PARTIE 1 : FX

Show : MATILDA

Timecode : 6s

Water simulation et ripples

Simulation d'eau en flip, avec interaction des gouttes tombant des pics.
Principe pour l'interaction :
Je crée en priorité la simulation principale (eau de la piscine)
Ensuite dans un souci d'optimisation je crée une simulation des gouttes qui tombent sur une surface plane et crée des ripples (simulation efficace et très légère)
Il ne me reste plus qu'à créer un script VEX qui me permet de transférer les déformations des ripples vers le cache de ma simulation principal.

Show : MATILDA

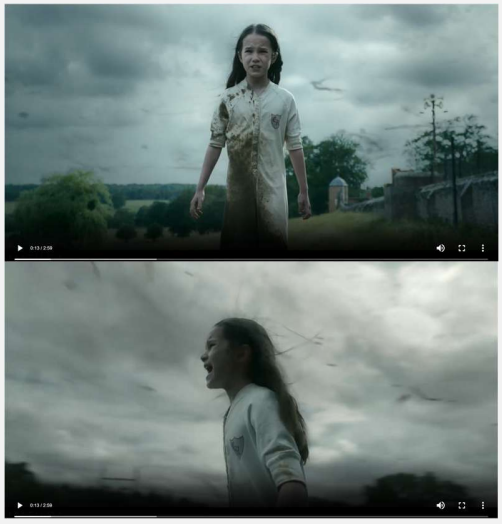
Timecode : 8s

Water simulation et splashes

Pour ce shot, le principe d'interaction est un peu différent car les splashes sont beaucoup plus conséquents que de simples ripples.

Principe :
La première étape est de créer les splashes seuls qui vont nous permettre de générer des vélocités.
Ensuite je crée la simulation de la piscine à laquelle j'ajoute la vélocité des splashes, ce qui va me permettre de créer des remous dans l'eau.

Ce procédé permet d'alléger considérablement la scène, et d'augmenter la résolution des splashes afin d'avoir un maximum de détails.



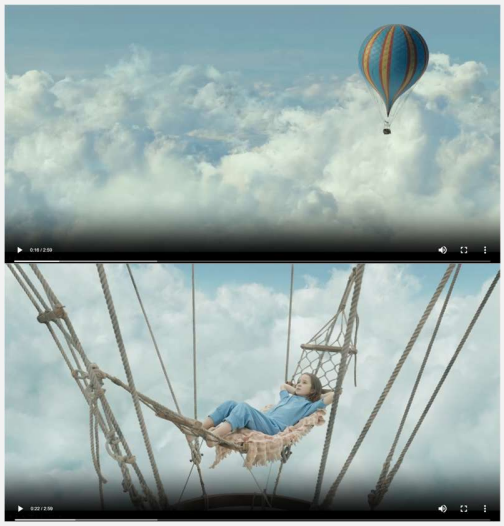
Show : MATILDA

Timecode : 10s

Tornado

L'élément le plus important de cet effet et qui m'a permis de driver tous les éléments est un **vector field** dont les vecteurs de vélocité forment une spiral.
Je connecte donc ce **vector field** aux différentes simulations de l'effet, à savoir :

- **Dust** : la poussière qui forme la forme générale de la tornade
- **Débris** : qui est une simulation de particules auxquelles je viendrais instancier des débris (branches et feuilles créés par le département '**Assets**' (modeling)
- **Grits** : petits graviers qui enrichissent l'effet



Show : MATILDA

Timecode : 15s

Clouds

Pour cet effet j'ai créé quelques nuages (VDB) puis créé quelques sets que j'ai exporte en deux versions :

- Une version VDB
- Une version proxy GEO (très light)

La version proxy GEO a été récupérée par le département environnement qui a créé le layout final.
La version VDB est récupérer par le département lighting.
Enfin, grâce à un script automatique dans Gaffer (équivalent de Katana) les versions de sets en proxy GEO du layout sont remplacée par les versions VDB.

Show : THE MIDNIGHT SKY

Timecode : 23s

Neige rampante au sol

Pour cet effet, des vecteurs sont placés au sol créant un **vecteur field** qui permet de driver la neige qui est constituée par deux éléments :

- Un volume
- Des particules

Une force (vent) avec un noise appliqué viendra rendre l'effet plus riche et intéressant.

Show : SPIDERMAN NO WAY HOME

Timecode : 29s

Smoke à l'intérieur du portail

Effet de smoke tournant en spiral sur les parois intérieures du portail.
Afin de créer le vector field qui me permettra de driver la smoke, j'ai récupéré le système de particules du portail que mon collègue a créé auparavant.
C'est ce qui a permis d'avoir une smoke qui interagis d'une manière cohérente avec les mouvements du portail.

Show : SKYSCRAPER

Timecode : 31s

Bullets hits sur des écrans TV

Cet effet est décomposé en 3 parties :

- Les craquelures sur l'écran
- De la smoke tombante, à l'endroit de l'impact

Des debris représentant des morceaux de verre qui volent.

Show : SKYSCRAPER

Timecode : 37s

FIRE

Les flammes les plus conséquentes sont des caches que j'ai positionnés manuellement.
Les petits débris enflammes sont des simulations.

Show : PAWS OF FURY

Timecode : 40s

River, water simulation

La simulation d'eau, avec les éclaboussures et la white water, a été utilisée dans 37 plans différents. Cette simulation couvrent une grande distance, car certains angles de caméra montrent la rivière sur une grande profondeur.

Show : PAWS OF FURY

Timecode : 52s

Destruction (piano qui passe a travers le toit)
Interaction piano/sol

Concernant la destruction du toit, l'asset original ne comportait pas d'éléments de structure porteuse interne destinée a soutenir le toit. Par soucis de gains de temps (qu'il aurait fallut pour renvoyer l'asset au modeling/texturing) j'ai créé quelques poutres et tasseaux en bois afin d'avoir plus de matière dans les débris qui volent dans les airs.
Afin d'obtenir une texture cohérente, j'ai fait correspondre les UV de ces poutres a des éléments de bois semblable.
La destruction du toit est donc décomposée en deux parties, a savoir :

- Débris
- Poussière

L'interaction avec le sol quant à elle se décompose de la manière suivante :

- Poussière d'impacte
- Poussière qui s'accumule derrière le piano lors du déplacement de celui-ci
- Déplacement du sable que le piano creuse lors du déplacement également.

Show : PAWS OF FURY Timecode : 1m00s

Balles de fusil

Pour réaliser cet effet, je suis partie d'une particule dont la source est positionnée à l'intérieur du canon ou a l'intérieur de la bouche pour le personnage de la seconde image. J'ai donné une vélocité parallèle au canon, et dirigée vers un point concernant la seconde image. Le système de particule était réglé de manière a avoir une particule toute les trois frames. Pour la forme de la balle, j'ai créé une forme en polygone de la forme exacte que je voulais (c'est-à-dire d'une forme ovale suivi de la trajectoire rectiligne)
J'ai ensuite instancié la géométrie sur chaque particule, donnant l'impression d'une balle projetée par le canon.
Concernant le shader, j'ai transformé la géométrie en VDB (field de température), puis, j'ai créé un « standard volume » dans Arnold qui m'a permis de récupérer mon field de température et de lui appliquer la couleur que je voulais.

Show : JUSTICE LEAGUE Timecode : 1m07s

Downwash et debris

Pour cet effet, des vecteurs sont places au sol créant un **vecteur field** qui permet de générer un volume qui donnera l'impression d'être propulse par le vaisseau. L'effet se décompose de :

- Un volume
- Des particules instanciées de différents débris
- Des petits gravillons (particules instanciées également)

Show : JUSTICE LEAGUE Timecode : 1m13s

Downwash et debris

Suite du précèdent plan qui se décompose de :

- Un volume pour le downwash suite au décollage du vaisseau

Feuilles qui volent (particules instanciée)

Show : JUSTICE LEAGUE

Timecode : 1m17s

Cendres et braises

Particules instanciées.

La principale exigence de cet effet est que les particules devaient maintenir une trajectoire ascendante constante et ne jamais descendre.

Show : RIVER DANCE

Timecode : 1m19s

Pluie et interraction

Système de particules avec interaction sur le sol et les différents éléments tel que les parapluies.

Show : RIVER DANCE

Timecode : 1m24s

Bonbons volants

Pour obtenir l'effet désiré, j'ai créé une simulation RBD des poissons dans le bocal en leur assignant un point de destination. Cependant, après avoir exécuté la simulation, j'ai remarqué que les poissons sortaient du bocal mais ne semblaient pas animés (c'est-à-dire qu'ils ne nageaient pas). Pour remédier à cela, j'ai créé trois courtes animations différentes au centre du monde. À l'aide d'un script vex, j'ai programmé les poissons pour qu'ils s'animent progressivement en sortant du bocal. Ils choisissent aléatoirement l'une des trois animations, ce qui leur donne un aspect varié (un peu a la manière de blenshape).

Show : RIVER DANCE

Timecode : 1m27s

Dance sur l'eau

La surface de l'eau est un « ocean spectrum ».
Principe pour l'interaction avec les gouttelettes et les pas dans l'eau :
J'ai dans un premier temps créé un Flip simulation pour les gouttelettes qui se forment sous les pieds.
Ensuite j'ai créé des ripples dont la source sont les pieds et les gouttelettes touchant la surface de l'eau.
Puis a l'aide d'un script VEX qui me permet de transférer les déformations des ripples vers le cache de « l'ocean spectrum ».

Show : RIVER DANCE

Timecode : 1m36s

Coup de fusil

Deux layer dans cet effet :

- Feu
- Fumée

Le feu du fusil a été créé à l'aide de particules qui sont projetés à partir du fusil et ont été transformées en vdb (field temperature).
Puis, à l'aide d'un « standard volume Arnold », j'ai pu plugger le field de température et ainsi lui appliquer une rampe de couleurs.

Show : RIVER DANCE

Timecode : 1m39s

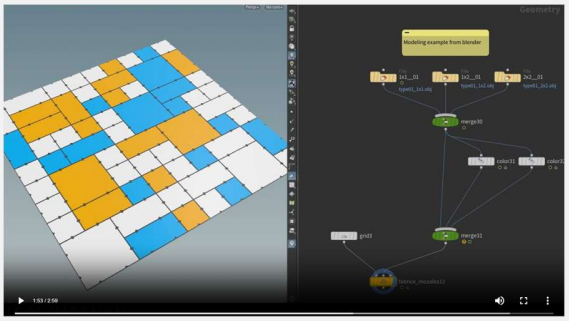
Sculpture dans un tronc d'arbre

Pour expliquer la scène, les oiseaux sont en train de sculpter un tronc d'arbre pour créer une crosse de hockey et un casque pour un des personnages de l'histoire.

L'effet a été décomposé en trois layer :

- Copeaux de bois (particules instanciées)
- Poussière (particules)
- Poussière (volume)

PARTIE 2 : PROCEDURAL MODELING



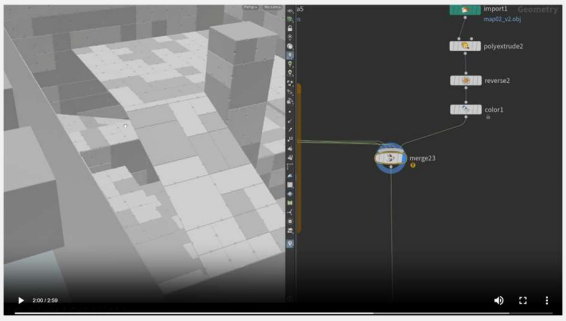
Tool : MOZAIKA

Timecode : 1m51s

Ce tool permet d'habiller une surface plane. Cela peut être un mur, un sol, un plafond, que ce soit un vaisseau, un château… en fait, toutes les surfaces qui nécessitent d'être habillées.
Le principe est simple, imaginons que nous voulons créer un habillage de sol avec des carreaux de tailles différentes, nous commençons par créer tous les éléments dont nous avons besoin, c'est-à-dire (et pour reprendre l'exemple illustre ci-contre) :

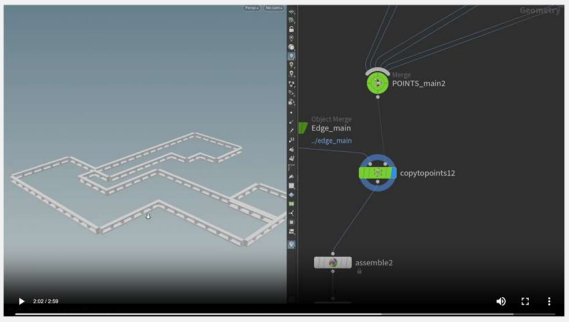
- un carreau de 1x1u
- un carreau de 1x2u
- un carreau de 2x2u

Nous avons dans ce cas-ci, 4 éléments à brancher sur notre tool, les 3 carreaux et la grille destinée à être habillée.
Mozaika se charge alors de remplir la grille en positionnant les éléments a la manière d'un Tetris. Il réagit en temps réel, c'est-à-dire que si nous lui injectons d'autres éléments (par exemple dans l'illustration des éléments de couleurs différentes, il refait un habillage instantanément.
Parmi les paramètres que nous pouvons modifier, nous pouvons faire varier l'échelle globale de tous les carreaux en entrant un coefficient de scale, nous pouvons faire varier l'épaisseur des joints entre éléments, nous pouvons faire tourner l'ensemble des éléments en lui donnant une valeur d'angle.



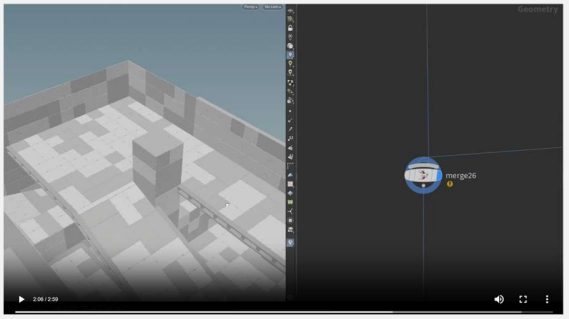
Timecode : 1m58s

Dans l'illustration suivante, j'ai créé une petite map de jeu afin de pouvoir tester le tool. C'est une map sur 3 niveaux de sol. J'ai donc procédé par niveau et par type, c'est-à-dire que pour chaque niveau j'ai décomposé murs et sols. J'ai ensuite merge le tout pour former la map.



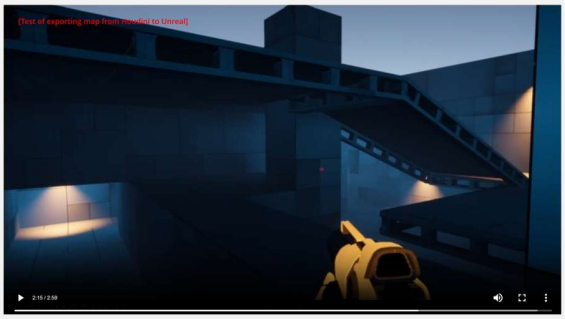
Timecode : 2m01s

Ensuite j'ai applique un autre outil qui permet d'habiller les parois latérales d'un sol, ou plus précisément d'un balcon ou d'une passerelle afin de lui donner une épaisseur. Ici ce sont des éléments métalliques formant une poutre qui sont créés et ajouté à la map.



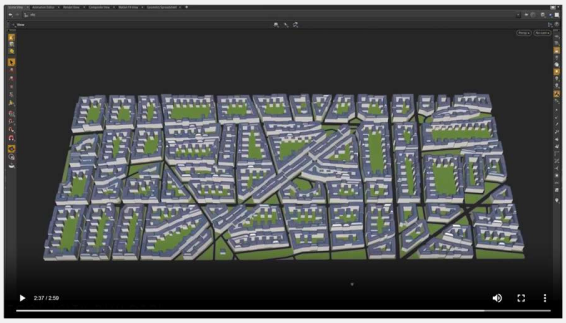
Timecode : 2m06s

Résultat avec les poutres et l'habillage des murs et des sols.



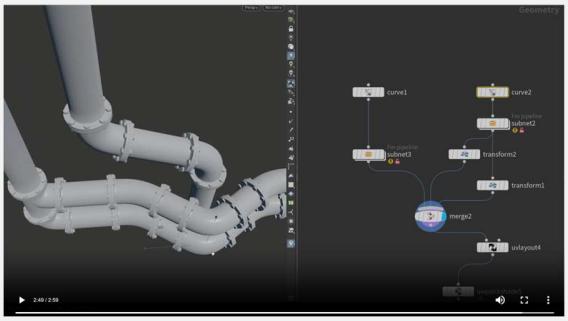
Timecode : 2m08s

J'ai ensuite fait un test d'export de la map vers Unreal



TOOL : **CITY BUILDER** **Timecode : 2m16s**

Ce tool permet de générer des villes à partir de splines, c'est-à-dire que nous dessinons des splines qui représentes les rues, et l'outil va créer dans un premier temps tous les trottoirs et terrains, puis va ajouter les bâtiments avec chacun des caractéristiques différentes, comme par exemple le nombre d'étage, le type de toiture … etc …



TOOL : **PIPE** **Timecode : 2m40s**

Outil permettant de créer un réseau de pipeline a partir de spline. Il va détecter les angles et ainsi créer des éléments séparés par exemple les coudes.