

Introduction aux nœuds de géométrie (Geometry Nodes) dans Blender 2.92 - Tutoriel 2 sur 2



Les modèles 3D utilisés dans ce tutoriel :

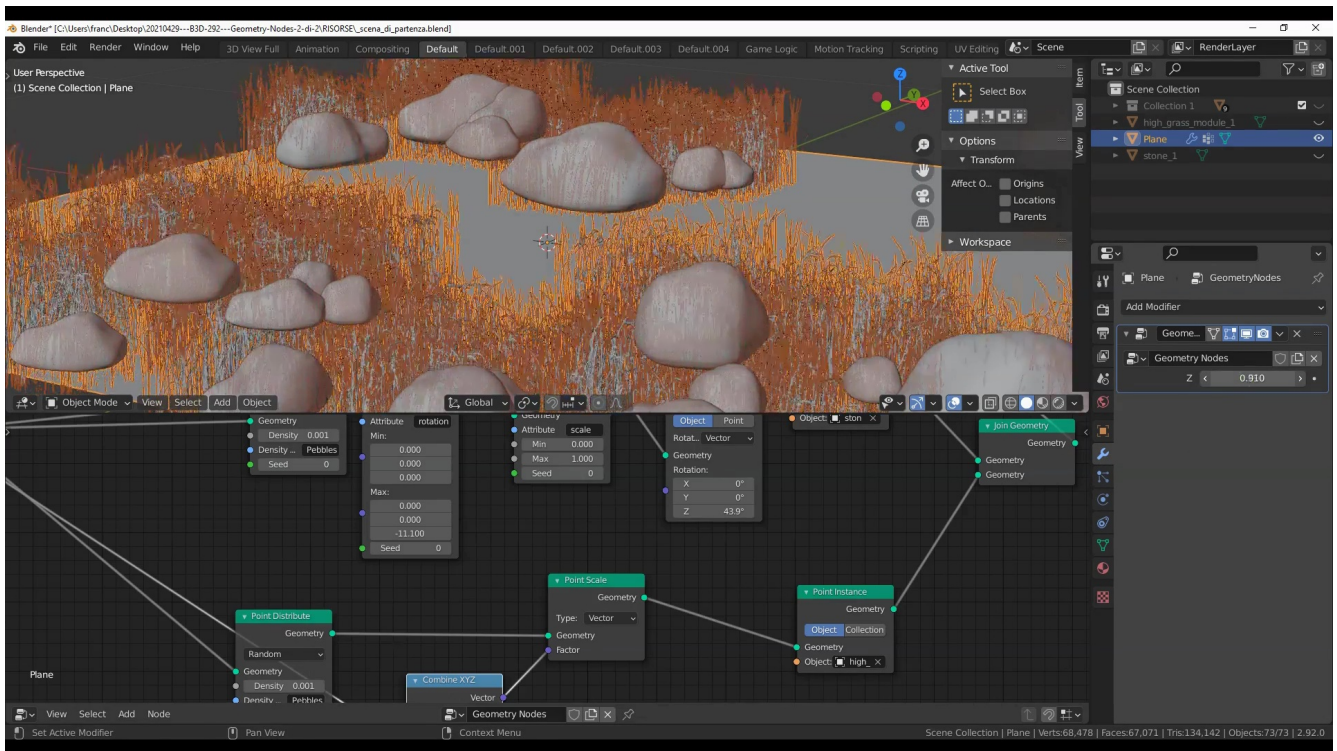
- Pebbles and stones set (<https://sketchfab.com/3d-models/pebbles-and-stones-set-8356674252f64a15ae329e1ecc78beeb>)
- High grass module (<https://sketchfab.com/3d-models/high-grass-module-e218da5203924b5d9ebd145ab0b11b2d>)

Bonjour à tous!

Ceci est le deuxième des deux didacticiels vidéo d'introduction sur le modificateur Geometry Nodes dans Blender 2.92.

Dans le didacticiel précédent, nous avons examiné les nœuds du groupe Geometry (c'est-à-dire Transform et Join, pour modifier et fusionner les géométries dans le modificateur); la possibilité de créer des sockets d'entrée pour Group Input afin de spécifier des paramètres dans l'onglet modificateur, et l'utilisation de certains nœuds Vector et Utilities (Math, en particulier) pour effectuer des opérations mathématiques et combiner des informations.

Dans cette vidéo, nous verrons une utilisation différente des nœuds géométriques, car nous les utiliserons comme systèmes de particules pour la "diffusion", c'est-à-dire la distribution d'instances d'objets à la surface des autres ; Cependant, tous les sujets abordés dans le premier didacticiel vidéo de cette mini-série vous seront utiles ici.



Le modificateur Geometry Nodes permet de transformer la géométrie de l'objet qui l'utilise en un générateur de points, qui seront répartis sur sa surface (c'est-à-dire : sur ses faces).

Ces points peuvent être utilisés pour ranger, dans leur correspondance, des instances d'autres objets ; de plus, les Nœuds Géométriques permettent également d'affecter des transformations (position, rotation et redimensionnement) à chacun de ces points, de manière à introduire l'aléatoire.

En listant ces opérations, j'ai dit trois choses en particulier :

1. transformer la géométrie en un ensemble de points DISTRIBUÉS sur la surface ;
2. INSTANCATION des objets en ces points ;
3. modifier les caractéristiques (ie : les ATTRIBUTS) des points générés et ainsi influencer les instances créées.

Ces trois opérations se traduisent par autant de nœuds géométriques, que nous voyons immédiatement.

Dans la scène visible à l'écran, il y a un Plan par défaut très simple (donc fait de seulement 4 sommets et une face) ; de plus, à l'aide d'APPEND j'ai importé les modèles 3D de quelques cailloux (les « Pebbles »), d'une pierre (« Stone 1 ») et d'un objet contenant des brins d'herbe.

Pour plus de commodité, je regroupe les cailloux dans une collection et cache tous ces éléments accessoires, en les désactivant dans un éditeur Outliner.

Vous pouvez trouver le lien vers ces actifs dans la description de la vidéo ; cependant, comme dans le cas de la montre de la vidéo précédente, il n'est PAS nécessaire d'avoir ces modèles pour suivre la vidéo et apprendre !

Je sélectionne le plan et ajoute un modificateur de nœuds géométriques à l'objet.

Pour effectuer la première opération, c'est-à-dire transformer le plan en un ensemble de points DISTRIBUÉS sur sa surface, j'ajoute un nœud Point DISTRIBUTE au milieu du groupe.

A l'écran on peut voir que le Plan a disparu, remplacé par un ensemble de points... enfin, en réalité cette information de départ n'a pas été perdue, en effet elle est toujours là : la manière non destructive de modélisation des Nœuds Géométriques nous permet pour ajouter un nœud Geometry Join entre Point Distribute et Group Output et pour connecter, à l'une des deux entrées Join, la géométrie provenant de Geometry Input (c'est-à-dire : le Plan d'origine).

On a ainsi à la fois le Plan d'origine (que l'on peut modifier et texturer, par exemple pour lui donner l'apparence d'un terrain), et les points répartis à sa surface.

Le nœud Point Distribute fournit certains paramètres, parmi lesquels Density (la Densité) se distingue, ce qui nous permet d'augmenter ou de diminuer intuitivement la quantité de points à répartir sur la surface.

Dans le didacticiel vidéo précédent, nous avons vu comment connecter une information de nœud à Group Input, afin de pouvoir modifier les valeurs dans l'interface du modificateur ; il peut être judicieux de le faire pour le paramètre

Point Distribute Density.

On passe ensuite à la deuxième opération des trois listées ci-dessus, c'est-à-dire INSTANCER un autre objet (ou une Collection d'objets), en plaçant une copie pour chaque point créé par Point Distribute : on insère alors un nœud Point INSTANCE entre Point Distribute et Join.

Le sens de l'opération est clair : Distribute génère les points et Instance prend les informations de ces points pour créer des copies d'un autre objet... oui, mais quel objet ?

Dans le nœud Point Instance, il y a un sélecteur qui nous permet de définir si nous voulons instancier un seul objet ou une Collection d'objets ; le mode choisi (Objet, dans mon cas), on peut cliquer sur la case vide située à l'intérieur du nœud et choisir l'objet qui nous intéresse (« Stone 1 », dans mon cas).

NOTE IMPORTANTE : pour un résultat correct, assurez-vous d'avoir appliqué les transformations Rotation et Scale des objets que vous souhaitez instancier ; vous pouvez le faire en les sélectionnant et en appuyant sur CTRL A, puis en choisissant « Rotation and Scale » dans le menu Apply qui apparaîtra à l'écran.

Nous ajustons ensuite le paramètre Density de Point Distribute, comme souhaité, jusqu'à ce que nous atteignons un résultat qui nous convient.

La troisième opération fondamentale à faire est de modifier les paramètres des différents points (et donc des instances générées) ; en fait, il est possible d'effectuer différents types d'opérations, nous ne parlons donc pas d'un seul nœud mais des nœuds des groupes Point et Attribute.

Dans les deux cas, les nœuds de transformation seront insérés entre Point Distribute et Point Instance, car ils devront modifier les POINTS créés, AVANT de générer les instances à partir de ces points.

Par exemple, commençons par la rotation, à appliquer à tous les points : nous ajoutons ensuite un nœud Point Rotate entre Point Distribute et Point Instance et modifions les angles de rotation des particules.

Le résultat est déjà meilleur que celui obtenu sans ce nœud, mais comment introduire de l'aléatoire dans la rotation et, pourquoi pas, aussi dans la taille de ces instances ?

Nous sommes aidés par le nœud Attribute Randomize (de "Random": "Aléatoire", bien sûr), que nous pouvons cascader entre Point Rotate et Point Instance (en fait ce flux de traitement s'effectue toujours sur les géométries, comme il est possible de le vérifier en observant le sockets Geometry liés entre les différents nœuds).

Attribute Randomize est un nœud qui permet d'introduire de l'aléatoire sur un ATTRIBUT des points, à préciser précisément dans le champ « Attribute » du nœud.

Parmi les attributs fondamentaux d'une particule nous avons, bien entendu, "rotation" et "scale", commençons donc par écrire "rotation" dans ce champ et modifions la valeur Max...

Le nœud introduit en fait de l'aléatoire dans l'orientation des particules, mais celles-ci tournent autour de tous leurs axes, ce qui n'est pas exactement un résultat intéressant : nous aimerions garder l'orientation de base donnée par Point Rotate et faire tourner les instances aléatoirement uniquement autour de l'axe vertical .

Pour résoudre ce problème, nous devons changer le type de valeur d'opération de Float (qui est une valeur numérique avec une virgule agissant uniformément sur les trois axes XYZ) en Vector, où nous pouvons mettre 0 pour toutes les valeurs à l'exception du MAX de l'axe que nous sommes intéressés à modifier

...

Mais même cette opération pose un problème : nous avons perdu la rotation donnée par Point Rotate !

C'est un problème de priorité, dans un certain sens : entre Point Distribute et Point Instance, en fait, il faut d'abord mettre Attribute Randomize puis Point Rotate.

Maintenant, dupliquez Attribute Randomize et ajoutez-le entre l'original et Point Rotate pour modifier également le paramètre Scale (à écrire, tout comme la rotation, entièrement en minuscules, sinon Blender ne le reconnaîtra pas).

Dans ce cas, cependant, nous devons modifier tous les paramètres min et max XYZ, pour obtenir des résultats intéressants (et peut-être éviter de mettre 0 pour les valeurs minimales, sinon les objets perdront une ou plusieurs dimensions).

Concernant l'attribut Scale, vous trouverez peut-être plus pratique de revenir au type Float, afin de n'utiliser qu'un seul paramètre pour le facteur d'échelle minimum et un seul pour le facteur maximum... à vous de choisir.

Avec seulement ces deux nœuds d'Attribut, cependant, nous obtenons une certaine diversité à partir d'un objet qui était composé d'une seule géométrie : Stone 1.

Si nous changeons le type d'objet à instancier dans Point Instance d'Objet à Collection et spécifions la collection de cailloux, les choses deviennent beaucoup plus intéressantes !

Redéfinissons Object avec la Stone 1 dans Point Instance, cependant, pour examiner une autre fonctionnalité disponible avec les Geometry Nodes.

Dans les systèmes de particules classiques, nous pouvons également définir sur quelles parties de la géométrie créer les particules et sur lesquelles non ; cela se fait notamment avec le Weight Painting, en définissant des valeurs de

densité (de 0 à 1) sur les différentes parties d'une surface... eh bien, on peut le faire ici aussi !

Avant de continuer, cependant, nous devons augmenter le nombre de subdivisions du Plan, car le Weight Painting (à opérer en dehors du Geometry Node, sur la géométrie d'origine) agit sur les sommets, donc les 4 sommets de base du Plan ne sont pas suffisants .

Dans une vue 3D, alors, nous passons en Edit Mode et subdivisons la surface entière un bon nombre de fois ; puis passons en mode Weight Paint.

Par défaut, la surface doit apparaître en bleu, c'est-à-dire « weight 0 » ; le nœud Point Distribute créera les points sur les zones de poids 1, nous devons donc peindre dans les zones sur lesquelles nous voulons faire apparaître les instances.

REMARQUE : si vous souhaitez appliquer les instances partout sauf à des endroits particuliers, vous devez procéder en sens inverse ; dans ce cas, en particulier, nous utilisons le pinceau dans les zones que nous voulons EXCLURE de Point Distribute (et Blender créera automatiquement un Vertex Group, qui devrait avoir « Group » comme nom par défaut, mais nous pouvons également le renommer en quelque chose de plus significatif, comme "Pebbles").

A la fin de l'opération, ouvrez le menu Weights dans la fenêtre 3D et cliquez sur « Invert », pour inverser les poids : maintenant, la zone que nous avons peinte aura le poids 0, tandis que dans le reste de la géométrie nous aurons le poids 1

.

Evidemment, rien ne nous empêche de modifier les poids ultérieurement, en faisant varier la valeur du paramètre **Weights** dans l'onglet **Tool** en mode **Weight Paint** et en brossant la surface ; nous devons juste nous rappeler que nous n'insérerons PAS d'instances dans des zones de pondération 0.

Dans l'éditeur **Geometry Nodes**, on tape le nom du **Vertex Group** dans le champ **Density** du nœud **Point Distribute** ("Pebbles", dans notre cas ; attention aux majuscules et minuscules) : la répartition des points sur la surface d'origine sera proportionnel aux poids de sommet définis par nous.

Avant de conclure ce didacticiel vidéo et donc la mini-série **INTRODUCTION to Geometry Nodes dans Blender 2.92**, voyons un autre exemple d'instanciation d'objet ... mais avec un changement de plus.

Dans le schéma que nous avons défini, nous insérons un nouveau nœud **Point Density**, en le connectant au port **Geometry** de **Group Input**, puis connectons-le à une nouvelle **Point Instance** (vide) et connectons les deux **Point Instances** disponibles dans la scène à une nouvelle Nœud **Join Geometry**, qui à son tour sera lié au nœud **Join** déjà disponible dans la scène ; en pratique, la géométrie résultante sera donnée par celle d'origine jointe à celle arrivant par deux nœuds **Instance**.

Dans le nouveau nœud **Distribute**, nous définissons le même attribut **Density** de "Stone 1", mais dans **Point Instance**, nous définissons l'objet **High Grass** comme celui à instancier.

Si nécessaire, nous pouvons insérer un nœud **Point Rotate** pour orienter correctement les brins d'herbe.

Insérons également un nœud **Point Scale** (entre **Point Rotate** et **Point Instance**), définissant notamment le mode **Vector**, afin que nous puissions spécifier le facteur d'échelle de la taille de l'herbe avec un vecteur.

Insérons un nœud Combine XYZ et connectons sa sortie Vector à l'entrée Factor (vector) de Point Scale; les brins d'herbe disparaîtront, car les paramètres Combine XYZ auront une valeur par défaut de 0,0, on ramène donc XY à 1,0 (ou, en tout cas, à une valeur adéquate pour ces dimensions).

Nous connectons ensuite l'entrée Z de Combine XYZ à un nouveau port Geometry Input ; comme vous l'aurez deviné, l'intention est d'ajuster la hauteur des brins d'herbe dans l'interface du modificateur... mais pourquoi se limiter à une valeur constante ou devoir modifier manuellement cette valeur ?

Ajoutons un éditeur Timeline à la fenêtre Blender et définissons une plage d'images pour une animation, par exemple 240 images.

Passons à l'image 1, puis fixons 0.0 pour le paramètre Z (que l'on peut renommer à volonté, comme vu dans l'épisode précédent avec les noms des aiguilles de l'horloge) ; puis faites un clic droit sur la valeur de ce paramètre et choisissez "Insert Keyframe" pour enregistrer une image clé d'animation, en fait.

Passons à l'image 240 de l'animation, modifions la valeur de Z comme souhaité et insérons également une image clé pour ce paramètre à la dernière image de notre animation.

Il suffit d'appuyer sur Play dans la Timeline et de regarder le résultat de notre animation.

Eh bien, c'est tout pour ce tutoriel aussi!

À bientôt!

