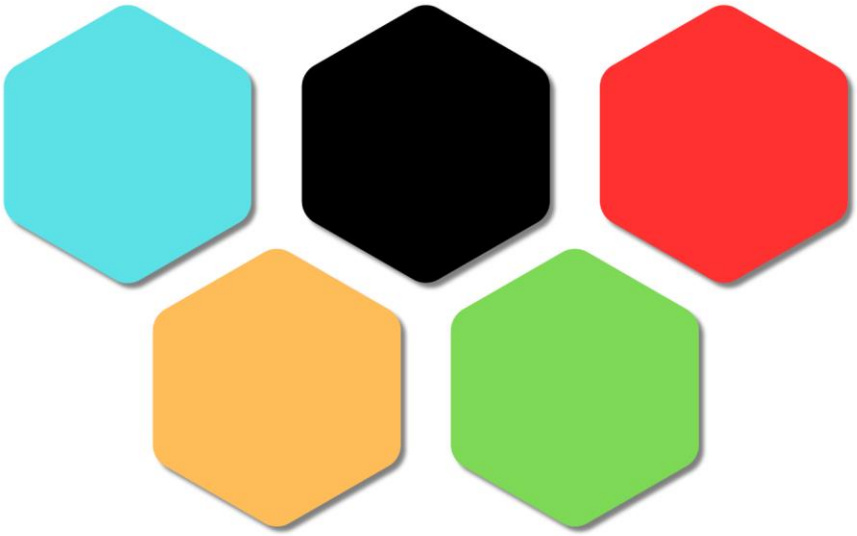


OLYMPUS TRIBUTE

FERTIG

Plan

I. Planning de réalisation
II. Présentation de notre travail
    I. Multijoueur
    II. ECS
    III. Grid
    IV. Menu
    V. Bâtiments
    VI. Assets
    VII. Site web
III. Conclusion
    I. Ce qui est fait
    II. Ce qui est à faire

FERTIG

I. Planning de réalisation

	Irulan	Fanny	Eliott	Gaël	Titouan
Assets					
Multijoueur					
Grid					
Site web					
ECS					
Menus					
Buildings					
Playtests					
Gestion de ressources					
Système d'attaques					
Son					

FERTIG

II. Présentation de notre travail

I. Multijoueur

- Solution principale : Steamworks
- Quelques extraits de code :

```
ChatMessageGameAction message = new("Client sent hello!");
connection.Send(message);
```

```
var memory = new MemoryStream();
var writer = new BinaryWriter(memory);
writer.Write(_registry.ToId(action.GetType()));
action.Encode(writer);
var data :byte[] = memory.ToArray();

SendDataThroughConnection(data);
```

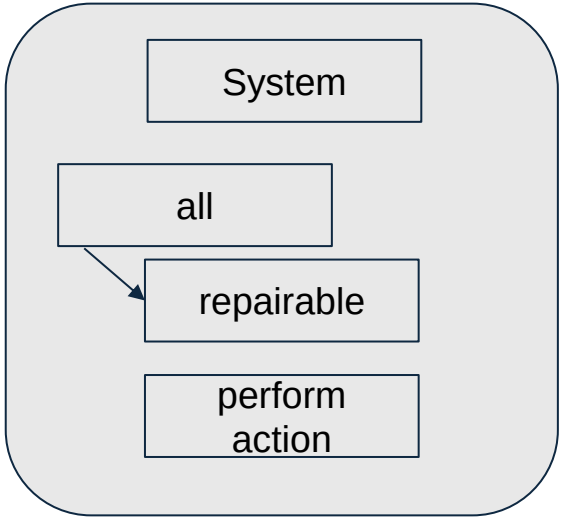
```
connection.AddListener<ChatMessageGameAction>(action =>
{
    Console.WriteLine("Received " + action.Message);
});
```

FERTIG

II. Présentation de notre travail

II. ECS

Principe



FERTIG

II. Présentation de notre travail

II. ECS

Archetypes

Archetype exemples

agora
- destroyable
- repairable
- attack target

gold node
- extractable
- resource type

habitation
- destroyable
- repairable
- attack target
- population contributor

Archetype
- component[][]
- BitSet signature

FERTIG

II. Présentation de notre travail

II. ECS

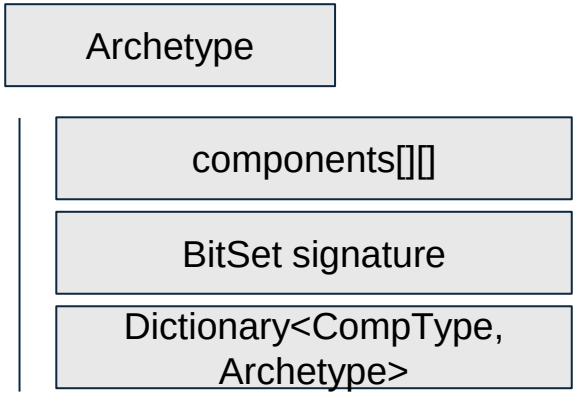
Entities

FERTIG

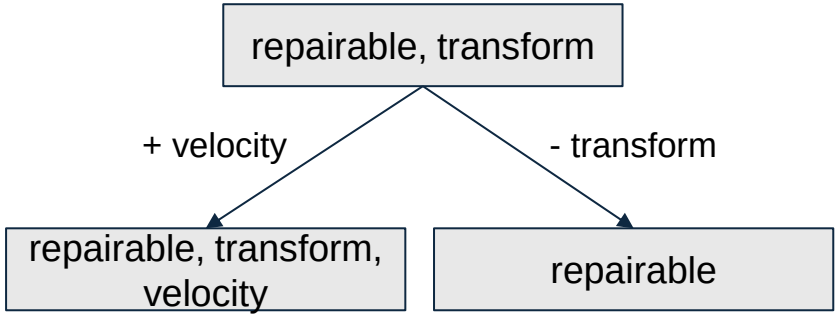
II. Présentation de notre travail

II. ECS

Modification des components



Graphe

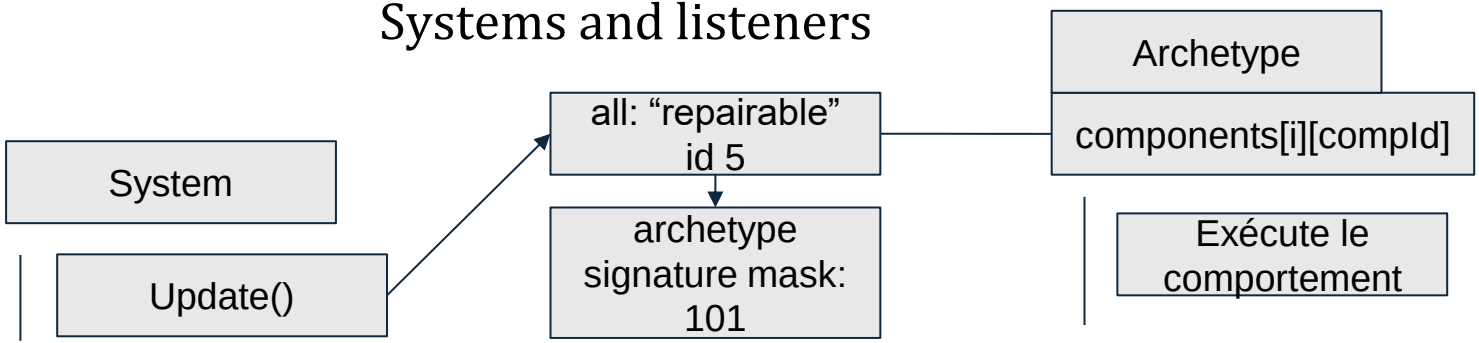


FERTIG

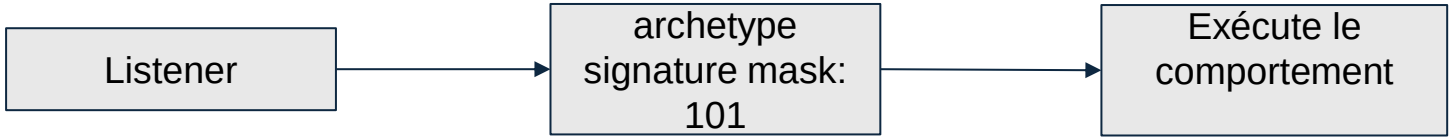
II. Présentation de notre travail

II. ECS

Systems and listeners




```
return OpenBitSet.andNotCount(requiredComponents, signature) == 0 && OpenBitSet.intersectionCount(excludedComponents, signature)==0;
```



FERTIG

II. Présentation de notre travail

II. ECS

Assignation d'une id aux components

```
var componentTypes = new ComponentTypes();
componentTypes.RegisterComponent<RealComponent>();
componentTypes.RegisterComponent<TestComponent>();
```

Ajout d'une entité

```
var engineEntity = new EngineEntity(entityManager);
entityManager.AddEntity(engineEntity, components: new IComponent[] { new RealComponent(), new TestComponent( hello: 4) });
```

Par entité

```
var component = engineEntity.GetComponent<TestComponent>();
Console.WriteLine(component.Hello);
```

Par components

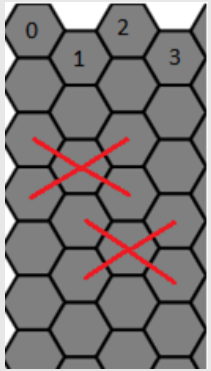
```
var componentQuery = new EngineOneComponentQuery<TestComponent>(componentTypes);
entityManager.AddQuery(componentQuery);

foreach (var (comp :TestComponent , entity) in componentQuery.GetComponents())
{
    Console.WriteLine(entity + " - " + comp.Hello);
}
```

FERTIG

II. Présentation de notre travail

II. Grid

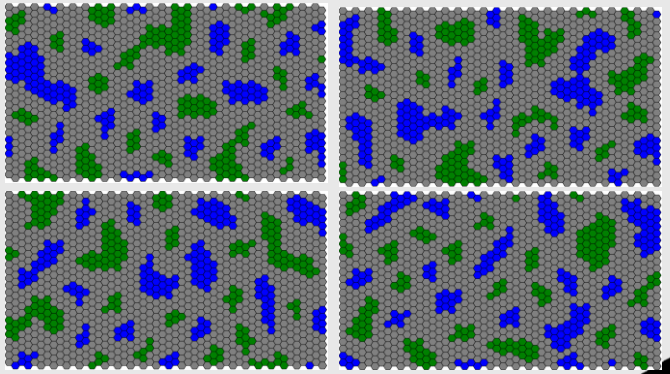
Système de coordonnées



Colonne pair :
→ (x - 1,y - 1)
→ (x - 1,y)
→ (x + 1,y - 1)
→ (x + 1,y - 1)

Colonne impair :
→ (x - 1,y)
→ (x - 1,y + 1)
→ (x + 1,y)
→ (x + 1,y + 1)

Génération procédurale



FERTIG

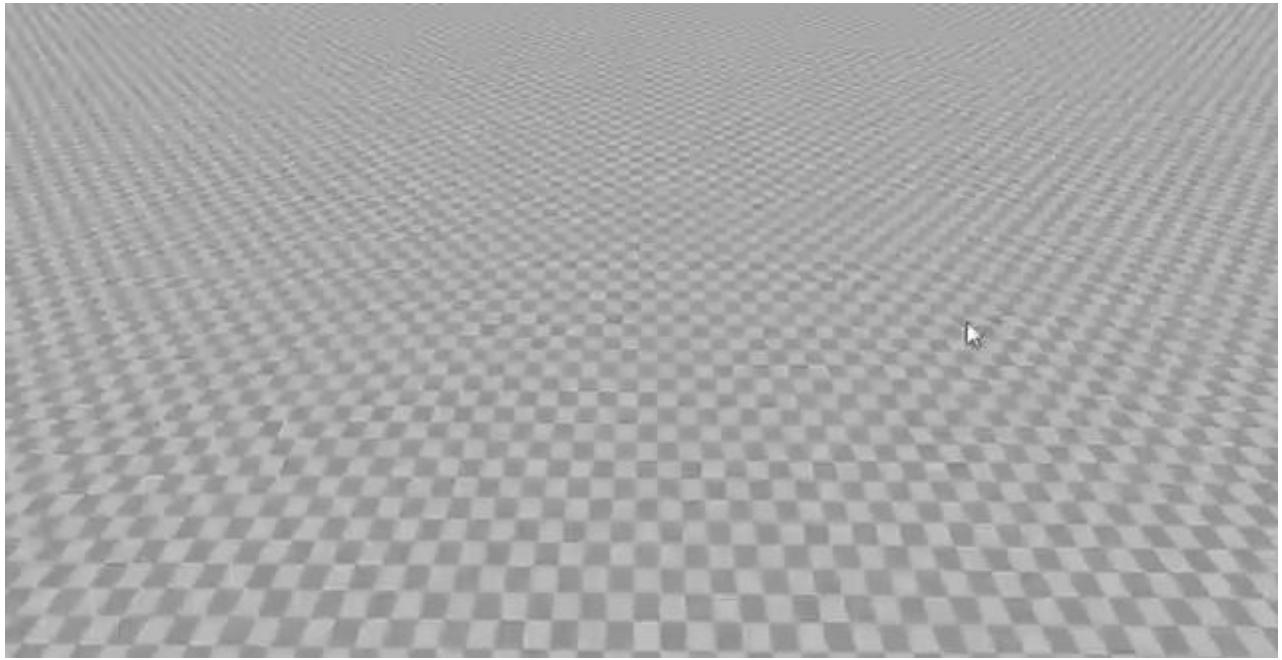
II. Présentation de notre travail

IV. Menu

FERTIG

II. Présentation de notre travail

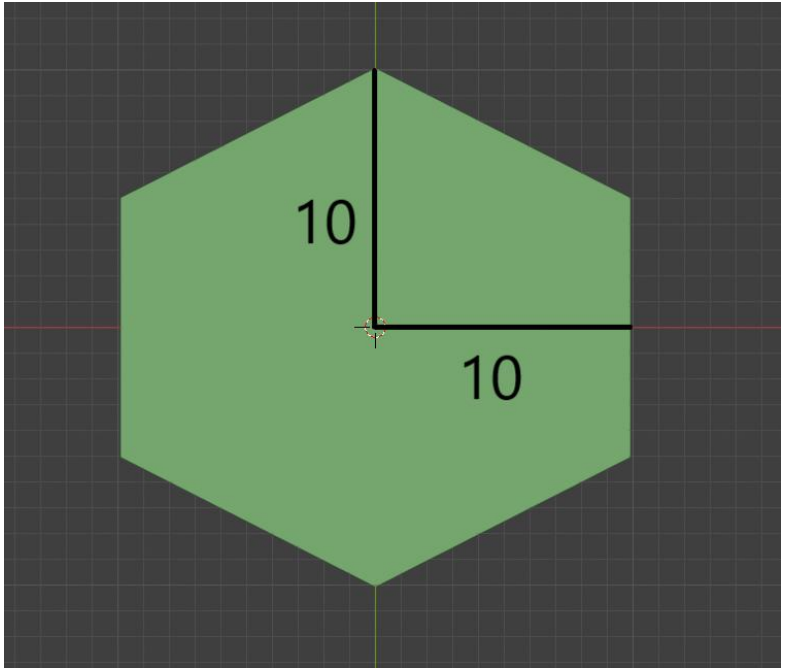
V. Bâtiments



FERTIG

II. Présentation de notre travail

VI. Assets



Hexagone irrégulier de 20x20

FERTIG

II. Présentation de notre travail

VI. Assets

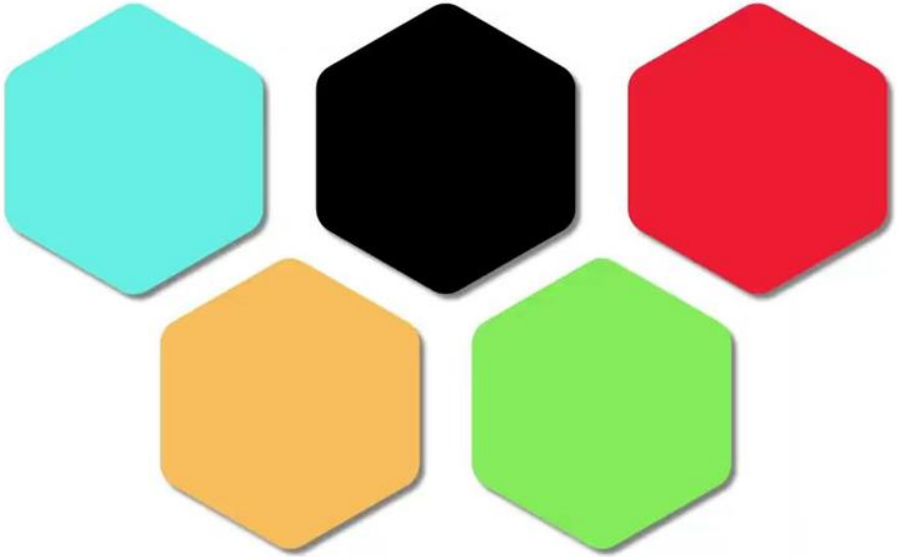
Rendu 3D des assets (bâtiments et ressources)

FERTIG

Menu

FERTIG
Membres
- Gaël Gentric
- Eliott Lukaszczyk
- Irulan Sanz
- Titouan Simon
- Fanny Zhu

Olympus Tribute
- Lore
- Inspirations
- Téléchargements



OLYMPUS TRIBUTE

FERTIG

Menu

FERTIG
Membres
- Gaël Gentric
- Eliott Lukaszczyk
- Irulan Sanz
- Titouan Simon
- Fanny Zhu

Olympus Tribute
- Lore
- Inspirations
- Téléchargements

Téléchargements

Jeu

- Git Hub

Cahier des charges

- Cahier des Charges Fonctionnel
- Cahier des Charges Technique

Soutenances

- Soutenance du 16/12/2024

FERTIG

Conclusion

	Taches	Durée totale	Oct	Nov	Dec	Jan	Fev	Mar	Avr	Mai
Grid		35 Heures								
	Système de coordonées hexagonales									
	Génération procédurale									
	Stockage des tiles									
	Mutation des tiles									
Ressources		65 Heures								
	Compteur de ressource									
	Gestion des ressources									
Attaques		65 Heures								
	Système de séléction de target									
	Réalisation de l'attaque									
	Mutation des états de batiments									
Multijoueur		60 Heures								
	Joueur contre joueur									
IA		60 Heures								
	Système de difficulté variable									
	Adaptation au joueur et cohérence de jeu									
Menus		45 Heures								
	Menu principal									
	Menu sélection de bâtiment									
	Menu du bâtiment d'attaque									
Assets		70 Heures								
	Modèles 3D des batiments									
Site Web		25 Heures								
	Création du layout du site									
	Style									
Audios		15 Heures								
	Création du soundtrack du jeu									
Playtest		20 Heures								
	Test du jeu									
ECS		10 Heures								
	Mise en place de Unity DOTS									

FERTIG

Merci de votre écoute 👍

FERTIG

Sources :

- Site web : https://olympus-tribute.github.io/
- ECS:
  - https://ajmmertens.medium.com/building-an-ecs-1-where-are-my-entities-and-components-63d07c7da742
  - https://devlog.hexops.com/2022/lets-build-ecs-part-1/
  - https://github.com/skypjack/entt
- Multiplayer:
  - documentation steamworks: https://partner.steamgames.com/doc/sdk

FERTIG