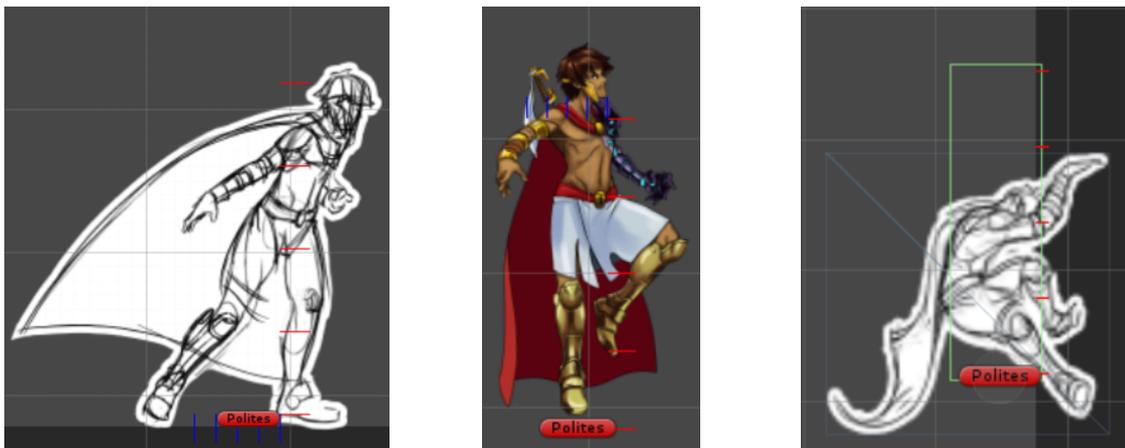
<sup>5</sup> Qui ont tous à un niveau différent un aspect mécanique, non organique.

<sup>6</sup> Ces transitions étant des interpolations entre positions il est impossible d'interpoler deux sprites

C'est là que la mission principale de mon stage a commencé, en effet lors de mon arrivée le **système d'animation par cut out n'était pas encore implémenté** pour le personnage de Polités. Un des désavantages de Spine lorsqu'utilisé avec Unity est qu'il **force son propre système de lecture d'animations** empêchant l'utilisation du système d'Animator, il fallait donc trouver une structure de code qui soit à la fois robuste mais aussi assez flexible pour **pouvoir grossir avec le projet** et le nombre d'animations à gérer<sup>7</sup>.

## ▪ UNE PHYSIQUE SPÉCIALE

L'écriture de cette structure a commencé par la **réécriture d'une partie du code legacy** gérant la physique arcade du jeu, afin de découpler la logique de mouvement des inputs et de l'animation. La physique de Ilios betrayal of gods est une « physique » non réaliste **basée sur des raycast**, des rayons partant de l'intérieur de la bounding box des personnages et un algorithme de correction de position utilisant les informations issues des raycasts est utilisé pour empêcher les objets de se traverser. Le système mis au point reprend ce grand principe et y ajoute une gestion de la vitesse des objets inspirée des forces des rigidbody, une **approximation de la friction** et une structure de code se voulant **la plus modulaire possible**. Plusieurs améliorations ont été apporté sur la façon de gérer des masques de collisions et les one-way platform<sup>8</sup> tout en restant le plus fidèle possible au code de base.



Raycasts servant a la detection de collisions et bounding box

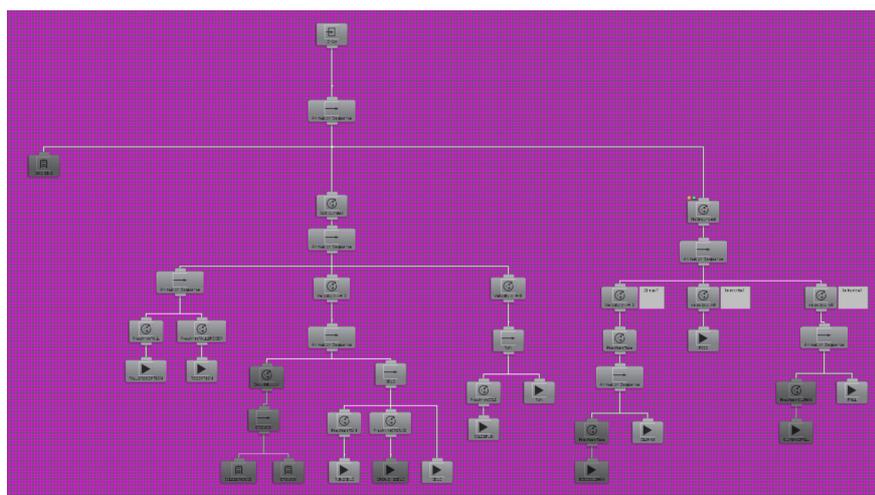
<sup>7</sup> Lors de cette première passe d'intégration la famille « mouvement » comportait une trentaine d'animations différentes sans compter les transitions.

<sup>8</sup> Plateformes que le personnage ne peut traverser que dans un sens, il peut par exemple passer du dessous mais pas en venant du dessus.

## ▪ DES ANIMATIONS À LA PELLE

La deuxième partie de mon travail et la plus complexe était la mise en place d'**une nouvelle structure permettant la gestion de toutes les animations et leurs transitions**. La première approche retenue était l'approche naïve, une approche non structurée afin de mieux me rendre compte des problèmes à résoudre. Parmi les spécifications du système nous pouvons trouver des animations qui peuvent être interrompues sous certaines conditions, des animations qui influent sur la façon de contrôler le personnage et des animations ne pouvant être déclenchées sauf situations spéciales.

La deuxième approche que j'ai suivie était très **inspiré du monde de l'IA**, les animations pouvant être vues comme les noeuds d'un GOAP<sup>9</sup> avec des conditions de réalisation et une action effectuée lors de la validation de ces mêmes conditions. J'ai commencé par les décrire sous cette forme, avant d'ajouter une **phase d'optimisation en transformant une liste de noeuds en arbre** afin de factoriser certains tests. J'ai ensuite testé ce système en modifiant un plugin unity permettant de créer des arbres avec un éditeur visuel. Ce système répondait bien pour le choix d'animations mais la **complexité de l'arbre explosait** lors de l'ajout de noeuds dépendant de l'état du personnage. **Après réflexion sur l'échec de cette structure** plusieurs points ont émergés, il était important de factoriser le choix des animations selon l'état du personnage, les animations des ennemis ne peuvent être gérées avec le même système que celui de Polîtès et une mise en relation inputs/animations est à un certain niveau obligatoire.

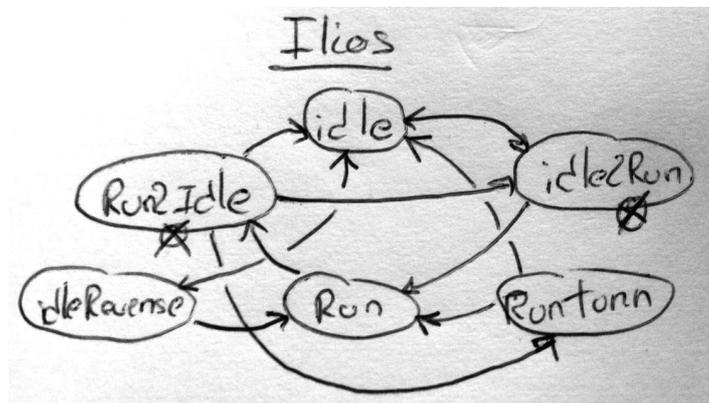


Arbre de la moitié des animations clefs

---

<sup>9</sup> Goal Oriented Action Planning

La mise en avant des relations entre « états » du personnage m'a permis de comprendre qu'un **système de type FSM<sup>10</sup> serait plus adapté** à la fois pour la différenciation du comportement de Politès selon son état<sup>11</sup> et pour séparer les animations en sous familles réparties entre ces mêmes états. **L'implémentation de cette structure s'est faite au dessus de celle du moteur physique**, ce dernier possédant un système de plugin il suffisait d'y relier une FSM modifiée pour la rendre fonctionnelle. Grâce à la FSM, le système choisit un état principal qui gère la mise à jour des inputs, modifie la vitesse de l'objet selon ces mêmes inputs, analyse la situation de Politès pour choisir la bonne animation à jouer et gère la transition vers un nouvel état selon certaines conditions. **Cela permet donc à la fois une modularité bienvenue du code** et des comportements tout en gardant sous contrôle les interactions intra et inter états.



Animations de l'état « Au sol - course »

## ▪ LA SUITE DU STAGE

Aujourd'hui ce système d'animation est en test mais a déjà donné quelques bons résultats, nous sommes actuellement en **pleine session d'itération sur les animations** afin d'affiner les transitions et de valider la structure. Une fois le système validé plusieurs points resteront à approfondir, **implémenter les états manquants**, créer une documentation solide et corriger les derniers bugs. **La date de fin de stage restant assez éloignée**, plusieurs missions pourront m'être confiées, dont un **premier jet de l'IA** de quelques ennemis qui seront très certainement fait à l'aide de behaviour trees ou encore l'intégration de mécanismes uniques des niveaux présents dans la démo.

<sup>10</sup> Finite State Machine, autre structure empruntée au monde de l'IA

<sup>11</sup> On ne peut pas exemple se baisser en l'air ou il est impossible de courir accroché à un mur.

## 3. LA FIN DE L'AVENTURE

Même si aujourd'hui mon stage n'est pas encore arrivé à son terme, il fût riche d'enseignements. J'ai beaucoup appris au contact du lead programmeur David Deckeur au sujet du rendu 3D, de la structure de code et de la vie d'un projet. Mais j'aurais aussi vécu la création d'un studio de jeu indépendant, compris l'aventure humaine et matérielle qu'une telle création d'entreprise représente et aurais vécu au contact d'autres entreprises du milieu connaissant les mêmes joies et problèmes. Et c'est de tout mon coeur que je souhaite bonne chance a Aurora Studio et a leur membres pour la suite de leurs projets.



# 4. ANNEXES

- ILIOS



YOU HAVE: 403

POSEIDON TRIDENT

DIVINE ASCENSION

ATTACK LIFTING ENEMIES FROM THE GROUND. DIVINE ASCENSION PROJECT YOUR OPPONENTS IN A WAVE OF FURY. WHEN AT RANGE OF ARIAL ATTACKS.

COST: 2500

RB + Y

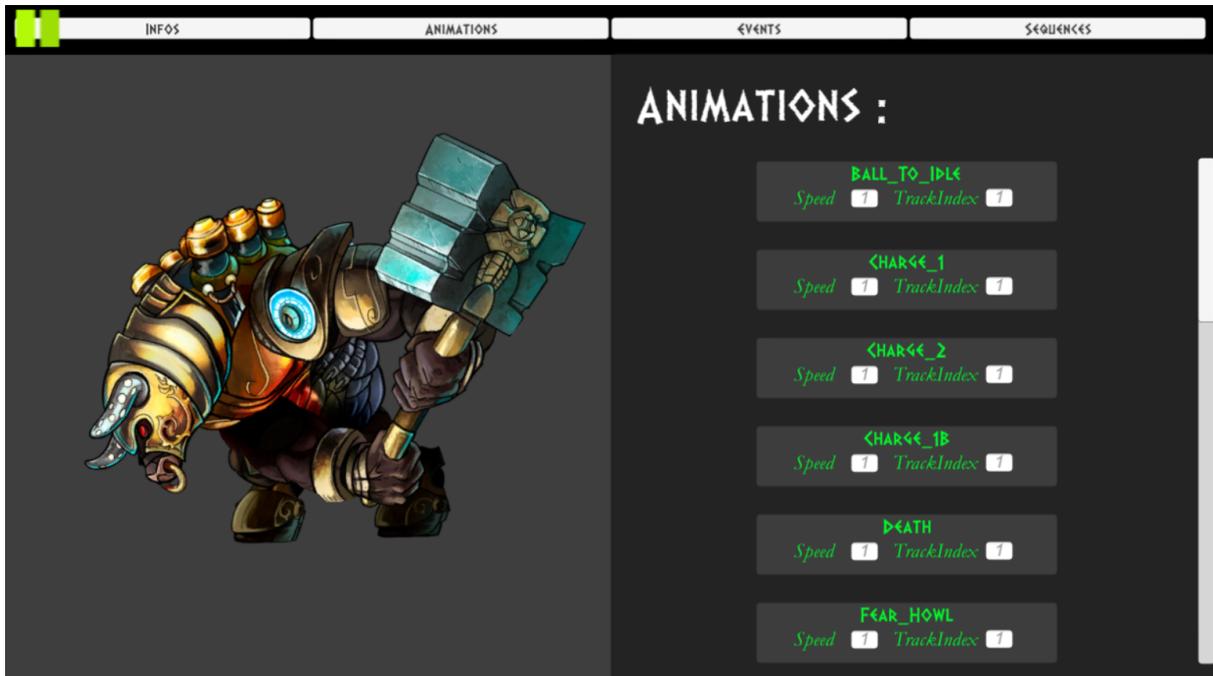
A ACCEPT B DECLINE

ILIOS  
BETRAYAL OF GODS

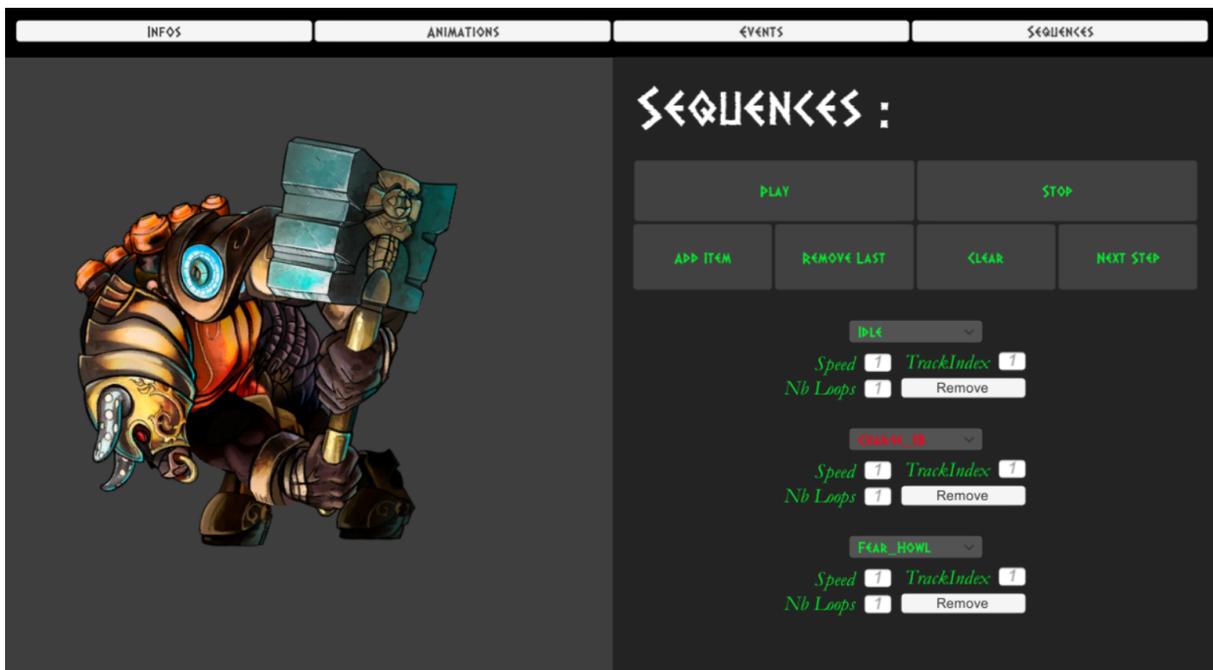




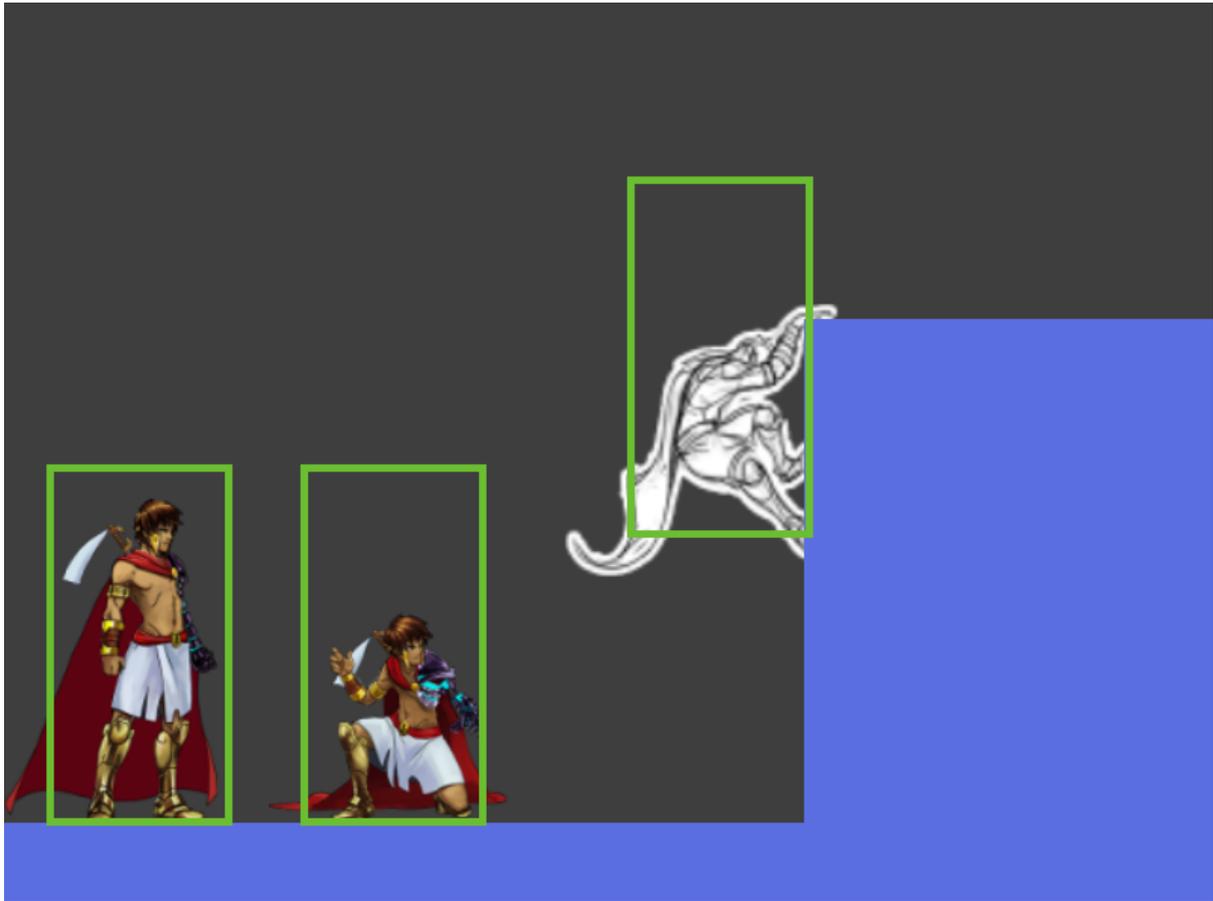
## ▪ OUTILS



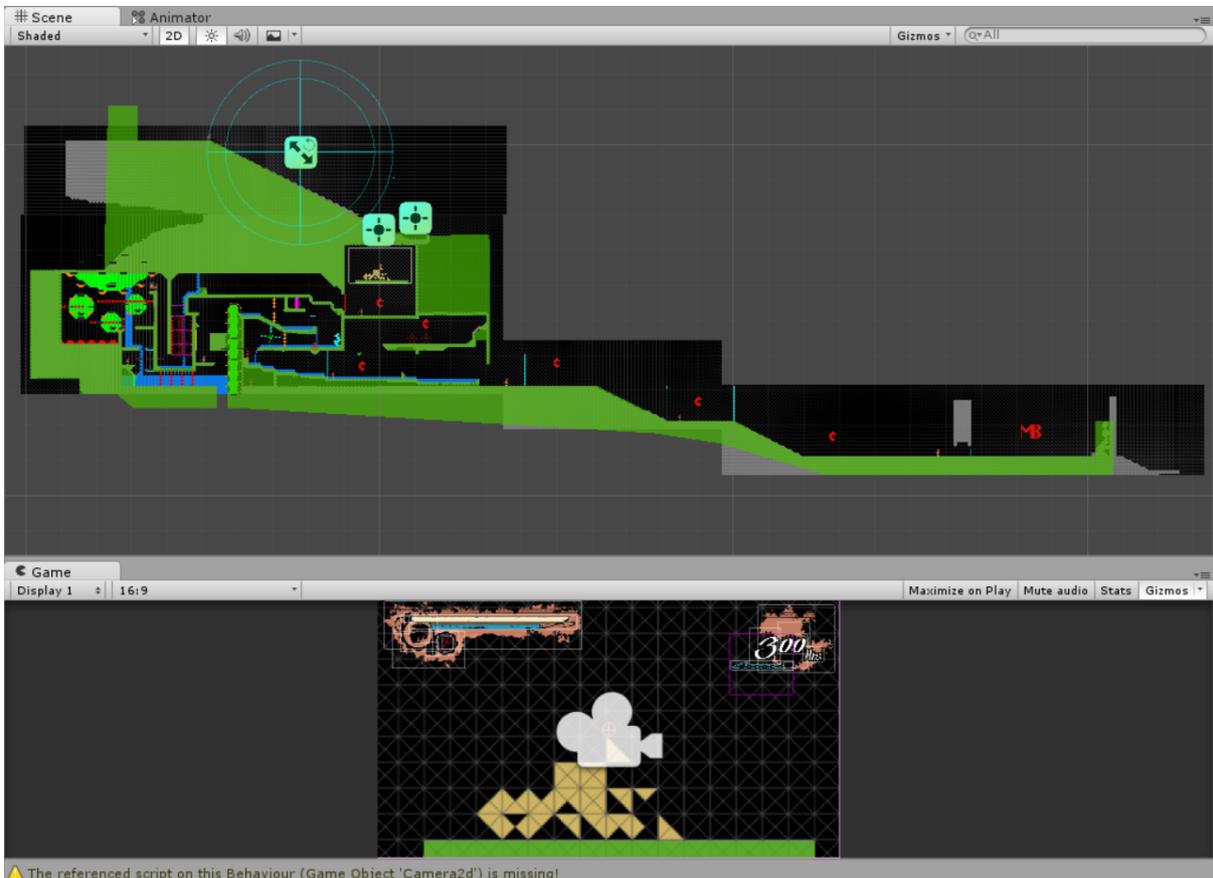
Visualisateur d'animations Spine



Visualisateur d'animations Spine - Séquenceur



Bounding box de Politès



Exemple de level design en cours