

Introduction aux Geometry Nodes (nœuds de géométrie) dans Blender 2.92 - Tutoriel 1 sur 2



Le modèle 3D utilisé dans ce tutoriel: Office wall clock 1
(<https://sketchfab.com/3d-models/office-wall-clock-1-7fafd736d520456eac2066808b0a0006>)
(<https://sketchfab.com/3d-models/office-wall-clock-1-7fafd736d520456eac2066808b0a0006>)

Bonjour à tous!

Il s'agit du premier des deux tutoriels d'introduction aux nœuds géométriques dans Blender (réalisés notamment avec la version 2.92 du logiciel). Ces tutoriels sont destinés à ceux qui ont déjà une connaissance de base de Blender, car je prendrai pour acquis certaines informations sur l'interface et la modélisation (à la fois en mode Object et en mode Edit).

Geometry Nodes est un outil introduit avec Blender version 2.92 et qui vous permet de transformer des objets de manière paramétrique, c'est-à-dire en utilisant des valeurs ; c'est, en effet, un modificateur qui vous permet d'effectuer plusieurs opérations, en les combinant, au sein d'un groupe.

Sur Youtube il existe de nombreux tutoriels qui montrent, avant tout, comment utiliser des Geometry Nodes pour remplacer un système de particules, afin de "dispenser" des instances d'objets sur (ou à l'intérieur) d'autres, comme des rochers et des brins d'herbe sur une surface.. bien, dans le prochain tutoriel je ferai le même exemple aussi, mais pour ce premier tutoriel sur le sujet je veux me concentrer sur quelques concepts de base.

Nous avons dit que Geometry Nodes est un modificateur et nous savons que les modificateurs se distinguent des opérateurs de Blender car ils permettent d'effectuer des transformations de manière non destructive, ce qui signifie qu'il est possible de modifier certains paramètres également ultérieurement et de trouver les modifications s'appliquaient aussi dans les autres modificateurs ; par exemple, en vidéo je montre la différence entre une colonnade réalisée avec le

modificateur Array (en haut) et une autre réalisée avec l'opérateur Spin (en bas) : pour modifier la hauteur des colonnes, dans le cas de Array, il me suffit de passez en mode Edit et augmentez la géométrie de base pour trouver le changement dans tous les autres éléments ; en utilisant l'opérateur Spin, à la place, je devrai supprimer tous les éléments créés, modifier le premier élément en mode édition, puis appliquer à nouveau Spin.

De plus, une fois une opération effectuée avec un opérateur, si nous passons à un autre outil nous ne pourrons plus modifier les paramètres appliqués, alors qu'avec les modificateurs ces paramètres sont toujours disponibles et nous pouvons également les modifier ultérieurement (jusqu'à ce que nous appliquions le modificateur, bien sûr).

Il s'agit d'un exemple vraiment élémentaire, mais l'intérêt d'utiliser des modificateurs - quand c'est possible - est évident, surtout lorsqu'il y a le risque d'avoir à apporter diverses modifications aux éléments ou, pire, d'avoir à faire des opérations successives, pour lesquelles une modification peut nous obliger à refaire également des opérations ultérieures ; ce qui a été dit explique aussi pourquoi la modélisation avec des modificateurs est dite paramétrique et non destructive.

Geometry Nodes est donc un modificateur mais, contrairement aux autres modificateurs de base de Blender, il peut être personnalisé en combinant des nœuds.

Les nœuds sont essentiellement la représentation visuelle de deux types d'éléments : des informations ou des opérations ; si vous avez de l'expérience dans la création de matériaux ou de post-production dans Blender, vous saurez par exemple que vous pouvez prendre une information d'entrée (définie dans un nœud), effectuer des opérations (par la combinaison d'un ou plusieurs nœuds intermédiaires) et obtenir fin le résultat en sortie (qui sera à son tour défini dans un nœud). Voyons donc comment ce concept est appliqué aux transformations d'objets, à l'aide de Geometry Nodes.

Comme modèle pour l'exemple de ce tutoriel, j'utilise une horloge murale ; vous pouvez trouver le lien vers ce modèle dans la description de la vidéo, mais il n'est pas nécessaire de l'avoir : vous pouvez suivre le tutoriel et apprendre les concepts sans forcément les reproduire avec ce modèle, bien sûr !

J'ai configuré l'interface de Blender pour travailler avec les nœuds de géométrie, en ouvrant un éditeur de type "Geometry Nodes" sous la vue 3D et en ouvrant l'onglet Modifiers pour l'objet sélectionné dans l'éditeur de "Properties" (remarque : ici je règle cette mise en page car je J'enregistre une vidéo sur un moniteur ; si vous utilisez un deuxième moniteur, je vous recommande d'utiliser l'éditeur 3D Viewport and Properties sur un moniteur et d'ouvrir l'éditeur "Geometry Nodes" sur l'autre).

Les aiguilles des minutes et des heures sont des objets séparés, apparentés à l'horloge ; les deux objets ont l'origine où il devrait être (dans leur pivot), donc une rotation autour de l'axe Y (local) de chaque objet conduira à une rotation correcte ; de toute façon, pour ce tutoriel, récupérons l'Origin d'une des deux mains (en la sélectionnant et en choisissant Cursor to Selected, dans le menu que l'on peut ouvrir avec SHIFT S), puis on supprime les trois mains présentes dans la scène.

On ajoute ensuite un Cube, qui aura l'Origin là où se trouvaient celles des autres mains ; c'est important parce que les transformations auront lieu autour de ce point.

Associions le cube à la base de l'horloge d'origine, afin que nous puissions également déplacer, faire pivoter et redimensionner le cube lorsque nous transformons l'horloge.

Ajoutons un modificateur Geometry Nodes ; cette opération peut être effectuée depuis le menu Modifier de l'objet ou depuis le bouton «New» de l'éditeur de nœuds géométriques.

Par défaut, deux nœuds sont créés, appelés « Group Input » et « Group Output » ; l'entrée fournit la géométrie de l'objet telle qu'elle est fournie au modificateur (donc ici le pointeur a déjà été modélisé et l'origine a été définie manuellement : l'entrée de géométrie commencera à partir de ces données), tandis que la sortie est ce qu'elle sera renvoyée à la fin de tous les traitements, que nous effectuerons ensuite entre ces deux nœuds.

Les nœuds sont ajoutés en les choisissant dans le menu Add (raccourci : SHIFT A) ; dans ce menu, en particulier, on note que les nœuds sont divisés en différents groupes, ainsi par exemple les nœuds des groupes Mesh et Geometry vont agir sur l'ensemble de l'objet et sa géométrie (avec des opérations en Mode Object et Edit), tandis que ceux du groupe Point agira sur les particules générées ; d'autres nœuds serviront par exemple à fournir des valeurs avec lesquelles transformer les objets (groupe Input), tandis que d'autres serviront à modifier ces valeurs (groupes Utilities et Vector).

Pour transformer le cube en aiguille des minutes, ajoutons un nœud Geometry - Transform, en le plaçant en plein milieu du lien qui va de Group Input à Group Output (Blender fera automatiquement les connexions).

Jetons un œil à la couleur des ports d'entrée et de sortie des différents nœuds, avec lesquels nous pilotons le flux de données et leurs opérations : les Géométries ont une couleur verte et indiquent que des informations sur toute la géométrie de l'objet y sont passées ; par contre, les portes de type vecteur (qui contiennent des triplets de valeurs numériques, une pour chaque axe de rotation) sont violettes.

Nous fonctionnons sur la traduction et l'échelle pour définir l'aiguille des minutes ; en particulier, nous avons défini Translation Z à 0,049 et Scale XYZ à 0,001, 0,001 et 0,050 respectivement.

Je tiens à signaler une chose tout de suite : la rotation autour de l'axe Y local, pour définir les minutes, ne doit PAS être effectuée dans ce nœud, car ici les opérations ont lieu par rapport à la géométrie déplacée et redimensionnée.

Pour faire pivoter autour de l'Origin du cube, nous devons ajouter un nouveau nœud Geometry - Transform entre celui qui vient d'être créé et Group Output : là nous pourrons tourner autour de l'axe Y.

Une autre chose à noter est que ces transformations ont lieu, pour ainsi dire, "à l'intérieur du modificateur": en examinant l'onglet Item - Transform de l'objet, dans la vue 3D, nous constatons qu'en réalité Location, Rotation et Scale n'ont pas été appliqués à l'objet d'origine ; de plus, si l'on passe en mode Edit dans une vue 3D, on s'aperçoit qu'en réalité le cube est toujours là, avec ses caractéristiques d'origine !

Ce comportement, similaire en tous points à celui que l'on retrouve par exemple avec Subdivision Surface, nous fait comprendre que Geometry Nodes est en fait un modificateur (et en fait il est possible de l'appliquer, dans l'onglet Modifiers, rendant ainsi les transformations définitives... mais ne l'appliquez pas, maintenant).

Revenons à l'éditeur Geometry Node, sélectionnez le premier nœud Transform et dupliquez-le (avec SHIFT D), puis faites glisser la sortie Geometry de Group Input vers l'entrée Geometry de ce nouveau nœud, puis passez la sortie de ce nœud à la Geometry entrée de Group Output.

À l'intérieur du nœud, nous modifions Scale Z à 0,025 et Translation Z à 0,025, obtenant ainsi une aiguille des heures.

Comme on peut le voir, il est possible de prendre la géométrie initiale de l'objet (le cube, dans ce cas) et de la passer, de Group Input, à plusieurs opérateurs... mais comment obtenir les DEUX géométries modifiées dans le même Objet ?

C'est simple : en utilisant un nœud Geometry - Join Geometry, qui nous permet de fusionner plusieurs géométries en une seule et de transmettre le résultat à d'autres nœuds ; c'est donc ce que fait l'opérateur Join en mode Edit, mais ici nous l'utilisons de manière non destructive, à l'intérieur d'un modificateur !

Nous remettons la valeur Rotation Y de l'aiguille des minutes à 0, dupliquons ce nœud et plaçons la copie dans le workflow de l'aiguille des heures créée il y a un instant : de cette façon, nous pourrions régler l'angle de rotation indépendamment à la fois pour les heures que pour les minutes.

Voyons donc comment créer une petite interface, avec Geometry Nodes, pour régler rapidement les heures et les minutes dans l'onglet Modifier et faire bouger les aiguilles automatiquement.

Ce que nous voulons faire, c'est définir la rotation Y en fonction du temps (un nombre compris entre 0 et 60 pour les minutes et entre 0 et 12 pour les heures, inclus). Commençons par la Rotation des minutes.

Notez que Rotation a un socket d'entrée violet, c'est donc un vecteur ; pour séparer les canaux d'entrée, nous ajoutons un nœud Vector - Combine XYZ et connectons la sortie Vector du nœud qui vient d'être ajouté à l'entrée Rotation de Transform, puis connectons l'entrée Y de Combine XYZ à une nouvelle entrée de Group Input (il suffit de faire glisser le lien de Combine XYZ vers la prise d'entrée semi-visible de Group Input, sous Geometry).

Dans l'onglet du modificateur Geometry Nodes, le paramètre d'entrée « Value » apparaîtra avec une valeur par défaut ; pour changer le nom de ce paramètre en "minutes", ouvrez la barre latérale de l'éditeur "Geometry Nodes" (raccourci: N, tandis que le curseur de la souris est dans cet éditeur), puis changez le nom de "Value" en "Minutes" .

Nous avons également défini la valeur 0,0 pour Min (valeur minimale que l'utilisateur peut saisir) et 60,0 pour Max.

Cependant, en tapant n'importe quelle valeur entre 0 et 60, nous remarquons que l'aiguille des minutes ne tourne pas correctement ; ceci est dû au fait que la valeur de l'angle Y est exprimée en radians et, en particulier, l'angle rond vaut 6,28 (soit : deux fois pi).

Pour réaliser ce mapping (c'est-à-dire : amener une valeur de la plage 0 - 60 dans la plage 0 - 6,28), nous ajoutons un nœud Utilities - Map Range entre la sortie "Minutes" de Group Input et l'entrée Y de Combine XYZ, puis on fixe les valeurs du nœud (qui ont des significations assez intuitives), de cette façon :

- 0 dans From Min;
- 60 dans From Max;
- 0 dans To Min;
- 6.28 dans To Max.

Maintenant, dans l'onglet Modifier, nous pouvons facilement taper la valeur des minutes que nous voulons voir apparaître sur l'horloge et l'aiguille des minutes tournera de manière cohérente : plus pratique que de devoir faire pivoter l'objet autour de l'axe Y !

Quant à l'aiguille des heures, le motif des nœuds est pratiquement identique, à l'exception des valeurs de mappage dans le nœud Map Range ; procédons donc ainsi :

- nous sélectionnons et dupliquons Map Range et Combine XYZ des minutes, en plaçant les copies dans un espace vide de l'éditeur ;
- nous connectons le vecteur du nouveau Combine XYZ à l'entrée Rotation de l'aiguille des heures ;
- nous connectons l'entrée Value de la nouvelle Map Range à une nouvelle sortie Group Input (que nous pouvons renommer en Heures, entre autres) ;
- modifions la valeur « From Max » du nœud Map Range des Heures de 60 à 12.

ATTENTION : sélectionnons Group Input et assurons-nous que les valeurs Max des ports d'entrée pour les minutes et les heures sont correctement réglées à 60 et 12 respectivement !

À ce stade, nous pouvons régler les heures et les minutes des aiguilles dans l'interface de Blender et spécifier l'heure que nous voulons ... avons-nous terminé ? Et bien non : dans une vraie montre, quand l'aiguille des minutes avance, l'aiguille des heures bouge aussi (d'une heure à l'autre), alors que cela n'arrive pas dans ce modèle (essayez par exemple de régler « 2:50 » ou « 4:55 » : l'aiguille des heures s'arrête à 2 ou 4, alors qu'elle devrait être proche de 3 ou 5).

Ce dont nous avons besoin, c'est d'un petit incrément, d'un « décalage » de rotation de l'aiguille des heures, qui doit être, notamment, d'au plus 30 degrés (un douzième d'angle de tour, lorsque l'aiguille des minutes est proche de 60).

Par conséquent, nous devons d'abord prendre la valeur de l'entrée Minutes et la diviser par 2, ce que nous pouvons faire avec un nœud Utilities - Math, en changeant son mode en Divide et en définissant "Minutes" sur la première valeur et 2 sur la deuxième entrée valeur.

Le résultat obtenu est un décalage de rotation à fournir à l'aiguille des heures, mais cette valeur est exprimée EN DEGRÉS, alors que nous la voudrions en RADIANS... eh bien, nous pouvons facilement le convertir via un autre nœud Utilities - Math, cette fois en « To radians » !

Pour ajouter cette valeur à celle provenant de la Map Range de l'aiguille des heures (qui est déjà en radians), ajoutons un nœud Utilities - Math supplémentaire, cette fois en mode Add, en ajoutant Map Range et la sortie de "To Radians" , pour ensuite passer le résultat de l'opération à Combine XYZ puis, enfin, à la rotation Y de l'aiguille des heures.

En théorie, à ce stade, tout fonctionne, comme nous pouvons le voir en faisant quelques tests avec l'interface dans l'onglet Modifier ; de plus, si on transforme l'horloge, les changements sont appliqués de manière cohérente aux aiguilles aussi... ou mieux : ils sont appliqués à l'objet Cube !

Les deux aiguilles sont en fait des géométries du même Cube d'origine, qui est toujours là et est lié (en tant qu'enfant) à l'horloge : toutes les modifications ont été apportées dans le nœud Geometry Nodes et peuvent être modifiées ultérieurement, jusqu'à ce que nous appliquions le modificateur.

Avant de clore ce tutoriel, cependant, une dernière observation : les champs "Heures" et "Minutes" de l'onglet modificateur acceptent aussi des valeurs décimales et les prennent en compte dans leurs calculs, donc le résultat visuel peut être incorrect (essayez par exemple de mettre 0 pour minutes mais 9h30 pour les heures).

Une manière inélégante mais fonctionnelle de résoudre ce problème consiste à insérer des nœuds Utilities - Math, en mode Round - Floor (c'est-à-dire arrondir à l'entier INFÉRIEUR le plus proche), immédiatement après les sockets « Minutes » et « Heures » de l'entrée de groupe ; cela ne masquera pas les décimales dans l'interface du modificateur, mais au moins les ignorera dans les calculs.

Voilà, maintenant nous en avons vraiment fini avec l'interface de notre montre... et aussi avec ce premier tutoriel sur les nœuds de géométrie dans Blender 2.92 ! Dans la prochaine (et dernière, pour cette mini-série), nous examinerons un exemple de dispersion d'objets sur une surface.

À bientôt!