

Generování 3D modelů měst v Blenderu

Jan Silák

Zvolte typ práce
2024



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav informatiky a umělé inteligence

Akademický rok: 2023/2024

ZADÁNÍ BAKALÁŘSKÉ PRÁCE (projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: Jan Silák
Osobní číslo: A20318
Studijní program: B0613A140020 Softwarové inženýrství
Forma studia: Prezenční
Téma práce: Generování 3D modelů měst v Blenderu
Téma práce anglicky: Generating 3D City Models in Blender

Zásady pro vypracování

- Seznamte se s nástrojem Geometry Nodes v Blenderu.
- Popište dostupné metody tvorby 3D modelů měst v nejnovější verzi Blenderu.
- Vytvořte vlastní modul pro Geometry Nodes, který bude umožňovat procedurálně generovat 3D model města ve vybraném stylu.
- Při tvorbě modulu využijte co nejširší spektrum funkcí a možností Blenderu.
- Pro tento modul navrhnete uživatelské rozhraní, kde bude možné měnit parametry pro generování tohoto modelu.

Forma zpracování bakalářské práce: tištěná/elektronická

Seznam doporučené literatury:

1. *Geometry Nodes — Blender Manual* [online]. [cit. 2023-11-04]. Dostupné z: https://docs.blender.org/manual/en/latest/modeling/geometry_nodes/index.html
2. *Blender: Geometry Nodes – Simply Explained | All3DP* [online]. [cit. 2023-11-04]. Dostupné z: <https://all3dp.com/2/blender-geometry-nodes-simply-explained/>
3. *Blender geometry nodes fundamentals guide* [online]. [cit. 2023-11-04]. Dostupné z: <https://artisticrender.com/blender-geometry-nodes-fundamentals-guide/>
4. BAILY, Joe. *What Are Geometry Nodes In Blender? – blender base camp* [online]. [cit. 2023-11-04]. Dostupné z: <https://www.blenderbasecamp.com/home/what-are-geometry-nodes-in-blender/>
5. MAJUMDER, Yankee. *THE BEGINNERS GUIDE TO BLENDER*. Blenderhd, 2015.

Vedoucí bakalářské práce: **Ing. Pavel Pokorný, Ph.D.**
Ústav počítačových a komunikačních systémů

Datum zadání bakalářské práce: **5. listopadu 2023**

Termín odevzdání bakalářské práce: **13. května 2024**

doc. Ing. Jiří Vojtěšek, Ph.D. v.r.
děkan



prof. Mgr. Roman Jašek, Ph.D., DBA v.r.
ředitel ústavu

Ve Zlíně dne 5. ledna 2024

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 09.05.2024

Jan Silák v.r.
podpis studenta

ABSTRAKT

Cílem této práce je navrhnout a implementovat funkční modul k programu Blender, který umožňuje procedurální generování 3D modelů měst dle parametrů zadaných uživatelem. K tvorbě tohoto modulu je využito co nejširší škály funkcí nástroje Geometry Nodes.

Klíčová slova: Blender, modul, procedurální generování, měst, Geometry Nodes

ABSTRACT

The aim of this thesis is to design and implement a functional module for Blender, which allows procedural generation of 3D models of cities according to user-specified parameters. To create this module, the widest possible range of features of the Geometry Nodes tool is used.

Keywords: Blender, module, procedural generation, cities, Geometry Nodes

Chtěl bych poděkovat vedoucímu mé bakalářské práce Ing. Pavlu Pokornému Ph.D. za užitečné tipy, rychlou odezvu a připomínky během zpracování této práce.

Prohlašuji, že odevzdaná verze bakalářské/diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD	9
I TEORETICKÁ ČÁST	10
1 BLENDER	11
1.1 HISTORIE BLENDERU.....	11
1.2 UŽIVATELSKÉ ROZHRANÍ	12
1.2.1 Pracovní prostory	14
1.3 MODELOVÁNÍ.....	14
1.4 EXISTUJÍCÍ ŘEŠENÍ GENEROVÁNÍ MĚST	16
1.4.1 Blossm	17
1.4.2 Object Scatter add-on	20
1.4.3 SceneCity	22
2 GEOMETRY NODES	24
2.1 PROCEDURÁLNÍ GENEROVÁNÍ	25
2.2 FIELDS.....	26
2.3 NODE SOCKETS	27
2.4 TYPY UZLŮ.....	28
2.4.1 Attribute Nodes	29
2.4.2 Input Nodes	30
2.4.3 Output Nodes	30
2.4.4 Geometry Nodes.....	30
2.4.5 Curve Nodes, Mesh Nodes, Point Nodes, Volume Nodes	30
2.4.6 Simulation Nodes	31
2.4.7 Material Nodes	31
2.4.8 Texture Nodes	31
2.4.9 Utilities Nodes.....	32
2.4.10 Layout a Hair Nodes	32
II PRAKTICKÁ ČÁST	33
3 KONCEPT A STYL MĚSTA	34
4 MODUL NA GENEROVÁNÍ MĚSTA	35
4.1 PLOCHA MĚSTA	35
4.2 CENTRUM MĚSTA.....	38
4.3 PŘEDMĚSTÍ (SUBURB)	45
4.4 OKOLÍ (OUTSKIRT).....	47
4.5 PARKY.....	51
4.5.1 Stromy	54
4.5.2 Kamenná cesta	55
4.6 SILNICE	59

4.6.1	Chodník	61
4.6.2	Přechod.....	62
4.6.3	Semaforey a lampy	63
4.6.4	Duplikování silnic	63
4.6.5	Rotace silnice	66
4.6.6	Přesahující část.....	67
4.7	RAYCAST.....	69
4.7.1	Silnice.....	70
4.7.2	Parky	70
4.7.3	Centrum.....	70
4.7.4	Předměstí.....	71
4.7.5	Vzdálené předměstí	71
4.7.6	Okolí.....	71
4.8	UŽIVATELSKÉ ROZHRANÍ	72
ZÁVĚR		75
SEZNAM POUŽITÉ LITERATURY.....		77
SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK.....		83
SEZNAM OBRÁZKŮ		84
SEZNAM PŘÍLOH.....		86

ÚVOD

V dnešní době, kdy se technologie stále rozvíjejí a nabízejí velké množství nových možností, je vytvoření a vizualizování 3D modelů měst hlavním prvkem pro různá odvětví jako například architektury, filmového průmyslu a herního průmyslu. Za pomoci propracovaných nástrojů a technik je možné vytvořit komplexní a realistické 3D modely. Manuální vytváření modelů měst může být složitý a časově náročný proces. Proto použití open-source softwaru Blender a využití procedurálního generování za pomoci nástroje Geometry Nodes, nám dokáže ulehčit tenhle proces. Geometry Nodes nám umožňuje manipulovat s geometrií a vytvářet různorodé scény a objekty za pomoci uzlů a procedurální logiky.

Tato práce se zaměřuje na vytvoření vlastního modulu pro procedurální generování 3D modelů měst ve stylu Art Deco v Blenderu pomocí nástroje Geometry Nodes. Modul dále také bude mít uživatelské rozhraní, které umožní uživatelům měnit parametry generovaného města a přizpůsobit ho jejich představám a preferencím.

Art Deco je architektonický styl, který vzkvétal ve 20. a 30. letech 20. století, ovlivňující celosvětové městské scény od Paříže až po New York. Proslavila se zejména v Americe, kde se stala doménou mrakodrapů. Art Deco je známé svými lesklými povrchy, geometrickými tvary a vertikálními okny. Vyznačuje se elegantními budovami kombinující různé materiály jako ocel, dekorativní sklo a hliník.

Klíčovým aspektem je jednak vytvoření samotného modulu, ale také seznámení se základními funkcemi Blenderu, nástrojem Geometry Nodes a popisem dostupných metod tvorby 3D modelů měst.

I. TEORETICKÁ ČÁST

1 BLENDER

Blender je software určený pro práci s 3D počítačovou grafikou. Je open-source a bezplatný, což znamená, že je veřejně dostupný. Obsahuje velké množství různých nástrojů, které hrají roli v mnoha odvětví mediální produkce. Lze ho použít od animovaných filmů přes 3D tištěné modely, reklamy, virtuální realitu, pohyblivou grafiku, vizuální efekty, interaktivní 3D aplikace až po počítačové hry. Díky takhle velké široké škále nástrojů Blender používají jak nováčci, tak zkušení profesionálové v oblasti 3D grafiky. I když je tedy bezplatný a veřejně dostupný, tak díky obrovské komunitě a nástrojů se vyrovná i placeným možnostem 3D softwaru. Blender podporuje celý proces tvorby 3D pipeline. Znamená to tedy, že prochází 3D modelováním, riggingem, animací, simulací, renderováním, kompozicí a sledováním pohybu. Oproti ostatním softwarům je Blender licencován pod General Public License (GPL), díky tomu si jej můžete stáhnout, upravovat a používat ke komerčním účelům bez jakýchkoliv omezení. Taktéž je multiplatformní, může běžet na systémech Windows, Linux i MacOS. Obsahuje vysoce kvalitní 3D architekturu, díky které může běžet rychle a efektivně. [1][2]

Běžné nová verze Blenderu vychází každých pár měsíců a změny jsou zahrnuty v poznámkách k vydání. Ke stažení Blenderu je důležité mít zkontrolované, zda máme grafické ovladače aktuální a zda je OpenGL dobře podporováno. Při stahování nabízí Blender výběr mezi třemi balíčky. U každého balíčku je kompromisem nejnovější funkce oproti stabilitě. Stable Release je balíček, který obsahuje nejnovější funkce a je stabilní, vychází každé tři měsíce. Balíček Long-term Support je určený pro dlouhodobé projekty, které vyžadují stabilnější verzi Blenderu, ale neobsahují žádné nové funkce ani vylepšení. Oproti tomu balíček Daily Builds je denně aktualizovaný, takže obsahuje nejnovější změny ve vývoji, na druhou stranu ale není tak důkladně testovaný a nemusí být tak stabilní. Před instalací je také důležité se ujistit, zdali máme splněné minimální nebo doporučené požadavky na hardware. Doporučené je mít 8 jádrový procesor, 32 GB RAM a grafickou kartu 8 GB VRAM. [2][3]

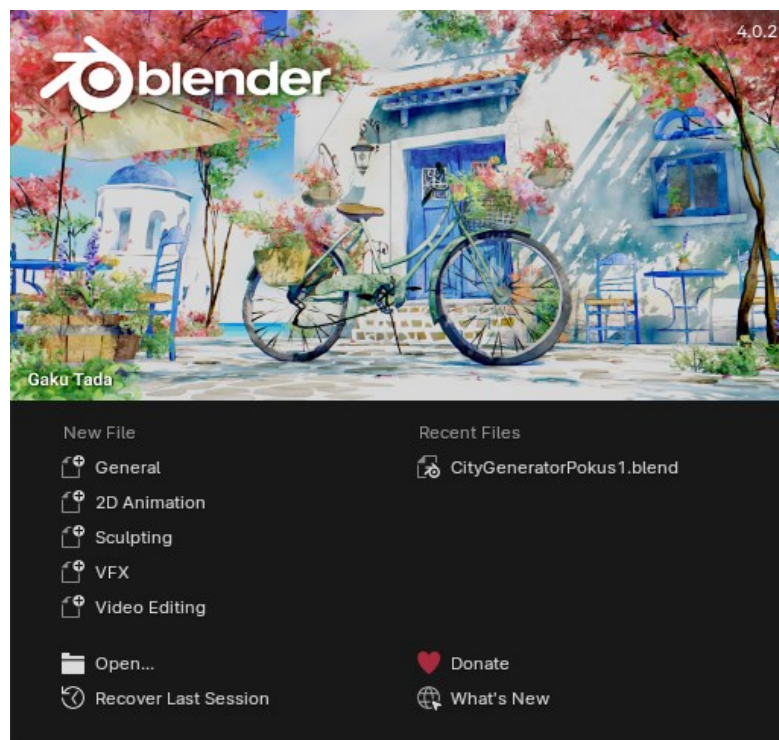
1.1 Historie Blenderu

Historie Blenderu sahá až do roku 1988, kdy Ton Roosendaal spoluzaložil NeoGeo, významné nizozemské animační studio. Studio NeoGeo získalo uznání ve světě za svou produkci. V roce 1995 Ton Roosendaal inicioval o přepracování zastaralého 3D nástroje NeoGeo, z něhož se později vyvinul Blender. V roce 1998 Ton Roosendaal založil

společnost Not a Number, aby dále vyvíjela a uváděla na trh Blender s představou bezplatného 3D softwaru na profesionální úrovni. V roce 2000 vyšla verze Blender 2.0, což byl veliký milník, jelikož obsahovala herní engine a počet uživatelů Blenderu rostl. V roce 2002 došlo kvůli finančním překážkám k uzavření společnosti Not a Number. Ton Roosendaal ale ve stejný rok v březnu založil nadaci Blender Foundation, aby pokračovala dále ve vývoji Blenderu. V říjnu 2002 byl Blender vydán pod licencí GNU GPL. Dnes Blender stále pod vedením Tona Roosendala vzkvétá díky celosvětové komunitě. [4]

1.2 Uživatelské rozhraní

Po spuštění programu Blender se nám zobrazí úvodní obrazovka, která umožňuje rychlý přístup základním možnostem, jako třeba vytvořit nový projekt, nebo otevřít nedávný projekt. V horní části vpravo úvodního obrázku můžeme vidět verzi Blenderu.



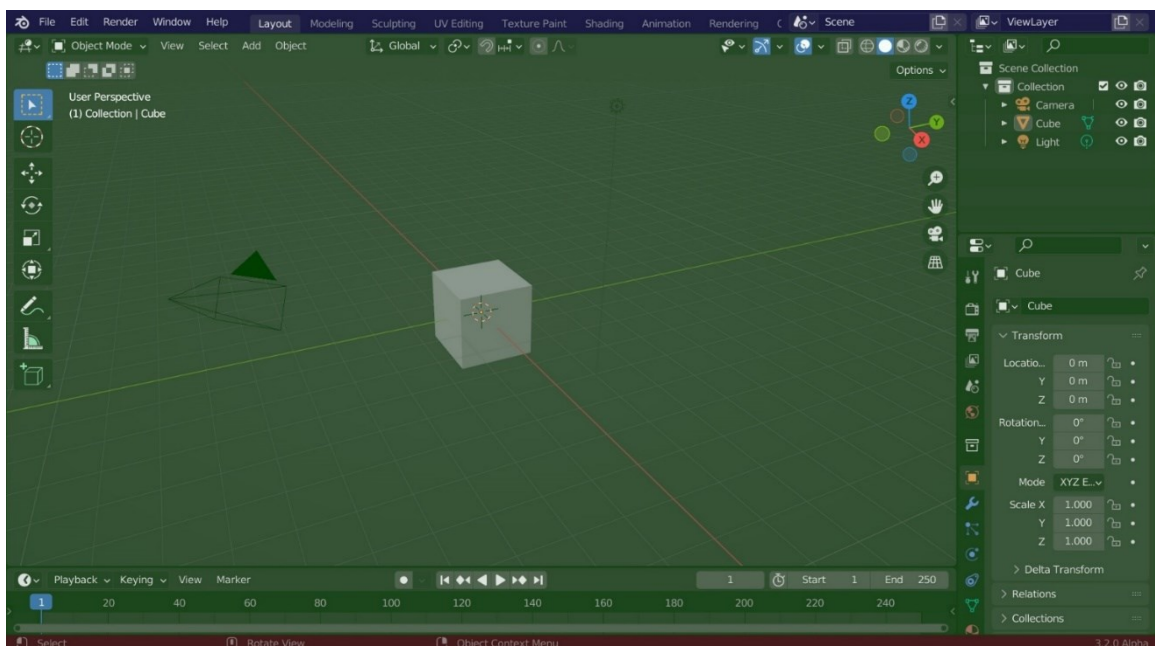
Obrázek 1 Úvodní obrazovka

Rozhraní programu Blender je rozděleno do třech základních částí. První část se jmenuje horní panel, nachází se nahoře a můžeme tam najít hlavní nabídku, která slouží k ukládání, importu a exportů souborů, konfiguraci nastavení a vykreslování. Dále se tam nachází seznam pracovních prostorů, který je rozebrán v kapitole 1.2.1 Pracovní prostory. [5]

Druhá část což je vlastně největší a zároveň hlavní pracovní plocha. Uprostřed můžeme vidět velký 3D Viewport, díky kterému vidíme náhled scény. Vpravo nahoře je potom

Outliner, což je vlastně seznam, který nám zobrazuje všechny objekty, které se právě nacházejí v naší scéně, nejen to ale také důležité informace a vztahy mezi nimi. Hned pod ním se nachází panel vlastností, který je velmi důležitý, jelikož obsahuje veškeré informace o vybraném objektu a také některé vlastnosti ohledně scény obecně. Jakýkoliv objekt, co přidáme, bude mít kartu vlastnosti objektu. Posledním základním nástrojem v téhle části je editor časové osy. Časová osa nám poskytuje přehled o animaci scény. Najdeme zde aktuální snímek (Current frame), klíčové snímky (Keyframes) aktivního objektu, počáteční a koncové snímky naší animační sekvence a námi nastavené značky (Markers). Všechny tyto popsané panely jsou konfigurovatelné. Můžeme jednoduše měnit velikost kliknutím a přetažením přes prostor mezi panely. Pokud chceme vytvořit nový, přesuneme kurzor do horního rohu a až se nám změní na křížek, tak stačí kliknout a přetáhnout ho do panelu Viewport, tím se zobrazení znásobí, pak už si jen nastavíme, jaký typ editoru chceme. [6][7]

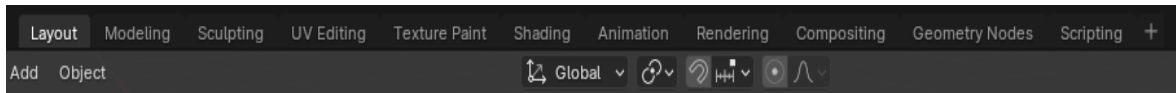
Třetí část takzvaně stavový řádek, který je v dolní části zobrazuje návrhy zkratk a příslušené statistiky. [5]



Obrázek 2 Výchozí rozložení, první část (modrá), druhá část (zelená), třetí část (červená) [5]

1.2.1 Pracovní prostory

V horní části rozhraní můžeme vidět předdefinované rozvržení pracovních prostorů viz obrázek 3. Jednoduše jsou to kombinace editorů různých obrazovek, které jsou uspořádány určitým způsobem. Blender má k dispozici několik výchozích pracovních prostorů. Stačí kliknout na záložku karty pracovního prostoru a rozhraní se nám změní. Každý z těchto pracovních prostorů je navržen tak, aby uživateli poskytl právě potřebné nástroje pro



Obrázek 3 Pracovní prostory

úlohy, jako jsou například modelování (Modeling), stínování (Shading). Výchozí pracovní prostory můžeme také změnit nebo vytvořit zcela nové. Na konci seznamu pracovních prostorů se nachází ikona plus. Pokud na ní klikneme, otevře se nám nabídka „Add Workspace“ a umožní nám si vybrat z dalších predvoleb pracovních prostorů a nebo také můžeme duplikovat aktuální pracovní prostor. Díky možnosti přizpůsobení mnoha pracovních prostorů nám Blender umožňuje plnou kontrolu nad tím, jak si uživatelské rozhraní rozvrhneme.

Pracovní prostor Modeling je skoro totožný jako úvodní rozvržení (Layout), jediné změny jsou zvětšení prostoru vyhrazeného pro 3D zobrazení a časová osa animace dole je odstraněna. Účelem použití je, abychom měli více prostoru pro modelování 3D objektů a zároveň přístup k nejdůležitějším editorům. [8][9]

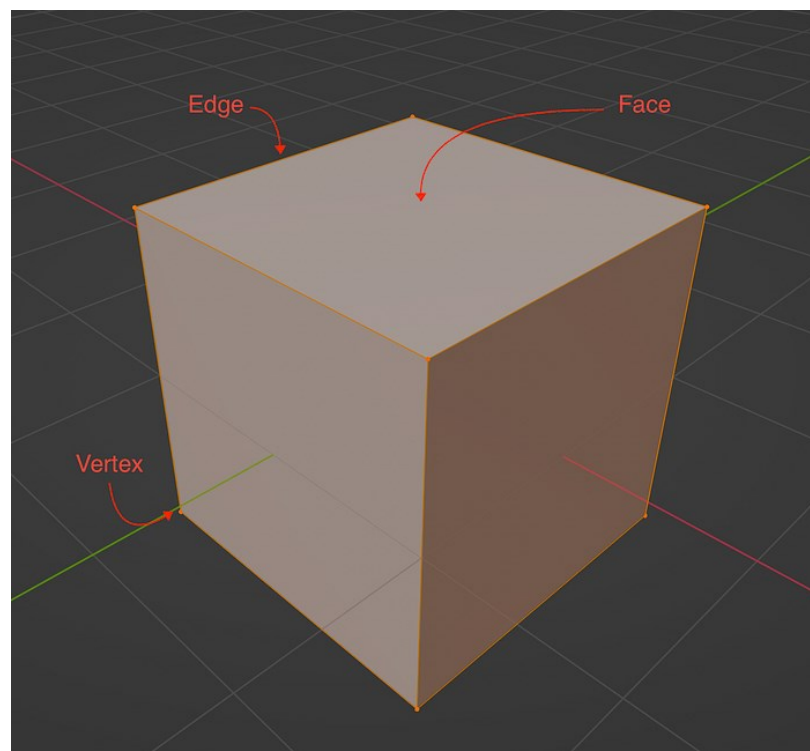
Pracovní prostor Shading použijeme právě pro vytváření materiálů v Blenderu, nachází se zde editor shaderů, který zobrazuje pracovní oblasti stínování. Dále v tomto pracovním prostoru najdeme 3D zobrazení, prohlížeč souborů pro přístup k souborům 3D materiálů, editor obrázků a panel vlastností [9]

1.3 Modelování

Zjednodušeně můžeme říct, že definice 3D modelování je matematická reprezentace jakéhokoli objektu nebo scén ve třech rozměrech pomocí nějakého speciálního softwaru v našem případě Blender. 3D objekt se nazývá Mesh a je vytvořený skutečnými vrcholy, hrany a plochy. V Blenderu při vytváření 3D objektů nám pomáhá několik různých režimů. Dva základní, ale hlavní režimy používané právě k vytváření 3D modelů se jmenují režim objektu (Object Mode) a režim úprav (Edit Mode). [10]

Výchozím režimem Blenderu je právě objektový režim, pomocí kterého můžeme manipulovat s objekty jako s celkem. To znamená, že nám umožňuje základní funkce jako přesouvání, otáčení a změny velikosti objektu. Pokud chceme upravit tvar objektů, použijeme režim úprav. V Režimu úprav máme k dispozici tři různé režimy výběru a to Vertex (vrchol), Edge (hrana) a Face (plocha). Dále k základním funkcím, které jsme měli u objektového režimu, přibyly další důležité funkce, které můžeme rozdělit do dvou skupin. První skupina jsou aditivní nástroje a mají zelené ikony, pomocí nich přidáváme geometrii. Nejpoužívanějšími jsou Extrude, Inset Faces, Bevel a Loop Cut. Ve druhé skupině najdeme „deformační“ nástroje, mají růžové ikony a manipulují s již existující geometrií. V praxi se mezi těmito dvěma režimy přepíná velmi často, proto abychom ušetřili čas, můžeme použít klávesu Tab pro přepnutí mezi režimem úpravy a objektu. [10][11][12]

K vytvoření modelů z různých objektů, které obsahuje Blender jako například Meshes, Curves, Surfaces, Metaballs a Text objects je právě zapotřebí režim úprav. Nejvíce známá a používaná forma modelování se nazývá Mesh Modeling. U těchto formy modelování je nejdůležitější základní struktura, která se skládá z vrcholů (Vertices), hran (Edges) a ploch (Faces) viz obrázek 4.[12]



Obrázek 4 Mesh objekt a jeho struktura

Nejzákladnější částí je vrchol (Vertex), který představuje jeden bod nebo pozici v 3D prostoru. V režimu úprav vrcholy najdeme jako malé tečky a jsou uloženy jako pole souřadnic. Takhle dva spojené vrcholy pomocí přímky nám představují hranu (Edge) a pokud máme 3 nebo více hran umístěné na jedné ose tvoří plochu (Face). Plocha je definována buď to jako oblast mezi třemi, čtyřmi nebo více vrcholy s hranou na každé straně. Můžeme se také setkat se zkracováním na trojúhelníky, čtyřúhelníky a n-úhelníky. Blender obsahuje řadu jednoduchých tvarů jako například rovina, krychle nebo kruh, které nám mohou pomoci se začátkem modelování. [12][13][14]

Další formou vytváření 3D objektů se kterou se můžeme setkat je Curve Modeling, která pracuje s křivky. Křivky se používají k vytváření zaoblených objektů a tvarů. Vyjádřeny jsou matematickými funkcemi, kterým se říká interpolace. V Blenderu můžeme najít Bézierovy křivky, Nurbs a Path. Beziérovky křivky se běžně používají v grafice a designu, kde křivka prochází svými definičními řídicími body, pomocí kterých můžeme manipulovat s tvarem křivky. Nurbs se liší od Beziérových křivek tím, že nemusí nutně procházet řídicími body, ale jsou jimi ovlivňovány, což umožňuje vytvářet hladší křivky. Tato vlastnost je dosažena pomocí matematických funkcí vyjadřujících NURBS (Non-Uniform Rational B-Spline), které umožňují flexibilnější manipulaci s tvarem křivek v porovnání s Beziérovými křivkami.

Path funguje podobně jako Nurbs, používají se hlavně pro animace a vytvářejí plynulý pohyb protažením do konečného řídicího bodu. Hlavní výhodou používání křivek místo mesh objektů je, že křivky jsou definovány menším množstvím dat, díky tomu mohou na konečné výsledky použít méně paměti a úložného prostoru.[13][15]

1.4 Existující řešení generování měst

Na internetu existuje velké množství různých řešení pro vytváření a generování měst. Velká většina z nich je vytvořena pomocí Blender Python API, ale také se začaly objevovat řešení na Blendermarketu za pomoci Geometry Nodes. Blender má vestavěné funkce na interpretování jazyka Python. Díky tomuto interpretu jsme schopni spouštět python skripty, které nám vykreslují uživatelské rozhraní a používá se také pro pár interních nástrojů Blenderu. Většina dostupných řešení je ale zpoplatněna, proto se zmíním o řešeních, které jsou zdarma dostupné ke stažení a pouze jednoho placeného řešení, které mě zaujalo. [16]

1.4.1 Blossm

Jako jeden z nejznámějších add-onů pro generování měst se nazývá Blossm. Blossm je zkratkou pro Blender-OSM (OpenStreetMap). Umožňuje nám pomocí pár kliknutí importovat 3D modely měst z [Google Maps](#), [OpenStreetMap](#) a terén s globálním pokrytím. Add-on má dvě verze. První verze je zdarma a uživatel, co si jej stáhne může darovat peníze, pokud chce. Jedná se o základní verzi, která dokáže importovat tyto následující položky. [17]

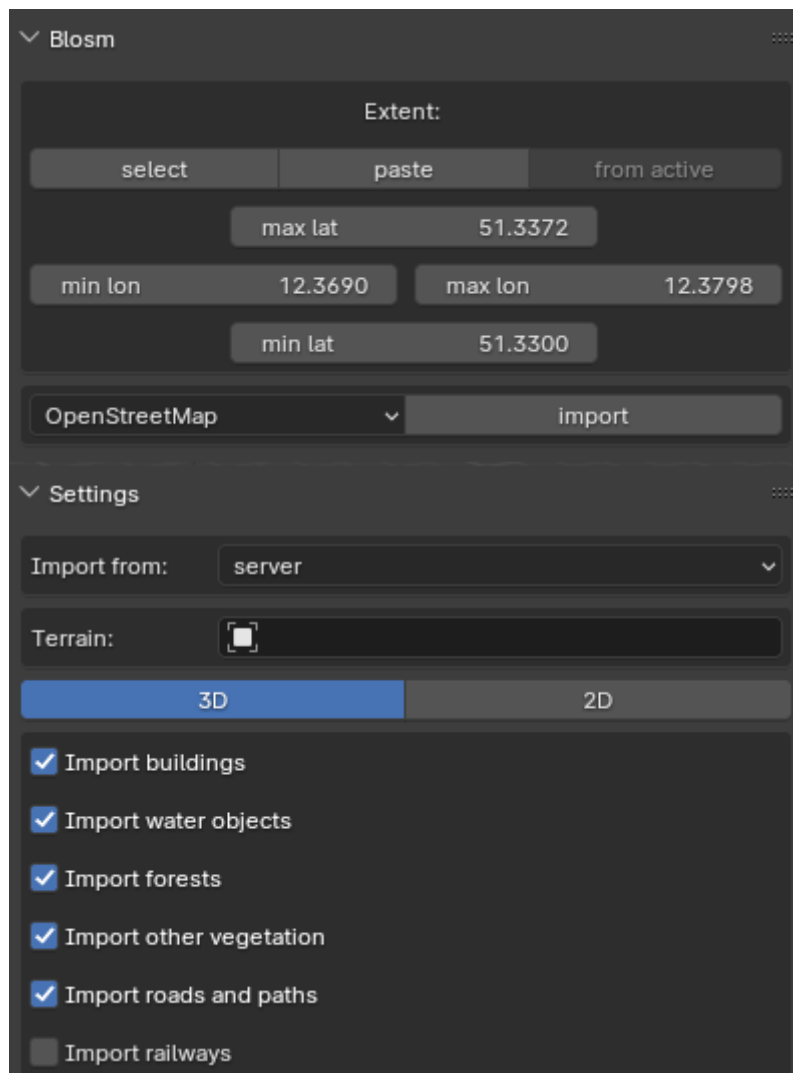
- Budovy: Za pomoci dat výšky budov a počet pater se vytvoří výsledná scéna. Pokud zahrneme nějaké složitější strukturované budovy, tak je kompozice zpracována do 3D částí. Zahrnuje velké množství tvarů střech od plochých přes kulaté a pyramidální až po kopule. Jeli k dispozici i terén, tak se budovy na něj sami automaticky umístí.
- Řeky a jezera: Jsou importovány jako polygony a pobřeží moří a oceánů jako hrany. Pokud má k dispozici terén, tak se zmíněné objekty na něj promítnou.
- Lesy, tráva a křoví: Fungují totožně jako řeky a jezera.
- Silnice, cesty a železnice: Jsou reprezentovány pomocí křivek
- GPX-cesty: dokáže importovat GPX-cesty a promítat je na terén. GPX je textový soubor s uloženými geografickými informacemi, jako jsou body cesty, stopy a trasy.

Pokud bychom si chtěli připlatit za prémiovou verzi, tak ta nám nabízí ještě k základním funkcím navíc materiály a textury napodobující osvětlená okna, importování lesů a jednotlivých stromů jako 3D objekty a možnost aplikování Geometry Nodes pro získání realistických 3D budov. [17]

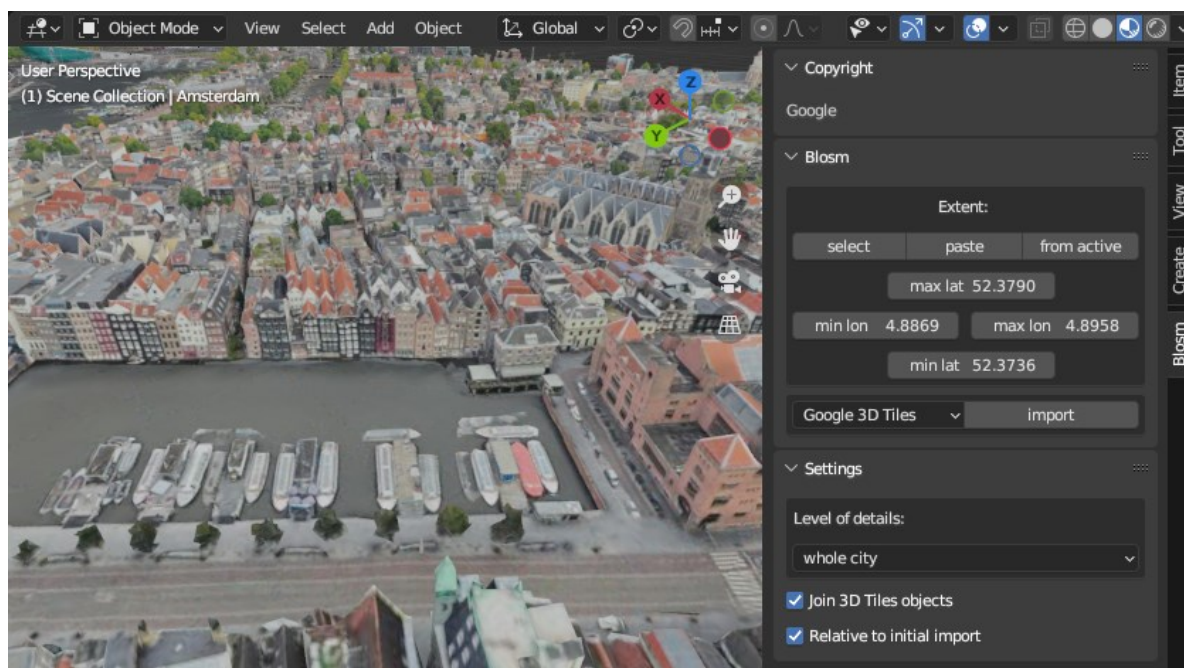
Před instalací je důležité zkontrolovat, že používáme nejnovější verzi programu Blender. Add-on lze stáhnout pomocí odkazu v oficiální dokumentaci. Stáhneme si .zip soubor, který v Blenderu nainstalujeme kliknutím na Edit dále Preferences, Add-ons a Install. Pak už jen stačí zaškrtnout políčko Enable Add-on. Dalším důležitým krokem, bez kterého tento add-on nebude fungovat je nastavení adresáře pro uložení stažených souborů z OpenStreetMap a terénu. Doporučuje se založit adresář přímo pro tyto soubory. Zbývá poslední krok vybrat si pomocí které funkce budeme chtít importovat města. Na výběr máme ze tří možností lokalizační služby ArgGIS, společnosti Mapbox anebo pomocí

Google. At už si vybereme jakoukoliv z těchto tří možností, stačí pouze kliknout na tlačítko Get It. Nejjednodušší možností je použít Google 3D Tiles, které lze použít k vizualizaci více než 2500 měst v 49 zemích. [17]

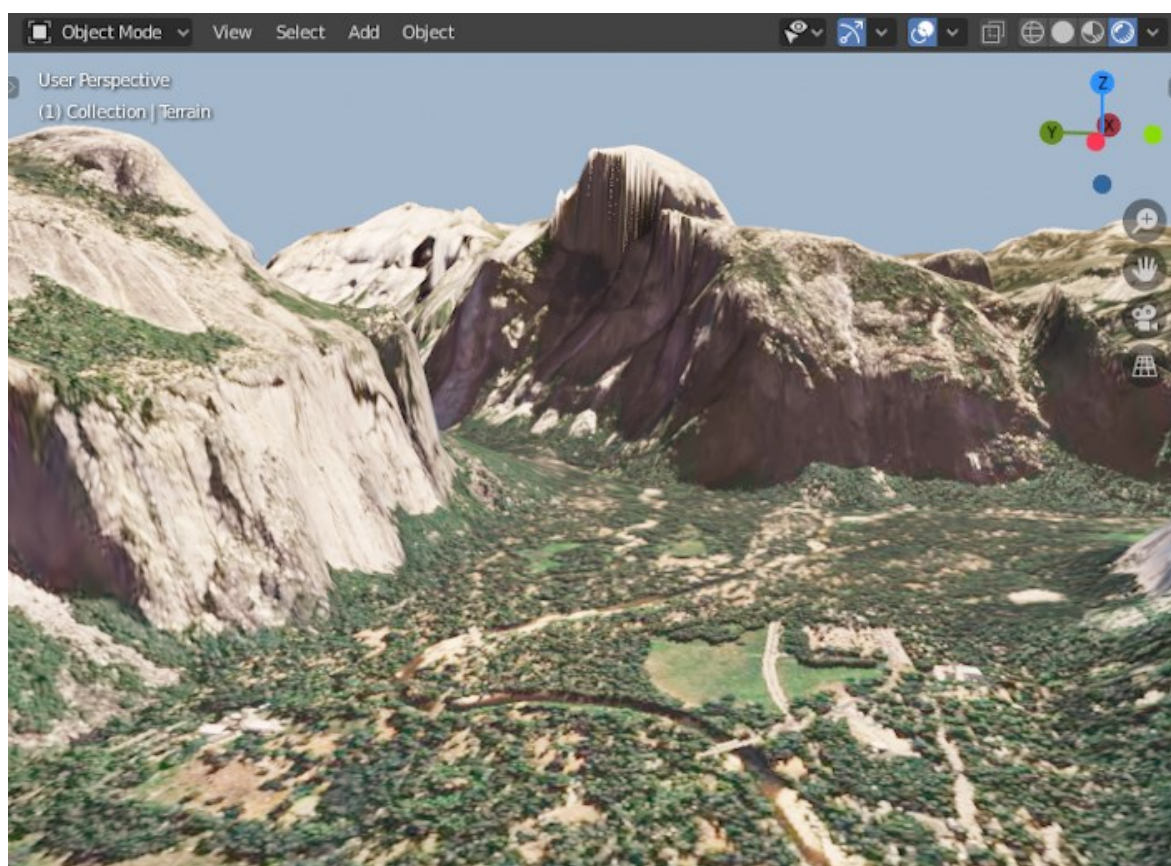
Menu Blossm se nachází v pravém bočním panelu, který otevřeme stisknutím klávesy N. V rozevíracím seznamu přenastavíme z OpenStreetMap na Google 3D Tiles. Stisknutím tlačítka Select vybereme rozsah mapy, který chceme nahrát. Po kliknutí se nám zobrazí speciální okno prohlížeče se stránkou pro výběr rozsahu mapy. Na stránce budeme následovat pokyny a na konci získáme řetězec se souřadnicemi, které si zkopírujeme do Blenderu pomocí tlačítka Paste. Následně si vybereme námi požadovanou úroveň detailů v seznamu Level of details a poté stačí kliknout na tlačítko import a máme hotovo.[18]



Obrázek 5 Blossm menu a jeho použití



Obrázek 7 Blosm ukázka importované části města z oficiální dokumentace



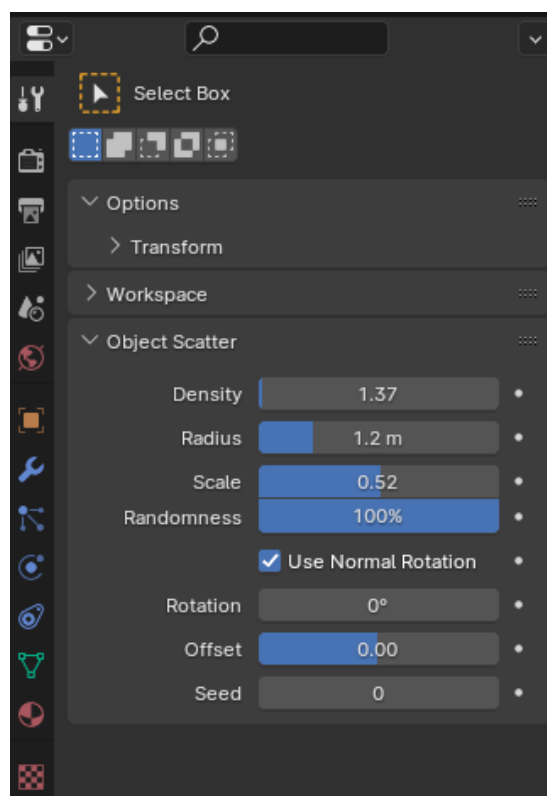
Obrázek 6 Blosm ukázka importované části terénu z oficiální dokumentace

1.4.2 Object Scatter add-on

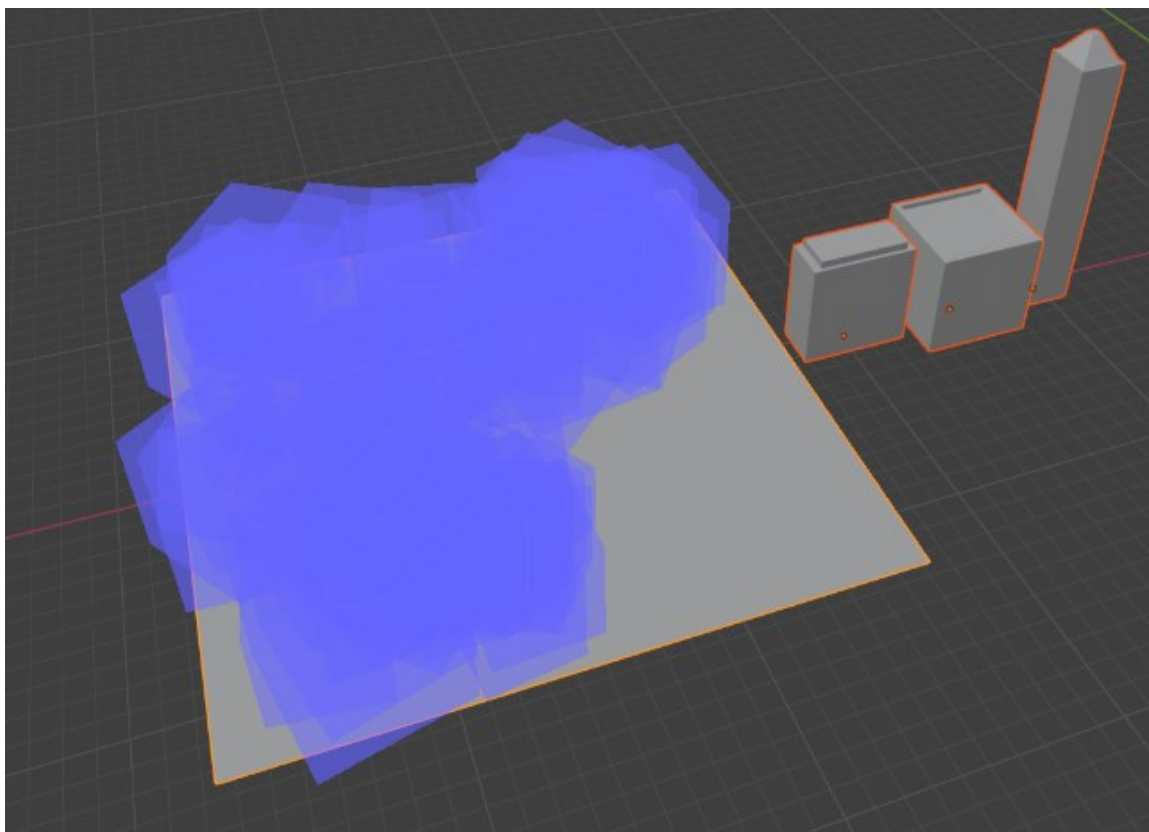
Tenhle Add-on není přímo specializovaný na generování měst, ale pomocí něj můžeme dosáhnout generovaného města. Add-on nám umožňuje rozmístit mesh objekty na jakýkoliv povrch, nemusí to být přímo rovina, můžeme použít třeba i krychli. Výhodou toho to addo-nu je, že je vestavěný přímo v Blenderu, stačí nám ho pouze povolit. Povolit ho můžeme stejně, jako u minulého řešení a to tak, že přejdeme do záložky Add-ons a ve vyhledávacím okně v pravém horním rohu napíšeme Object Scatter a zaškrtneme políčko Enable.

Jeho funkčnost je velmi jednoduchá, první si musíme určit, které objekty budeme chtít rozmístit a tím vytvořit naše město. Vytvořil jsem základní příklad, kde jsem udělal běžně používané transformace na třech krychlich, aby připomínaly budovy. Tyhle tři objekty si označím a dále označím plochu, na kterou je rozmístím v mém případě na objektu Plane. Poté klikneme F3, vyhledáme Scatter objects a pomocí myši nakreslíme trasu podél které se nám rozmístí. Nakreslenou trasu můžeme vidět na obrázku 11 a výsledný vygenerovaný výstup na obrázku 10. [19]

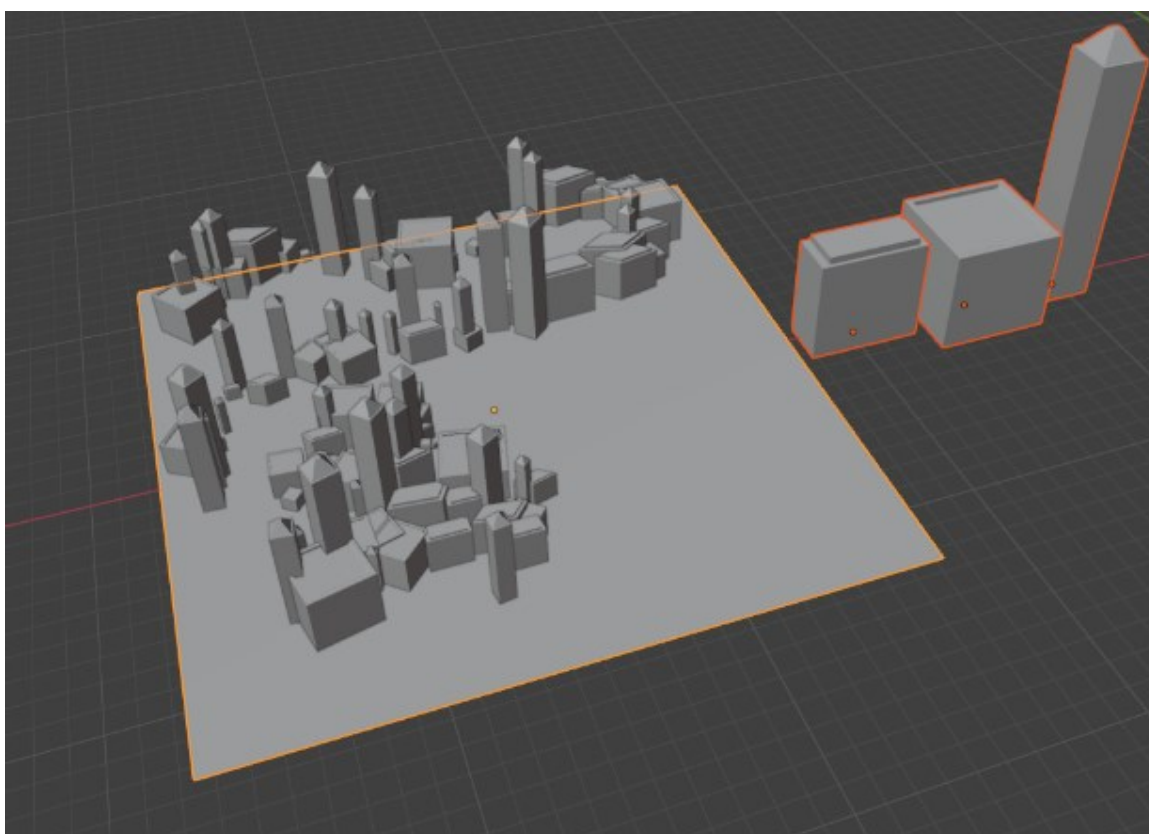
Add-on obsahuje i základní menu, ve kterém si můžeme upravit třeba hustotu objektů, měřítko, náhodnost a rotaci. Najdeme ho v panelu vlastností, který se, jak jsem již zmiňoval v kapitole 1.2 Uživatelské rozhraní nachází v pravém dolním rohu. [19]



Obrázek 8 Object Scatter menu



Obrázek 10 Příklad použití Addo-nu Object Scatter

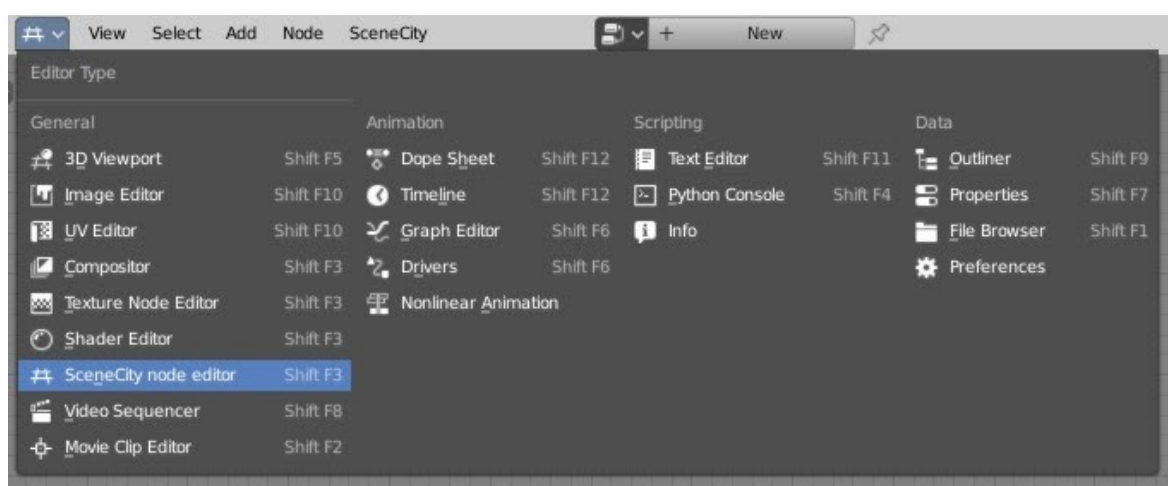


Obrázek 9 Výstup z Addo-nu Object Scatter

1.4.3 SceneCity

Jedná se o placený Add-on, který se dá koupit za 39 dolarů. Momentálně má dvě verze, první verze je již dokončena a druhá verze je stále ve vývoji. Pro instalaci addo-nu je zapotřebí nejprve stáhnout příslušné soubory ze stránky [Gumroad](#), včetně samotného addo-nu a tří balíčků pro města, přírodu a terén. Poté v Blenderu klikneme na Edit a Preferences. Pomocí tlačítka Install nainstalujeme samotný add-on a po instalaci je třeba doinstalovat jednotlivé tři balíčky, které byly taktéž zároveň staženy. Po dokončení instalace jsou stažené soubory již zbytečné a můžeme je smazat. [20]

SceneCity v Blenderu umožňuje tvorbu měst za pomoci grafů, které propojují uzly pro manipulaci s daty. Graf slouží jako popis města a lze jich mít v jednom .blend souboru libovolný počet. Uzly provádějí specifické operace se vstupními daty a generují transformovaná nebo nová výstupní data. Je důležité rozlišovat data v Blenderu a data interními pro SceneCity. Pro práci s grafy je nutné otevřít nový typ editoru v Blenderu,

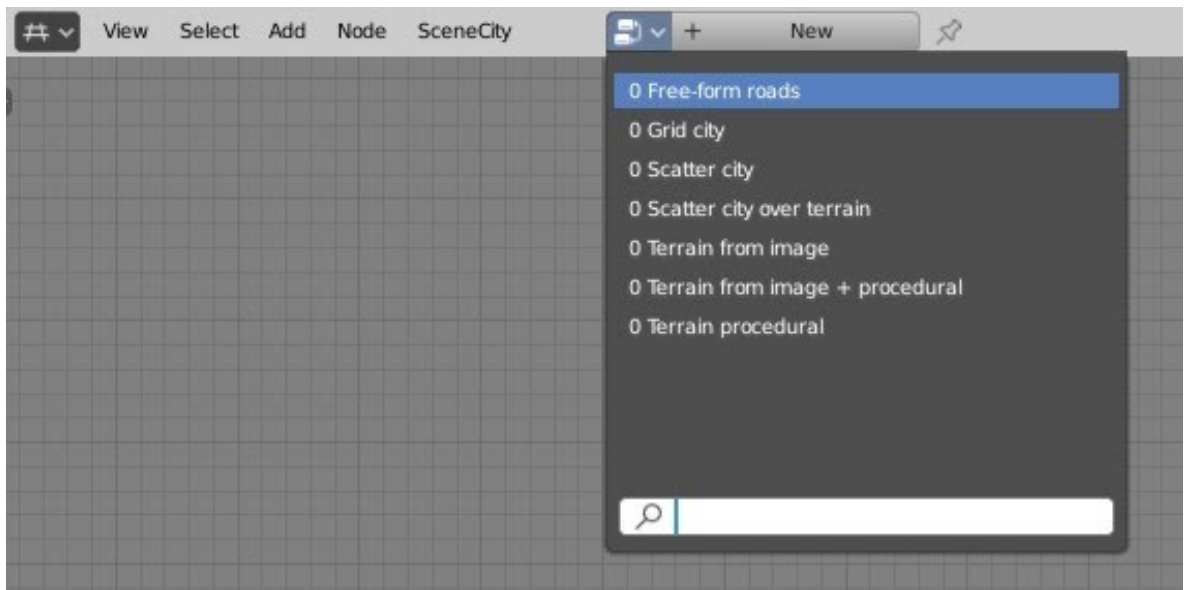


Obrázek 11 SceneCity editor pro práci s grafy [21]

který je k dispozici po aktivaci addo-nu SceneCity, můžeme ho vidět na obrázku 12. Lze buď importovat přednastavená města nebo vytvořit nové grafy od základů dle našich potřeb. [21]

Jelikož dokumentace funkcí placené verze tohoto addo-nu není zdarma dostupná, tak se zmíním pouze o popsání části, jak vložit již před vytvořením města. Klíčové je použít možnost Link nebo Append v editoru grafů pro import těchto dostupných modelů. Po importu budou dostupné v Outlineru, ale jsou původně skryté, aby se předešlo zbytečnému zaneprázdnění scény. Kromě toho jsou také k dispozici pomocné objekty, které usnadňují práci s některými přednastaveními, najdeme je pod názvem „SceneCity presets helper

objects“. Taky nám seznam grafů měst, který vidíme na obrázku 13, nabízí různé typy přednastavených měst, které lze generovat a upravovat podle potřeby. [21]

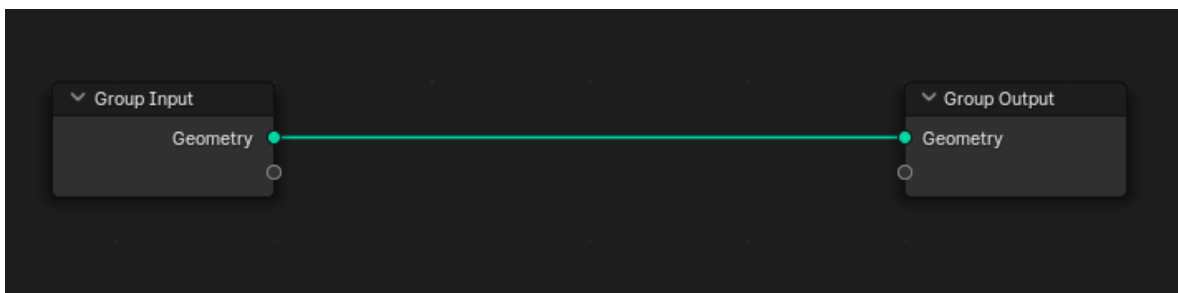


Obrázek 12 SceneCity seznam přednastavených grafů měst

Občas generování může trvat až několik minut, podle složitosti úkolu. Proto se doporučuje otevřít konzoli programu Blender ještě před spuštěním náročných úloh, jako je třeba generování velkých měst. Tohle opatření se doporučuje, protože takové procesy mohou dočasně způsobit, že uživatelské rozhraní Blenderu přestane reagovat. Konzole potom slouží jako nástroj pro sledování průběhů textových aktualizací SceneCity. [21]

2 GEOMETRY NODES

Jsou relativně novou funkcí spadající pod nástroje Blenderu. Byly poprvé představeny ve verzi 2.92 a to jako postup založený na uzlech (anglicky node-based workflow) pro konstruování a manipulování s geometrií. Jedná se o procedurální pracovní postup, který vede software krok za krokem při vytváření nějakého objektu prováděním souborů operací na základě uživatelem definovaných parametrů. Většinou se tok dat pohybuje zleva doprava, výjimkou budou pouze takzvané Fields, kde je to právě naopak. Za pomoci Geometry Nodes se vlastně všechny stávající činnosti v Blenderu od základních 3D operací, jako je modelování až po vytváření složitějších realistických scén stávají procedurálními. Můžeme například vytvářet složité tvary, vzory a animace pomocí Geometry Nodes, tak že spojujeme různé uzly dohromady a definujeme, jak se geometrie tvoří a upravuje. Uzly jsou v podstatě bloky, které obsahují strukturovaná data a transformují nám vstupy na výstupy na základě parametrů definovaných ve skupině uzlů. Každý uzel slouží k určitému účelu a jak jsem již zmínil, kombinací těchto uzlů můžeme dosáhnout složitých výsledků bez nutnosti ručního modelování. Každý uzel je charakterizován svým názvem, parametry, vstupy a výstupy. Skupina uzlů v Blenderu je nazývána jako strom vzájemně propojených uzlů. Začáteční bod stromu je uzel Group Input, který představuje počáteční stav objektu. Konečný bod se jmenuje Group Output a reprezentuje výsledek všech operací, které jsou mezi počátečním a koncovým bodem.



Obrázek 13 Začáteční a koncový bod stromu uzlů [23]

V nejnovější verzi Blenderu existuje nespočet různých typů uzlů od barev, geometrie, materiálů až po text. [22][23][24]

2.1 Procedurální generování

Je to metoda v počítačové vědě, která používá algoritmy, díky kterým dochází k automatickému vytváření obsahu. S touto technikou se můžeme setkat ve širokém spektru oborů například je velmi používaná k vývoji her a počítačové grafice. Často se používá pro vytváření měst, krajin, 3D objektů, animací, design postav a nebo také třeba dialogy nehratelných postav. Jeden ze známých příkladů jsou planety vytvořené ve hře „No Man’s Sky“. V této hře je hráč schopen prozkoumat až 18 kvintilionů unikátních planet a měsíců, které jsou generovány za pomoci algoritmů počítačem. Další zajímavý a možná známější příklad je počítačová hra Minecraft. Může se zdát, že je to jednoduchá hra s kostkami, ale skrývá za sebou složitý procedurální generování, který zajišťuje, že každý svět je unikátní. Procedurální generování se může uplatnit všude, kde je zapotřebí rychlé a efektivní generování velkého množství obsahu. [25][26][27]

Procedurální generování funguje za pomoci použití počítačového kódu, který nám vytvoří data za pomoci algoritmů a nikoliv ručně. Je to tedy způsob, jak vytvářet obsah za pomoci programů, na základě nějakých sady pravidel a parametrů definované vývojářem. To může zahrnovat cokoli od herních úrovní a textur až po různorodé 3D modely. Je to výkonný nástroj a umožňuje nám vytvářet rozsáhle, složitě světy s minimálním ručním vstupem, což šetří čas a zdroje. Dokáže také generovat jedinečné nepředvídatelné výsledky a přidat do her a simulací prvek překvapení a znovu hrátelnosti. Procedurální generování se dá také využít v datové vědě, lze jej použít ke generování a vytvoření obrovských datových sad pro trénování a testování modelů strojového učení. Hodí se to zejména tehdy, pokud jsou reálná data citlivá. [25][27]

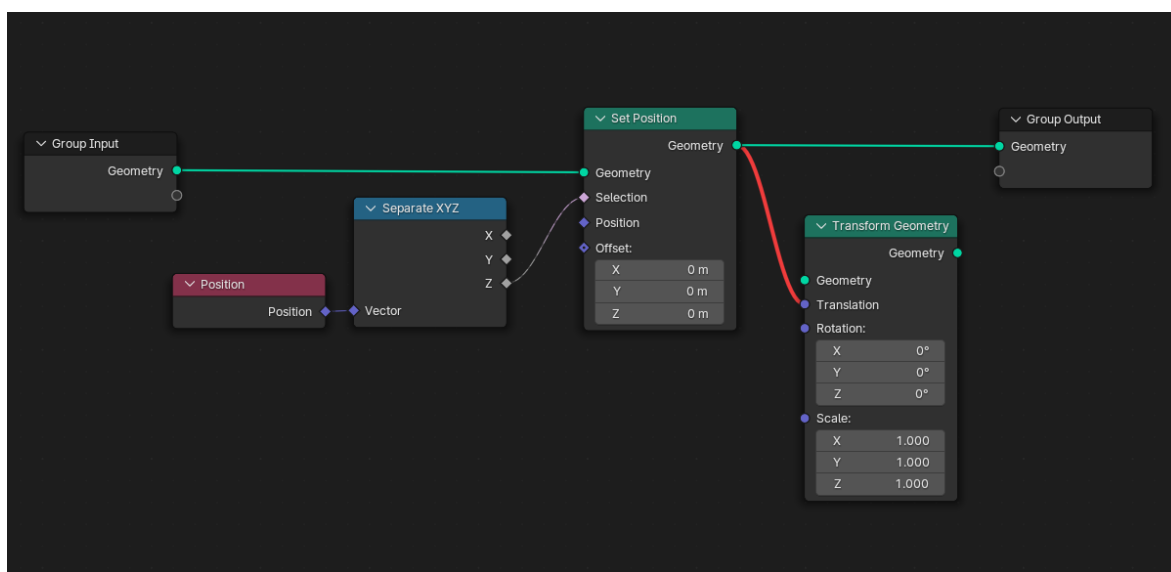
Proces procedurálního generování zahrnuje definování sady pravidel nebo algoritmů, které pak počítač dodržuje při vytváření stanoveného obsahu. Taková pravidla mohou být jak velmi jednoduchá, tak i složitá a zahrnovat náhodné či deterministické prvky. Například ve hře, ve kterém se hráč pohybuje ve městě, může vývojář definovat pravidla pro náhodné generování města, ulic nebo parků. Díky těmto stanoveným pravidlům je pak počítač schopen vytvořit jedinečný herní svět, pokaždé co se hra hraje. [25]

Procedurální generování nabízí při tvorbě obsahu různé výhody i nevýhody. Mezi hlavní výhody patří zvýšení efektivity díky automatizaci generování velkého množství obsahu, čímž potenciálně šetří čas a peníze oproti manuální tvorbě. Vyhýbá se opakujícím se úkolům a zajišťuje tak škálovatelnost, což umožňuje relativně snadno vytvářet jak malé

scény, tak i rozsáhlá prostředí. Podporuje konzistenci a opakované použití, protože zajišťuje, že prvky se řídí stejným stylem a lze je snadno opakovaně použít s drobnými úpravami. Procedurální generování má několik nevýhod. Jednou z nich je ztráta kontroly nad detaily a specifickými designy. Další nevýhodou je obtížné spravování výstupu a zajištění, aby generovaný obsah nebyl opakující se a nezajímavý, ale právě poutavý a zajímavý. Vzhledem k tomu, že obsah vytváří počítač pomocí algoritmů, může někdy postrádat kreativitu. [25][28]

2.2 Fields

Ne vždy ale musíme připojit právě jeden vstup do výstupu, je možné použít takzvané Fields, což jsou instrukce, které transformují libovolný počet vstupů do jednoho výstupu. Výhodou je, že výsledné výstupy mohou být vypočítány mnohokrát s různými vstupními daty. Používají se všude v Geometry Nodes, aby umožnily vypočítat výsledky pro různé prvky jako například ploch (Faces) nebo vrcholů (Vertices). Jak jsem se již zmínil na začátku kapitoly, typický tok dat proudí zleva doprava, ale u Fields data proudí zprava doleva. Pokud uzly mezi sebou správně zapojíme, bude spojení uzlu vykresleno jako přerušovaná čára viz obrázek 15. To se nazývá Function Flow Links, tahle čára se zobrazí, pokud se odkaz z uzlu vztahuje na více prvků. Spadá pod takzvané Function Nodes. Mezi uzlem Group Input a Set Position je propojení zelené, to znamená, že se jedná o Data Flow Nodes. Jsou to uzly, které vždy mají Geometry vstup i výstup. Obě tyto skupiny uzlů budou vysvětleny více na konci této kapitoly.



Obrázek 14 Příklad zapojení Fields

Když ale uděláme někde chybu při připojení vstupu mimo pole do vstupu k poli, bude připojení nakresleno jako plná červená čára, která indikuje, že došlo ke špatnému zapojení viz Obrázek 15. Na stejném obrázku se nachází uzel pozice (Set Position), který bude předávat data pozice pro samotný výběr. Dále je použit uzel Separate XYZ a pak výstup z kanálu Z, jako náš výběr. Pokud nastavíme následně Offset osy Z na 1 metr, dostaneme takovýto výstup. [29][30][31]

Uzly lze rozdělit do dvou kategorií, a to uzly toku dat (Data Flow Nodes), které většinou procházejí daty geometrií a uzly polí (Field Nodes) pracujících na datech. Například použijeme uzel Set Position, který je použit i na obrázku 14, tak vidíme, že můžeme předat geometry data do vstupu uzlu a zároveň i je pak předat z uzlu ven pomocí výstupu. Pokud se tedy geometry data přenášejí přímo skrz uzly, je to případ Data Flow Nodes. U Field Nodes obecně můžeme říct, že analyzují a mění hodnoty. Například předáme několik hodnot do jednoho uzlu a ten transformuje vstupní hodnoty na jedno číslo, které pak ovlivní výsledné geometry data, takže slouží ke změně hodnot v uzlovém systému. Rozdíl mezi nimi jde také zjistit pomocí tvarů Node Socketů. Má-li Node Socket tvar kruhu, označuje tak Data Flow a jeli tvar diamantu označuje Field. Co to znamená Node Socket se dozvíme v kapitole 2.3 Node Sockets. [30]

2.3 Node Sockets

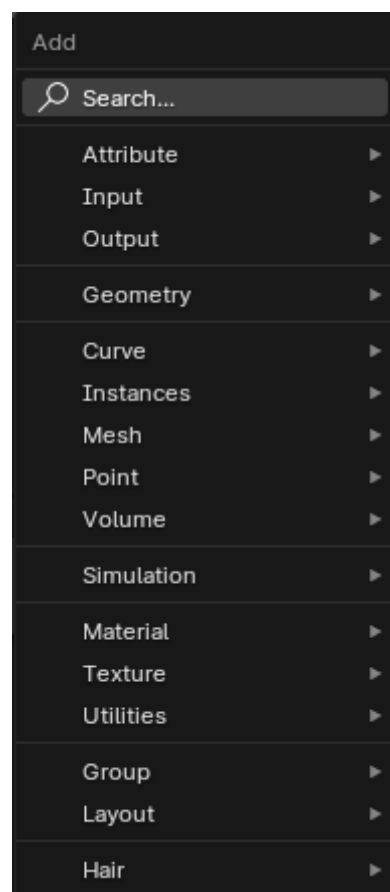
Pro rozlišení dat vstupů a výstupu se používají různé tvary a barvy. Rozlišujeme základní tři tvary a to kruh, diamant a diamant s tečkou. Barva potom udává, o jaký druh vstupu nebo výstupu se jedná. Kruhové tvary vyžadují pouze jednu skutečnou hodnotu, nemůže přijmout vstup pole. U výstupů to znamená, že uzel vždy vypíše jednu hodnotu. Diamantové tvary označují, že se jedná o vstup pole nebo také může vypsát výstup pole. Vstupy pro pole mohou také přijímat data jako vstup, který jinak prochází Kruhovými tvary. Diamantové tvary s tečkou mohou být pole, ale v současné době je jedinou hodnotou. Díky tomu nám to umožní sledovat, kde jsou vypočítány jednotlivé hodnoty, a ne pole s různými výsledky. [22]

Každý vstup nebo výstup je zbarven podle toho, jaký typ dat zpracovává. Blender má dvě zabudované barvy první je světle zelená, používá se pro shadery ve vykreslování v Cycles a EEVEE. Druhá je Tyrkysová, která se využívá v uzlech geometrie. Pro data jsou to potom barvy růžová, žlutá, šedá, limetkově zelená, světlé modrá a tmavě modrá. Barva růžová zastupuje datový typ Boolean a slouží k předávání 1 (True) nebo 0 (False). Žlutá je

pro barvu, většinou jsou to tři čísla reprezentující RGB. Šedá je typ Float a přijímá nebo produkuje čísla s desetinnou čárkou. Limetkově zelená zastupuje datový typ Integer, což je celočíselná hodnota. Světle modrá je datový typ String, používá se k předávání textové hodnoty a tmavě modrá označuje vektorovou informaci s hodnotami X, Y a Z. Vstupy jsou umístěny vlevo dole v uzlu a výstupy vpravo nahoře. I když nám může připadat, že rozlišení je jednoduché a stejné typy dat jsou vždy kompatibilní, není tomu tak. Datový typ se pouze odkazuje na formát uložení dat. Proto data, která jsou uložena podobně, tak mohou být velmi odlišené případy užití. Například geometrický výstup může obsahovat i více typů dat jako jsou křivka (Curve), objem (Volume) nebo síť (Mesh), ale vkládání geometrických dat, která jsou vytvořena pomocí křivky je něco jiného než daty sítě. Proto musíme kontrolovat, jaké údaje předáváme a jak se vztahují k jiným typům dat, když chceme přejít z jednoho typu na druhý. [22][32]

2.4 Typy uzlů

V aktuální verzi Blenderu 4.0 existuje 16 druhů uzlů. V Blenderu si můžeme zobrazit Node menu, které vidíme na obrázku 16 pomocí klávesové zkratky Shift+A nebo kliknutím na tlačítko Add, ale musíme být přepnutý v pracovním prostoru pro Geometry Nodes. [23]

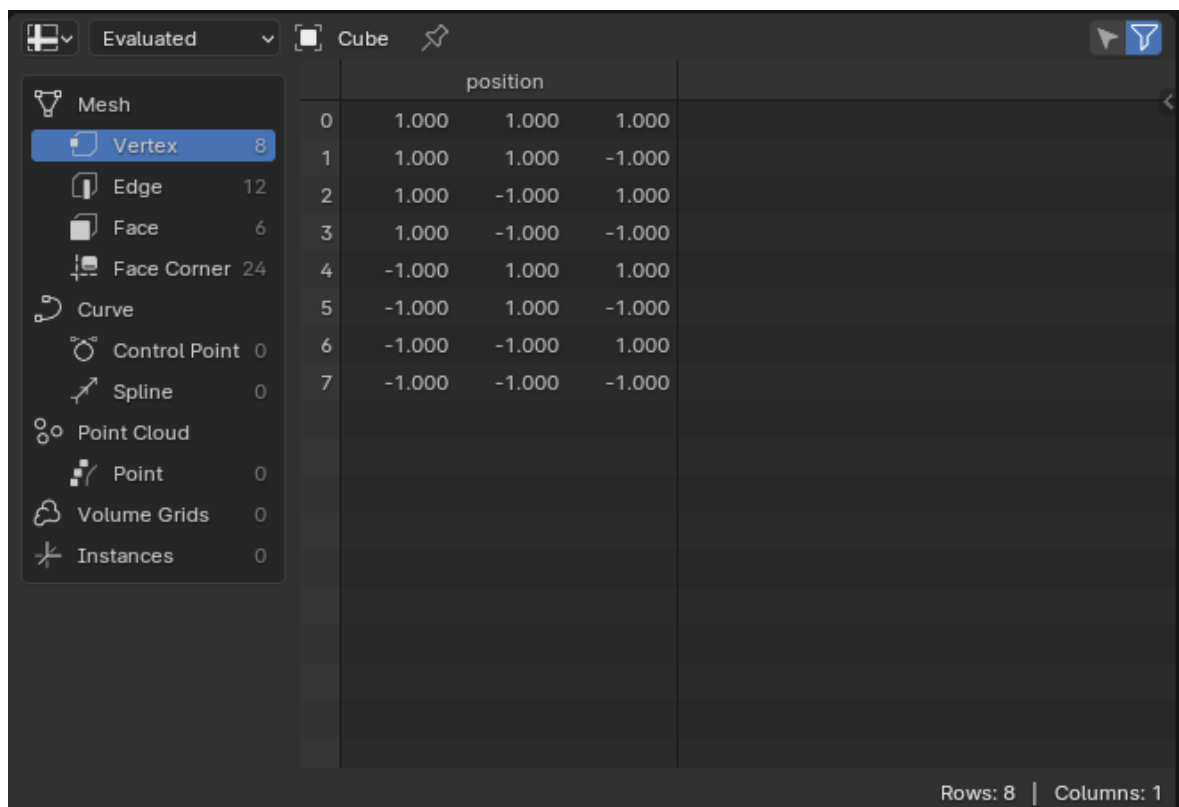


Obrázek 15 Blender Node menu

2.4.1 Attribute Nodes

Atribut je obecný název pro popis uložených dat pro jednotlivé prvky v datovém bloku geometrie. Nejběžnější atributy jsou Index, ID a Position. Pokud se nacházíme v pracovním prostoru Geometry Nodes, tak v levém horním rohu uvidíme nabídku, v níž najdeme data, jako jsou údaje o mesh, cloud anebo curve. Také pod mesh máme přístup k údajům o vrcholech, hranách, ploch a také i Face Corners. V Blenderu se jim říká domény a všechny atributy mají přiřazenou doménu a typ. Pokud bychom si chtěli tuhle nabídku zobrazit jinde, najdeme ji pod názvem Spreadsheet editor. [33]

Blender obsahuje pojmenované atributy, anonymní atributy a zabudované atributy. Pojmenované atributy jsou vytvořeny a používány v oblastech jako shaders a UV mapping. Sada obecných dat uložených na geometrii, která nemá název se říká anonymní atributy. Většinou veškeré atributy nacházející se v Blenderu mají svůj název, ale v Geometry Nodes se atributy můžou předávat pomocí node sockets, které jsou rozepsány v kapitole 2.3. V tomhle případě se tedy vytvoří výstupní pole atributu, které pak následně uzly používají k najetí dat atributu ve vstupní geometrii. Stále ale jsou uloženy v geometrii stejně jako ostatní atributy. Zabudované atributy nemohou být smazány a jejich datový typ a doména se nedá změnit, patří mezi ně třeba position, radius, index a další. [33][34]



	position
0	1.000 1.000 1.000
1	1.000 1.000 -1.000
2	1.000 -1.000 1.000
3	1.000 -1.000 -1.000
4	-1.000 1.000 1.000
5	-1.000 1.000 -1.000
6	-1.000 -1.000 1.000
7	-1.000 -1.000 -1.000

Obrázek 16 Blender Spreadsheet editor obsahující atributy

Attribute Nodes jsou tedy uzly pracující s daty uloženými v jednotlivých prvcích jako např. vertex groups. Mezi nejpoužívanější patří Stored Name Attribute Node a Capture Attribute Node. Stored Name Attribute ukládá výsledek fields (co jsou to fields je popsáno v kapitole 2.2 Fields) jako atribut se zadaným názvem. Capture Attribute také ukládá výsledek fields, ale předává data výstupem Attribute, aby jej mohly používat další uzly. [33]

2.4.2 Input Nodes

Uzly sloužící především jako vstup do jiných uzlů, dělí se na Input Constant Nodes a Input Scene Data Nodes. Do Input Constant Nodes se řadí například uzel Value Node, který zadává číselné hodnoty do jiných uzlů a do Input Scene Data Nodes se řadí uzel Object Info, přes který je možné nahrát informace kteréhokoliv existujícího objektu a dále s ním pracovat. [35]

2.4.3 Output Nodes

Existují dva uzly, a to Viewer Node, pomocí kterého můžeme jakoukoliv připojenou geometrii vizualizovat ve viewport, stačí stisknout klávesy Shift+Ctrl+Levé tlačítko na myši na jakýkoliv uzel a Group Output, který se používá k zobrazení konečného výsledku stromu uzlů. [35]

2.4.4 Geometry Nodes

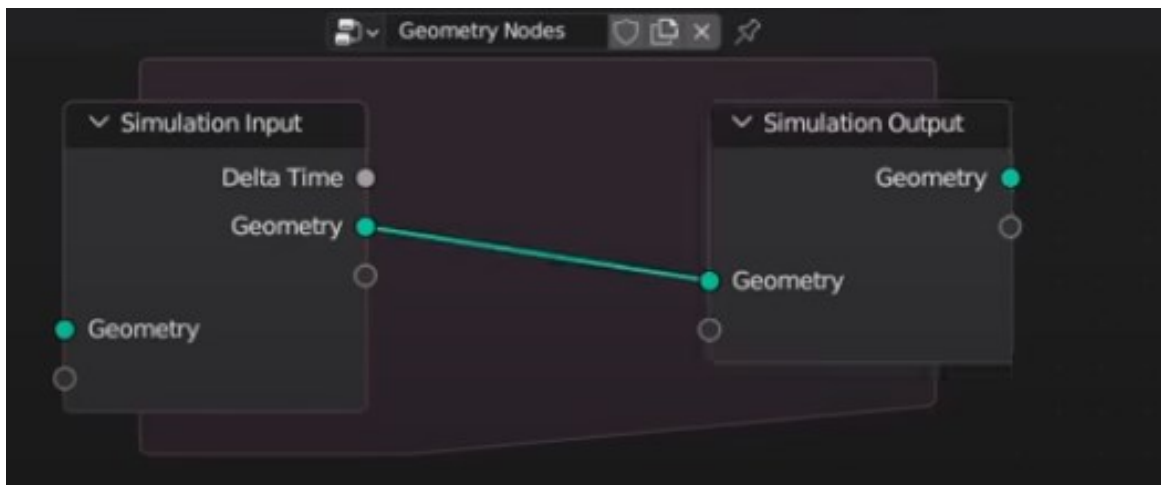
Uzly, pomocí kterých dokážeme vytvořit a manipulovat s různými objekty a tvary. Nejpoužívanější uzel je Transform Geometry Node, díky němu jsme schopní aplikovat základní operace jako přesouvání, otáčení a měnění velikost. [35]

2.4.5 Curve Nodes, Mesh Nodes, Point Nodes, Volume Nodes

Jsou to uzly pracující s různými typy geometrie, lze je použít k provedení základních funkcí jako je vytváření, manipulace nebo převod na jiný typ geometrie. [35]

2.4.6 Simulation Nodes

Ve verzi Blenderu 3.6 LTS byly představeny simulační zóny, které nám umožňují, aby výsledek jednoho snímku ovlivnil následující snímek. Funguje pomocí dvou uzlů, které mezi sebou definují simulační zónu. Vstupy připojené k uzlu Simulation Input jsou vyhodnoceny pouze jednou na začátku simulace a následně předány do dalšího stavu simulace zakončeny výstupem. Zvenčí téhle zóny můžeme připojit další jiné uzly. Ty jsou potom každý krok znovu vyhodnocovány pomocí jejich hodnoty v každém daném snímku. [36]



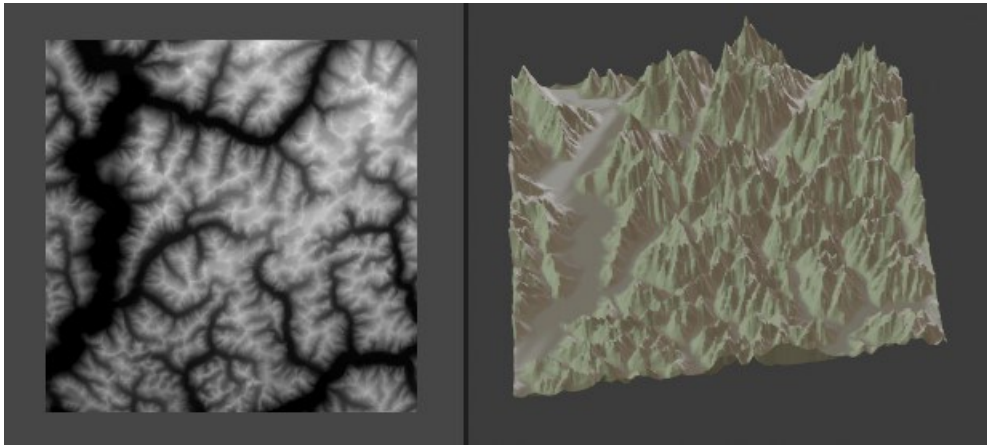
Obrázek 17 Blender simulační zóna

2.4.7 Material Nodes

Uzly pracující s materiály, slouží k nastavení určitého materiálu pro připojený geometry objekt. [35]

2.4.8 Texture Nodes

Textury jsou procedurálně generované vzory, můžeme je použít jako vstup několika způsobů. Například můžeme pomocí uzlu Image Texture Node přidat obrázek, kterým vytvoříme různé efekty od vzorů až po povrchové detaily, jako jsou hrboly. [38][39]



Obrázek 18 Blender ukázka použití Texture Nodes [39]

2.4.9 Utilities Nodes

Různé seskupení uzlů sloužící k úpravě dat. Nejvíce používané uzly v tomhle seskupení jsou Math a Vector. Uzly v sekci Math nám dávají přístup k základním funkcím matematiky, jako plus, mínus, násobek a dělení. V sekci Vector opět najdeme základní matematické funkce díky uzlu Vector Math nebo další užitečné uzly pro práci s vektory. [35]

2.4.10 Layout a Hair Nodes

V Layout můžeme najít základní funkce, díky kterým můžeme uzly uspořádat pro lepší čtení a pochopení. Patří sem Reroute, když ho vložíme na propojení mezi dva uzly, tak nám vytvoří bod, se kterým můžeme pohybovat pomocí klávesy G. Pokud máme skupiny uzlů, můžeme je seskupit do rámečku (anglicky Frame) pouhým označením a přetažením. [37]

Uzly zaměřené pouze na generování nebo úpravu křivek se jmenují Hair Nodes a anglický název již napovídá, že se hlavně používají k vytváření vlasů. [40]

II. PRAKTICKÁ ČÁST

3 KONCEPT A STYL MĚSTA

V úvodu jsem se již zmínil, že generovaný model města bude ve architektonickém stylu Art Deco a popsal jsem pár základních informací o tomhle stylu, v téhle kapitole navážu na úvod a rozvedu více svůj výběr.

Jeden z hlavních důvodů, proč jsem si vybral Art Deco je, že nabízí jedinečnou estetiku, která působí elegantně a výrazně. Moje volba stylu města byla ovlivněna právě městem, které jsem zmiňoval jako jedno z nejznámějších metropolí světa New York City. New York City je ikonickým symbolem městského života a architektury. V New York City můžeme najít spoustu slavných budov v Art Deco stylu, jako je Chrysler Building, Empire State Building nebo Rockefeller Center. New York City nabízí širokou škálu architektonických stylů, takže zde nenajdeme pouze Art Deco, ale i mnoho dalších stylů, což umožňuje kombinaci Art Deco s dalšími prvky a zvýšit tak rozmanitost modelu.



Obrázek 19 Empire State Building

4 MODUL NA GENEROVÁNÍ MĚSTA

V téhle části práce se zaměřím na návrh a implementaci modulu pro generování 3D modelů měst ve stylu Art Deco. Realizace je v programu Blender za využití nástroje Geometry Nodes. Hlavním cílem modulu je poskytnout uživateli uživatelsky přívětivé rozhraní, které umožní nastavování různých parametrů města a manipulaci s jeho strukturou.

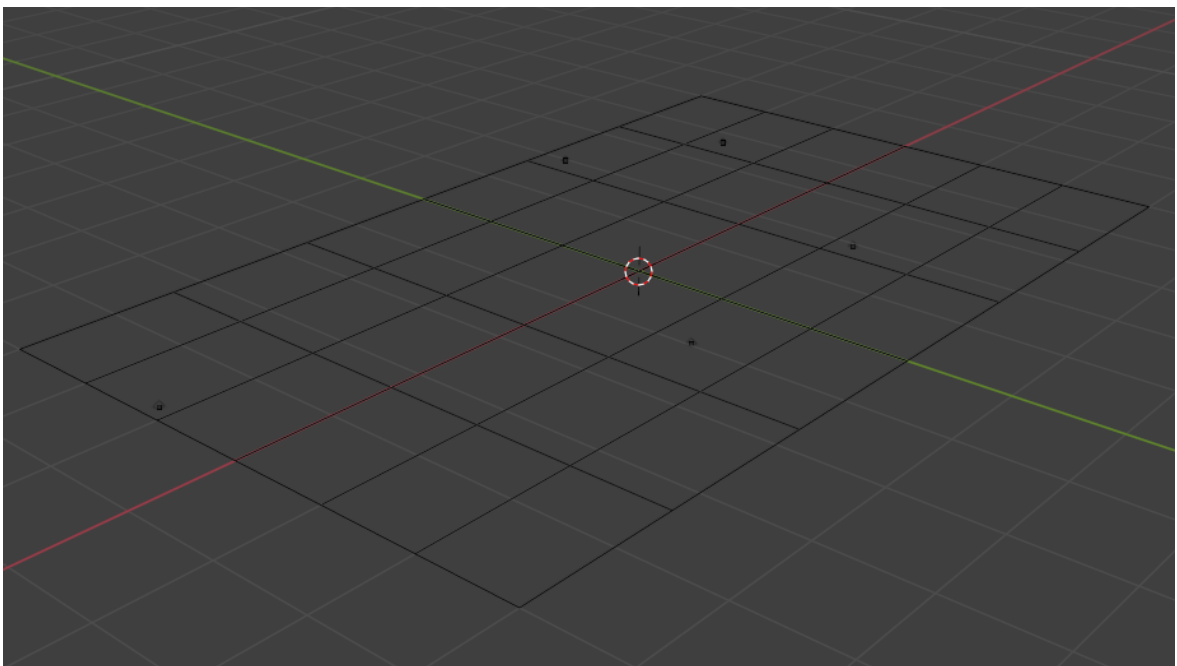
Modul obsahuje několik klíčových prvků, které umožní uživatelům efektivně generovat 3D modely měst. Prvním je velikost jednotlivých částí ploch města. Uživatel může upravit velikost všech čtyř částí města, což ovlivní celkovou rozlohu a rozmístění budov. Jedná se o části centrum, předměstí, vzdálené předměstí a okolí. Parametr hustoty budov umožní uživatelům definovat vzdálenost mezi jednotlivými budovami a díky tomu ovlivní hustotu a strukturu města. Silnice jsou vytvořeny za pomoci duplikování křižovatky, to umožní rychlé a přesné vytváření pravoúhlých cest, kde si uživatel může změnit pouze rotaci. Dále v modulu je speciální nástroj XYZScaler. Funguje tak, že uživateli umožňuje měnit výšku budov v určené části města, tam kam si přesune právě tenhle objekt XYZScaler.

4.1 Plocha města

Základním prvkem celého generátoru je plocha, na které se město bude generovat. První je důležité se přepnout v Blenderu na pracovní prostor Geometry Nodes, co jsou to pracovní prostory je popsáno v kapitole 1.2.1 Pracovní prostory. V dolní části se nachází prázdný Geometry Node Editor, aby se mohlo začít pracovat s Geometry Nodes, musí se nejprve přidat nový strom uzlů pomocí tlačítka New. Vytvoří se právě na základní objekt Cube. Po vytvoření nového stromu je vidět v Geometry Node Editor, ale také i v panelu vlastností konkrétně pod záložkou modifikátory. Základní název po vytvoření se vygeneroval jako Geometry Nodes, pokud by se vytvořil další, tak jeho název by byl Geometry Nodes.001. Pro změnu názvu stačí na něj pouze kliknout a přepsat ho. Pro lepší přehlednost byl název změněn na City Generator. S Geometrií základního objektu Cube se pracovat nebude, takže za pomoci tlačítka Ctrl + pravé tlačítko na myši a následným přjetím přes spojení uzlu Group Input a Group Output se spojení uzlu rozpojí. Pomocí klávesové zkratky SHIFT+A je možné vkládat nové uzly, stačí si v menu najít konkrétní uzel anebo napsat název uzlu.

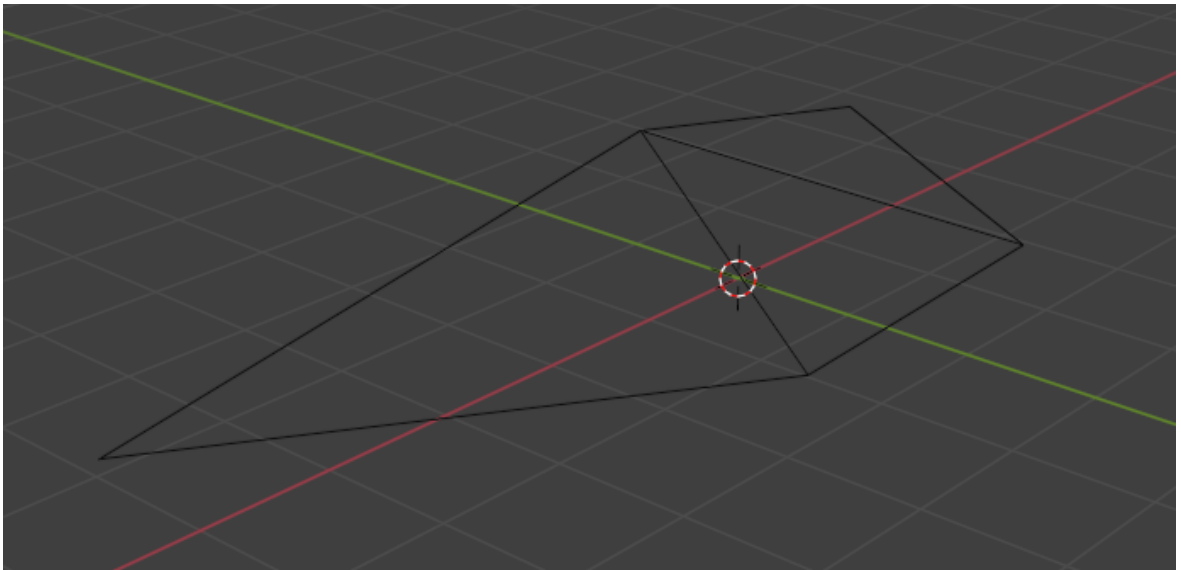
Náhodní plocha města se vygeneruje za pomoci několika uzlů. První a hlavním uzlem je Grid. Uzel Grid obsahuje čtyři vstupy Size X, Size Y, Vertices X a Vertices Y. Aby

uživatel mohl měnit tyhle vstupy a tím tak měnit velikost plochy města, tak je potřeba je propojit s uzlem Group Input. Uzel Group Input má vždy jeden prázdný nepoužitý socket. Kliknutím na něj a přetáhnutím do vstupu Size X uzlu Grid se vytvoří do rozhraní nový vstup, který se sám přejmenuje na Size X a přidělí typ Float. Stejná operace byla provedena s ostatními třemi vstupy uzlu Grid. První dva vstupy byly přejmenovány na Size of city X a Size of city Y. Poslední dva vstupy se taktéž přejmenovaly, a to na Vertices X a Vertices Y. Na téhle ploše grid se rozmístí body za pomoci uzlů Distribute Points on Faces. Tento uzel nám umožňuje rovnoměrně rozložení bodů na povrchu připojeného objektu.



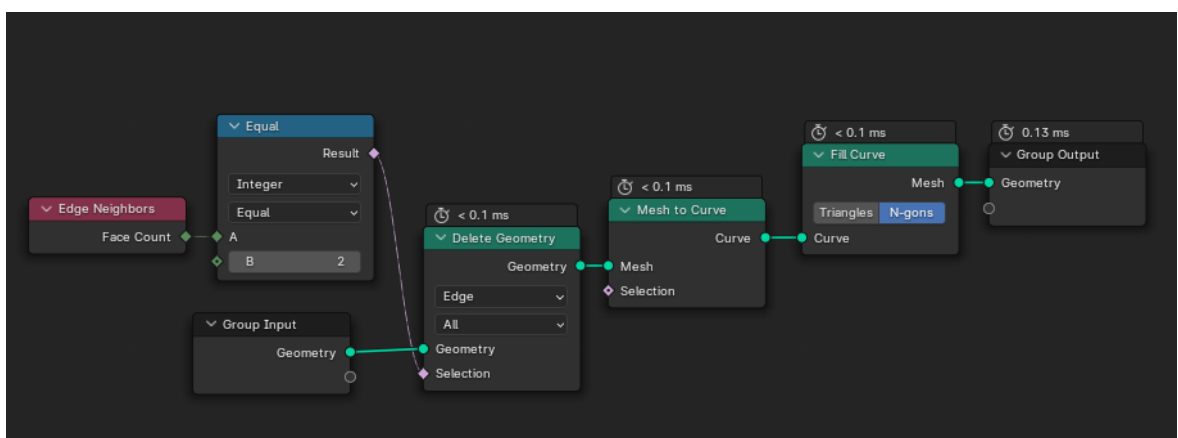
Obrázek 20 Rozmístěné body na ploše Grid

Následně výstup Points z tohoto uzlu je zapojen do geometry vstupů uzlů Sample Nearest a Sample Index. Sample Index se přepne z Float na Vector a do vstupu Value se zapojí uzel Position. Kombinace těchto uzlů nám spojí nejbližší pozice rozmístěných bodů a vznikne náhodný tvar. Na konec výstup Value se připojí do vstupu Position uzlu Set Position. Aby se tahle funkce aplikovala na náš grid, musí se výstup Mesh zapojit do vstupu Geometry.

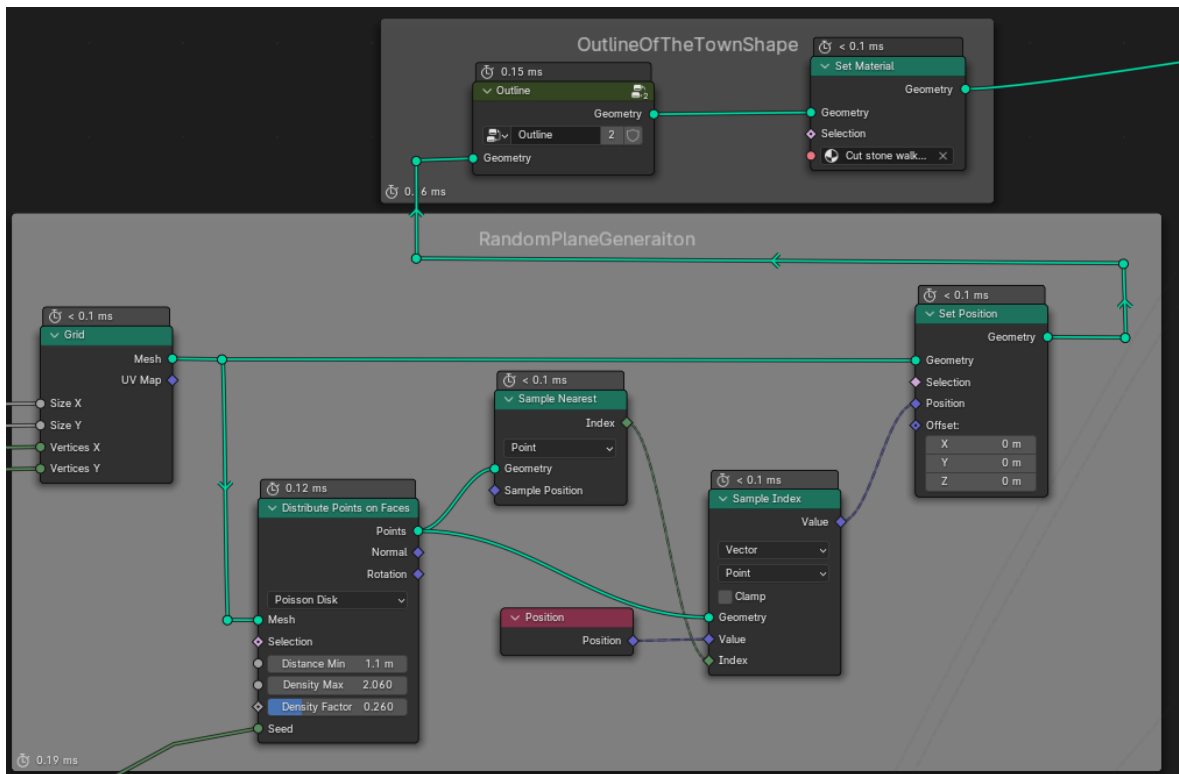


Obrázek 21 Spojené body vygenerované na ploše Grid

Momentálně se v ploše ale nachází hrany mezi jednotlivými body. Tyhle hrany se smažou za pomoci uzlu Delete Geometry, kde je nastaveno mazání na Edge. Výstup Geometry uzlu Set Position se zapojí do vstupu Geometry. Aby se smazaly pouze hrany uvnitř a zůstal obrys náhodného tvaru, tak do vstupu Selection je zapojen uzel Compare. Vstup uzlu je přepnutý na Integer a operaci Equal. Do vstupu A je zapojen uzel Edge Neighbors. Tímhle je získán obrys, který je následně převeden na křivku za pomoci uzlu Mesh to Curve. Křivka se vyplní uzlem Fill Curve přepnutým na N-gons a aplikuje se materiál uzlem Set Material, kde je vybrán materiál Cut Stone Walkaway. Celé tohle seskupení uzlů až na uzel Set Material je vloženo do skupiny pomocí klávesové zkratky Ctrl+G a přejmenováno na Outline, jelikož v dalších částech generátoru bude znovu použito.



Obrázek 22 Seskupení uzlů Outline



Obrázek 23 Generování náhodného tvaru města

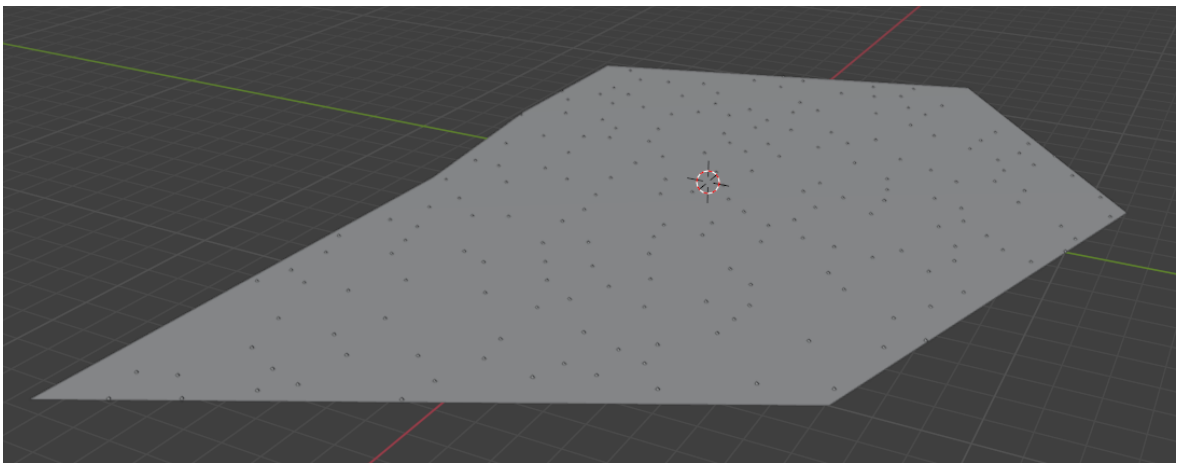
Plocha města je rozdělena do čtyř částí a to centrum (Center), předměstí (Suburb), vzdálené předměstí (Distant Suburb) a okolí (Outskirt). Aby si uživatel mohl měnit velikost těchto čtyř částí dle jeho preferencí, tak výstup Geometry z vygenerovaného náhodného tvaru je připojeno čtyřikrát do pokaždé nově přidaného uzlu Scale Elements. Poté jsou přidány nové vstupy uzlu Group Input. Při generování náhodného tvaru města byl popsán jeden ze způsobů vytvoření vstupu, ale existují i jiné možnosti. Pomocí tlačítka **N** se zobrazí postranní panel, kde v záložce Group je možné vytvořit vstup kliknutím na ikonu plus (+). Pro každý vstup Selection uzlů Scale Elements bude vytvořen nový vstup s názvem Scale a části města. Každý výstup Geometry čtyř nově přidaných uzlů Scale Elements bude použit v následujících částech generátoru.

4.2 Centrum Města

Pro generování města je nezbytné mít k dispozici kolekci 3D modelů budov. Tyto modely můžeme získat buď tak, že si je ručně vymodelujeme, nebo 3D modely získáme z dostupných online zdrojů. Existuje mnoho webových stránek, jako například Sketchfab, které poskytují velké množství 3D modelů zdarma i za poplatek. Při výběru modelů je důležité dbát na to, aby byly tyto modely dostupné pod licencí, která umožňuje jejich

použití pro tvůrčí, komerční nebo politické účely. Licencí, která umožňuje volné využití modelů je Creative Commons (CC).

Po úspěšném získání 3D modelů budov budou tyto modely vloženy do již náhodně vytvořené plochy centra města. Nejprve je nutné vygenerovat body na téhle ploše města. Toho je dosaženo pomocí uzlu Distribute Points on Faces. Výstup Geometry uzlu Scale Elements generující plochu pro centrum města, je následně připojen do vstupu Mesh uzlu Distribute Points on Faces, čímž se náhodně vygenerují body rozložené po celé ploše. Taky se přidá nový vstup uzlu Group Input s názvem Scale of Center, který je připojen do vstupu Scale uzlu Scale Elements.

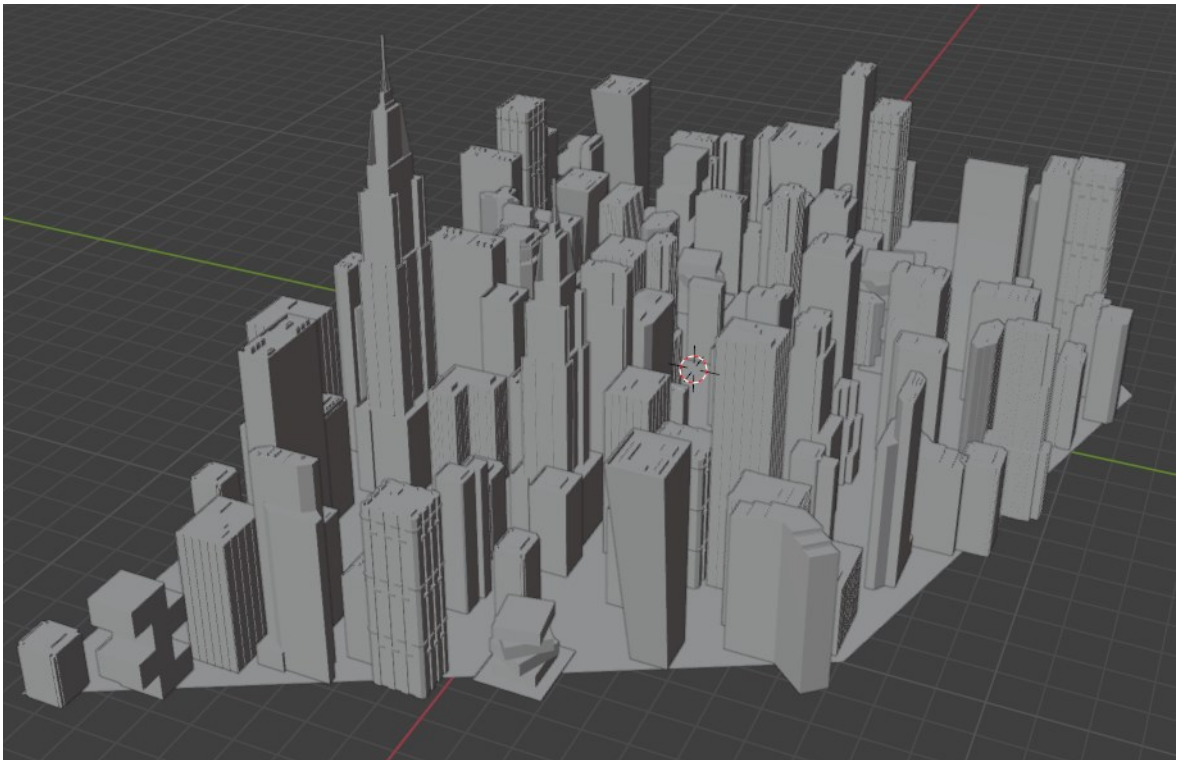


Obrázek 24 Vygenerované body na ploše centra města

Poté co jsou body rovnoměrně rozloženy, jsou na tyto body rozmístěny 3D modely budov. K tomu slouží uzel Instance On Points. Tento uzel vytváří instanci objektů na zvolených bodech. Instance mohou odkazovat na objekt nebo kolekci.

Důvod proč používat Instances je ten, že nám umožňuje zahrnout mnohem více geometrie, aniž by se skutečná data duplikovala. Funguje to díky vykreslovacím enginům jako je Cycles, který dokáže zpracovávat stejná data na mnoha různých místech lépe, než kdyby byla data duplikována. Instances musí být kopií přesně stejné věci, jakmile se stanou Instances nemůžeme je již individuálně upravovat. To znamená, že pokud se provede jakákoliv změna, tak se změny aplikují na všechny kopie, a proto je použit uzel Instance On Points. Po vložení tohoto uzlu je Výstup Points z uzlu Distribute Points on Faces připojen ke vstupu Points z uzlu Instance on Points. Kolekce objektů je připojena pomocí uzlu Collection Info, kde je vybrána kolekce ArtDecoBuildings. Dále je zaškrtnuto políčko Separate Children, Reset Children a režim je přepnut z Original na Relative. Seperate

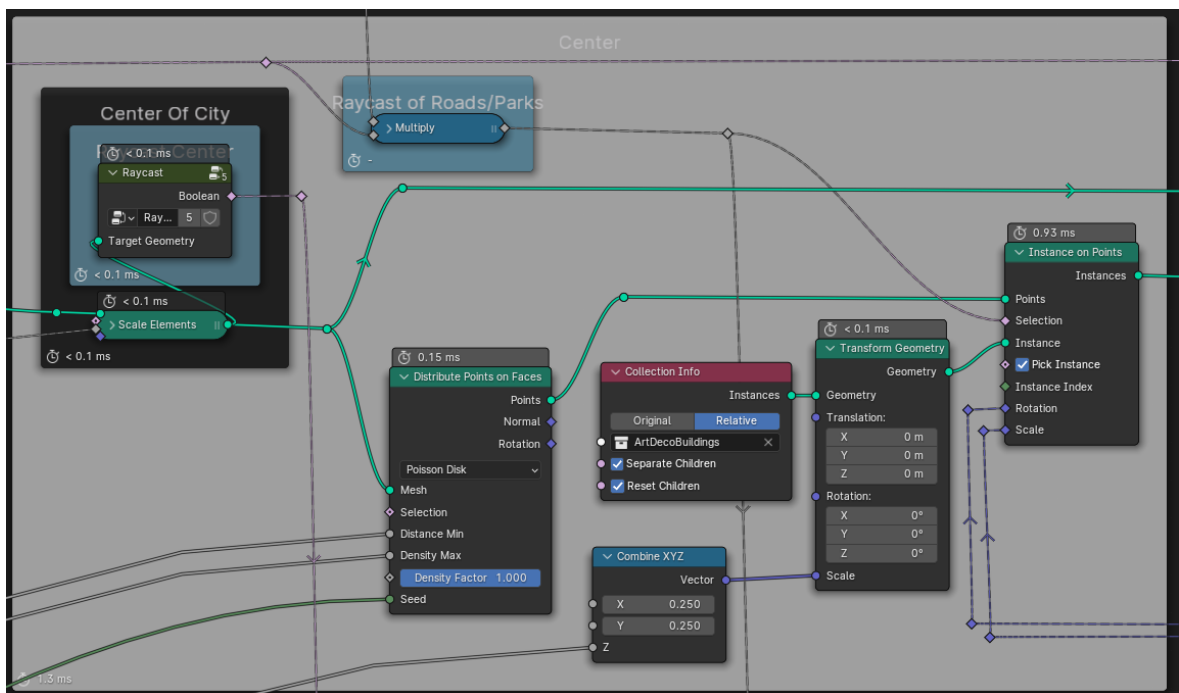
Children předává výstup každého potomka kolekce jako samostatnou instanci a Reset Children umožňuje později aplikovat náhodnou transformaci na každou vybranou instanci. Aby se modely budov nepřekrývaly, tak v uzlu Distribute Points on Faces je změněno v seznamu Distribution Method z Random na Poisson Disk. Poisson Disk nechává minimální vzdálenost mezi náhodně rozmístěnými body, tím pádem se modely budov nepřekrývají. Minimální vzdálenost mezi budovami se může měnit pomocí vstupu Distance Min.



Obrázek 25 Rozmístěné budovy na vygenerovaných bodech v centrum města

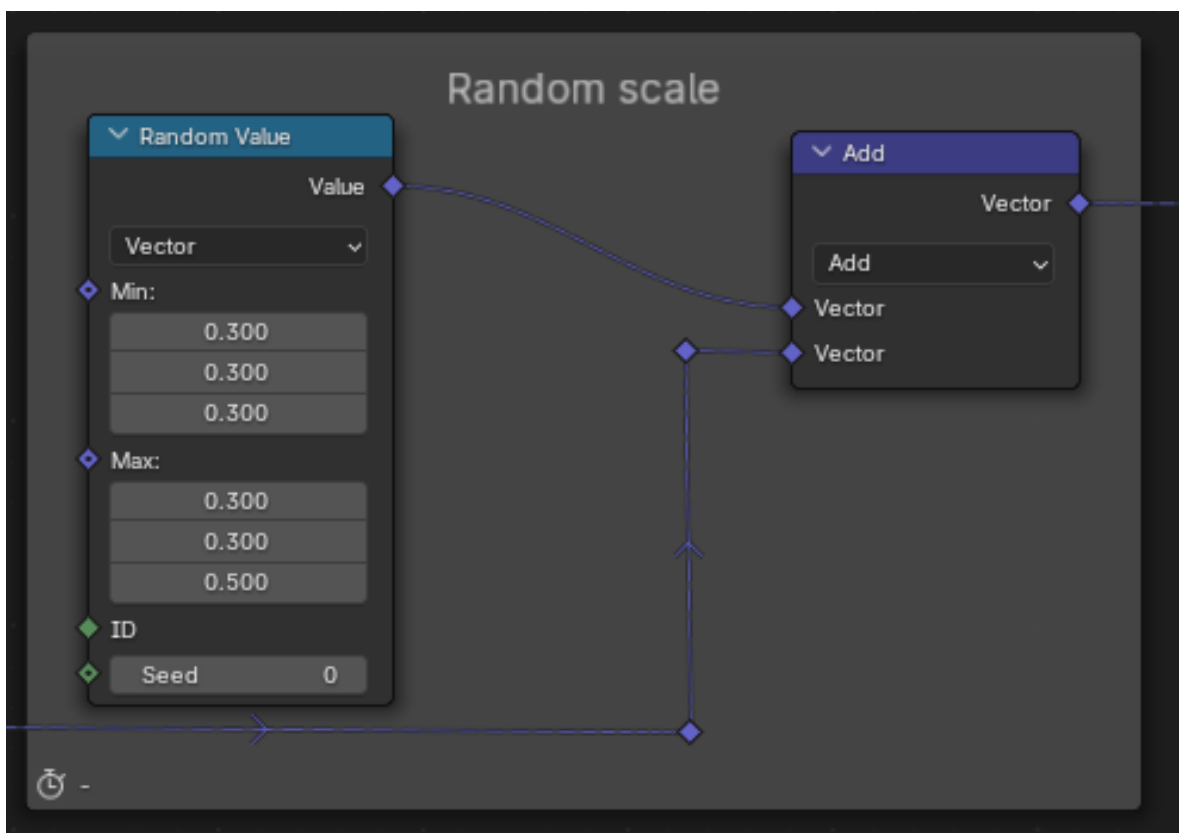
Aby si uživatel mohl nastavit svoji vzdálenost, je přidán nový vstup uzlu Group Input, který je připojen do vstupu Distance Min a pojmenován Distance of buildings. Čím vyšší hodnota je nastavena, tím větší mezera mezi budovami bude. Další vstupy uzlu Distribute Points on Faces, které jsou použity, se jmenují Density Max a Seed. Vstupní parametr Density Max ovlivňuje hustotu budov. Bude pro něj přidán nový vstup uzlu Group Input s názvem Density of buildings a připojen do vstupu Density Max.

Seed v Blenderu je číslo, které určuje počáteční stav generátoru a ovlivňuje posloupnost pseudonáhodných čísel. Blender používá generátory pseudonáhodných čísel pro dosažení reprodukovatelných výsledků. Změnou Seedu v tomhle případě vygeneruje nové rozmístění budov. Proto pro Seed je také přidán nový vstup uzlu Group Input, aby si uživatel mohl najít vyhovující Seed rozmístění budov.



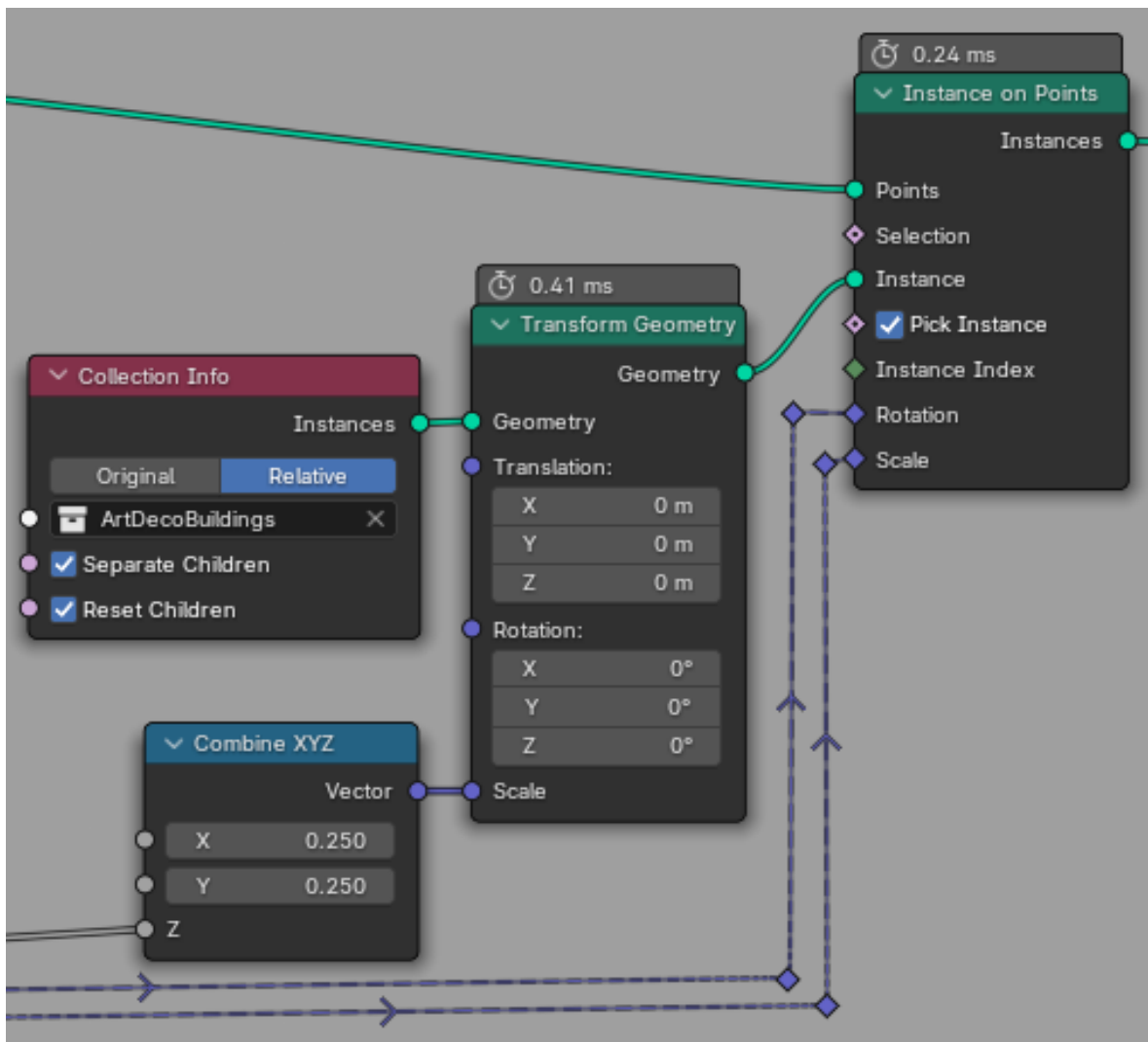
Obrázek 26 Zapojení Geometry Nodes pro vygenerování centrum města

Pro náhodnou transformaci budov je použit uzel Random Value. V uzlu je vybrán z nabídky atribut Vector. V sekci Min jsou nastaveny minimální hodnoty pro transformaci budov na osách X, Y a Z na 0.3. V sekci Max jsou hodnoty nastaveny stejně až na hodnotu Z, která je 0.5. Jelikož v další části bude ještě přidán XYZscaler, je nutné přidat nový vstup Vector Math, který bude nastaven na funkci Add, aby se mohly tyto dva vstupy ovlivňující výšku budov spojit. Výstup z uzlu Random Value se tedy zapojí do prvního vstupu uzlu Add. Druhý vstup bude zatím prázdný, dokud se do něj nezapojí výstup z funkce XYZscaler. Teď už stačí jen výstup uzlu Add zapojit do vstupu Scale uzlu Instance on Points a tím se aplikuje náhodná přednastavená transformace objektů budov.



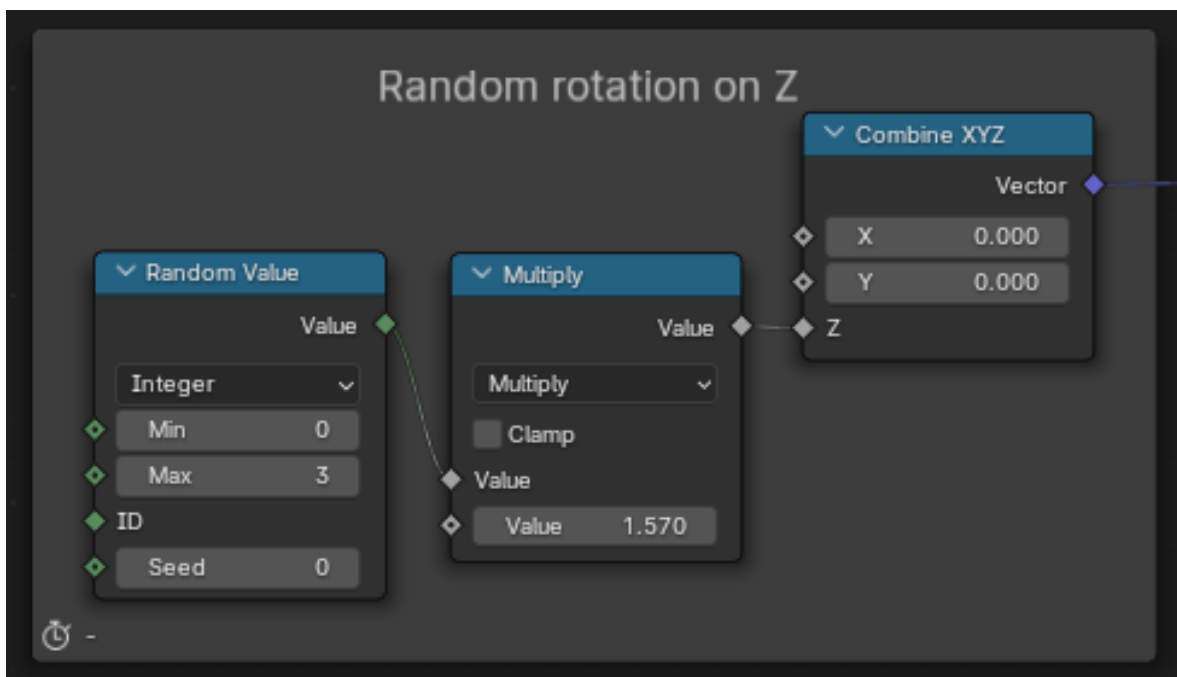
Obrázek 27 Náhodná transformace budov

K náhodné transformaci je přidán i nový vstup Height of the building uzlu Group Input. Díky tomuto vstupu bude možné si měnit náhodně vygenerovanou výšku budov. Mezi připojení uzlu Collection Info a Instance on Points je vložen nový uzel Transform Geometry. Vstup Height of the buildings je zapojen do vstupu Scale. Nyní je ale hodnota aplikována na osy X, Y a Z. Jelikož se jedná o transformaci pouze výšky budov, je mezi tohle propojení vložen nový uzel Combine XYZ, kde vstup Height of the buildings je připojen do vstupu Z. Hodnoty X a Y jsou nastaveny na 0.250.



Obrázek 28 Vstup Height of the buildings

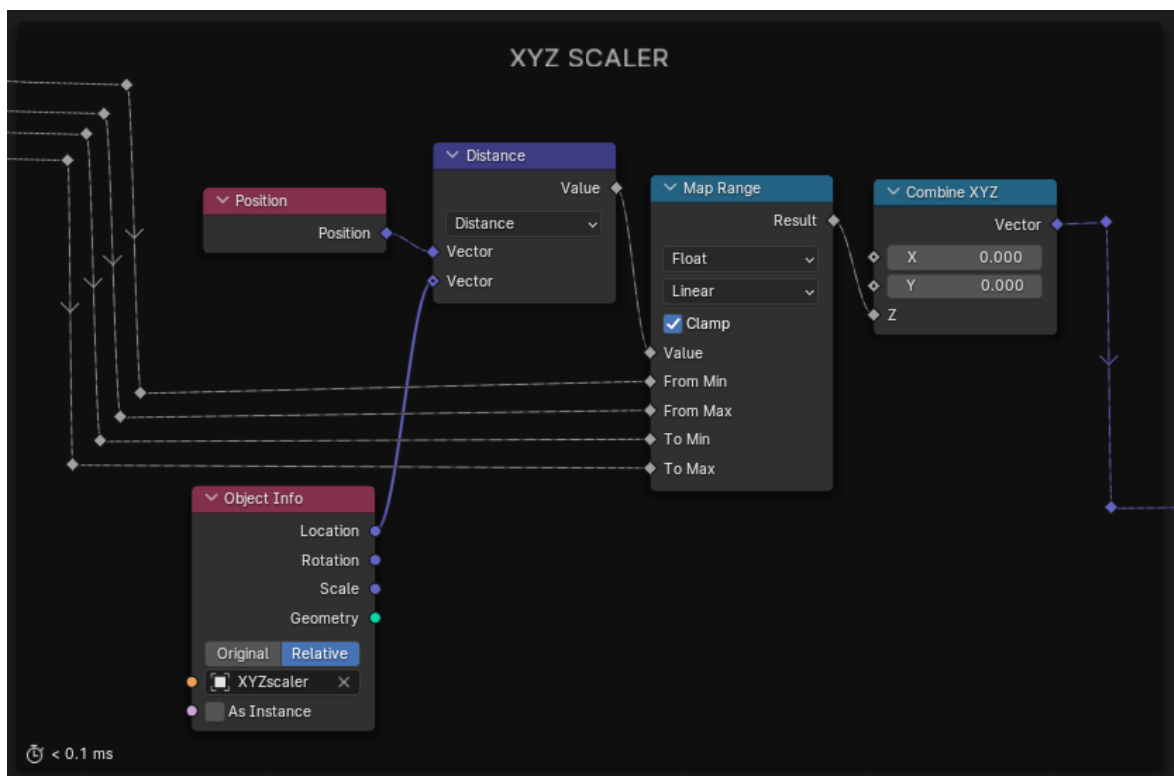
Kromě transformace budov je přidána i rotace pomocí uzlů Random Value, Multiply a Combine XYZ. První je vložen uzel Random Value, kde je v nabídce vybrán atribut Integer. V poli Min je nastavena hodnota 0 a v poli Max 3. Výstup Value je zapojen do nového uzlu Multiply, kde je hodnota nastavena na 1.570. Následně výstup Value uzlu Multiply je připojen do vstupu Z uzlu Combine XYZ. Tím se náhodná rotace na všechny strany aplikuje pouze na osu Z. Výstup Vector uzlu Combine XYZ je připojen do vstupu Rotation uzlu Instance on Points a rotace je aplikována.



Obrázek 29 Náhodná rotace budov na ose Z

Pro ještě větší modifikaci města byl přidán XYZscaler. Jedná se o objekt typu Plain Axes, který umožňuje úpravu výšky budov na místech, kde je umístěn. Pro měření vzdálenosti mezi dvěma body je použit uzel Vector Math. Tento uzel je nastaven na operaci Distance a vstupy jsou propojeny tak, že do prvního je připojen uzel Position. Position vypisuje vektor každého bodu geometrie, ke které je uzel připojen. Do druhého vstupu je připojen uzel Object Info, ve kterém je vybrán objekt XYZscaler a je přepnutý na Relative. Tímhle propojením uzlu Vector Math byla změřena vzdálenost mezi těmito dvěma vektory. Následně je tato vzdálenost připojena výstupem Value uzlu Distance do uzlu Map Range, který přemapovává hodnoty z jednoho rozsahu do druhého cílového rozsahu. Tento uzel má čtyři vstupy. První dva From Min a From Max určují vzdálenosti, zatímco druhé dva To Min, a To Max určují měřítko zvětšení výšky budov. Momentálně se výpočet aplikuje na všechny osy. Pro ovlivnění výšky budov pouze na ose Z je použit uzel Combine XYZ. Výsledek uzlu Map Range, reprezentující změněnou výšku budov, je poté připojen

k vstupu Z uzlu Combine XYZ. Pro aplikaci téhle funkce stačí pouze připojit výstup uzlu CombineXYZ do druhého vstupu uzlu Add, který byl vytvořen v předešlém odstavci. Operace Add funguje jako součet. Díky ní se nám jak funkce XYZscaler tak i náhodná transformace spojí do jednoho výstupu. Výstup uzlu Add již byl zapojen do vstupu Scale uzlu Instance on Points, proto není potřeba dalšího zapojení.



Obrázek 30 Funkce XYZscaler

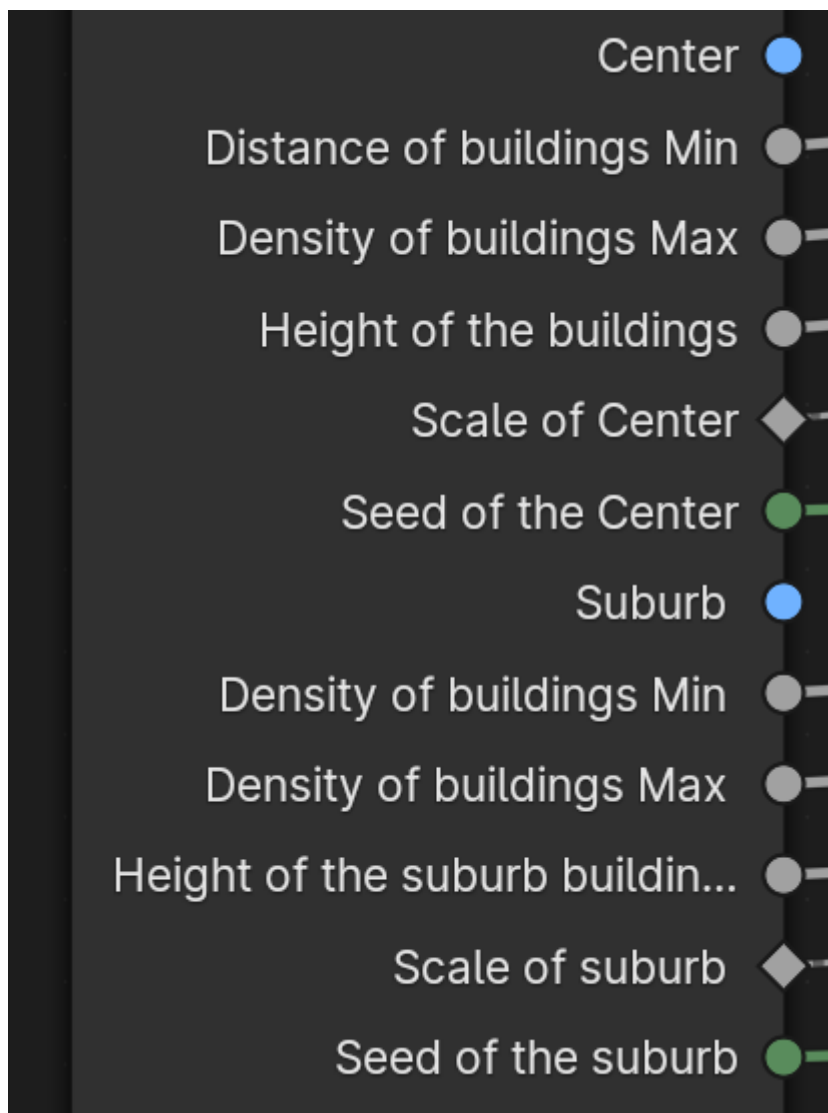
4.3 Předměstí (Suburb)

Při generování předměstí budou použity stejné uzly jako při generování města s několika změnami. První se stejně zapojí výstup uzlu Scale Elements, tentokrát ale druhý uzel generující plochu předměstí. Pro úpravu velikosti je přidán další vstup uzlu Group Input s názvem Scale of Suburb a je zapojen do vstupu Scale.

Následuje stejný postup jako v kapitole 4.2. Nejprve je nutné mít opět nachystanou kolekci budov, které budou rozmístěny na plochu předměstí. Poté se na vytvořenou plochu předměstí rozmístí body pomocí uzlu Distribute Points on Faces. Následně se 3D modely budov z kolekce umístí na tyto body pomocí uzlu Instance on Points. Pro odstranění překrytí budov se změní Distribution Method z Random na Poisson Disk. V uzlu Group Input jsou přidány nové vstupy, které se zapojí do vstupu Distance Min, Density Max a Seed uzlu Distribute Points on Faces. Kolekce objektů se připojí pomocí uzlu Collection

Info a vybere se opět kolekce ArtDecoBuildings. Nakonec se opět zaškrtnou pole Separate Children, Reset Children a režim je přepnut z Original na Relative.

Pro lepší přehlednost se také přidá nový vstup uzlu Group Input typu String. Tento vstup bude vždy před vstupy k jednotlivým částem města. Nebude se nikam zapojovat, pouze se změní jeho název, který se bude rovnat části města. Tím pádem se pojmenuje Suburb.

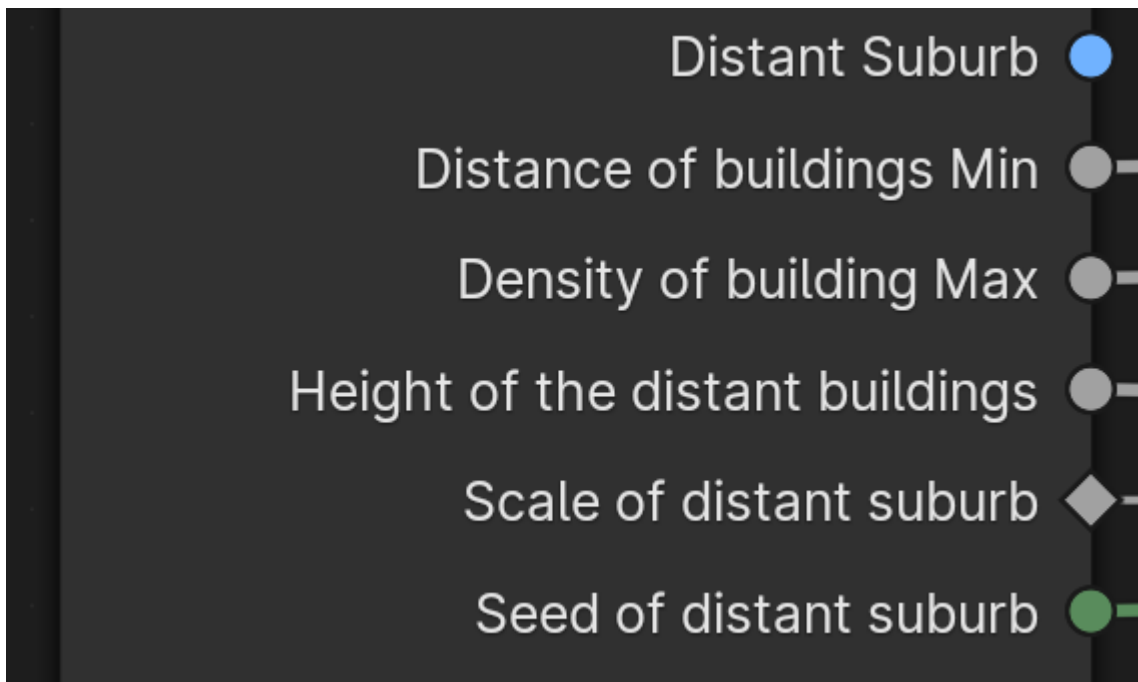


Obrázek 31 Ukázka uzlu Group Input pro centrum a předměstí

4.3.1 Vzdálené předměstí (Distant Suburb)

Pro ještě reálnější generování a úpravu města, je část generování předměstí jakoby z duplikována a tím rozdělena na dvě části. K rozmístění a vygenerování budov je použit stejný postup, jak v části předměstí. Pouze v uzlu Collection Info se vybere kolekce Suburbs Buildings namísto ArtDecoBuildings. V téhle nově vybrané kolekci Suburbs Buildings už nejsou vysoké budovy jako mrakodrapy, ale spíše menší budovy zapadající

do předměstí. Dále bude použit třetí uzel Scale Elements, který slouží ke generování plochy vzdáleného předměstí. Ještě se přidá nový vstup uzlu Group Input s názvem Scale of Distant Suburb a zapojí se do vstupu Scale třetího uzlu Scale Elements. Ostatní vstupy, které byly přidány v části předměstí, jsou přidány i tady, jen jsou změněny názvy. Nyní je uživatel schopen měnit velikost předměstí jako dvě části a díky tomu má více možností, aby si přizpůsobil město svým představám.



Obrázek 32 Ukázka uzlu Group Input pro vzdálené předměstí

4.4 Okolí (Outskirt)

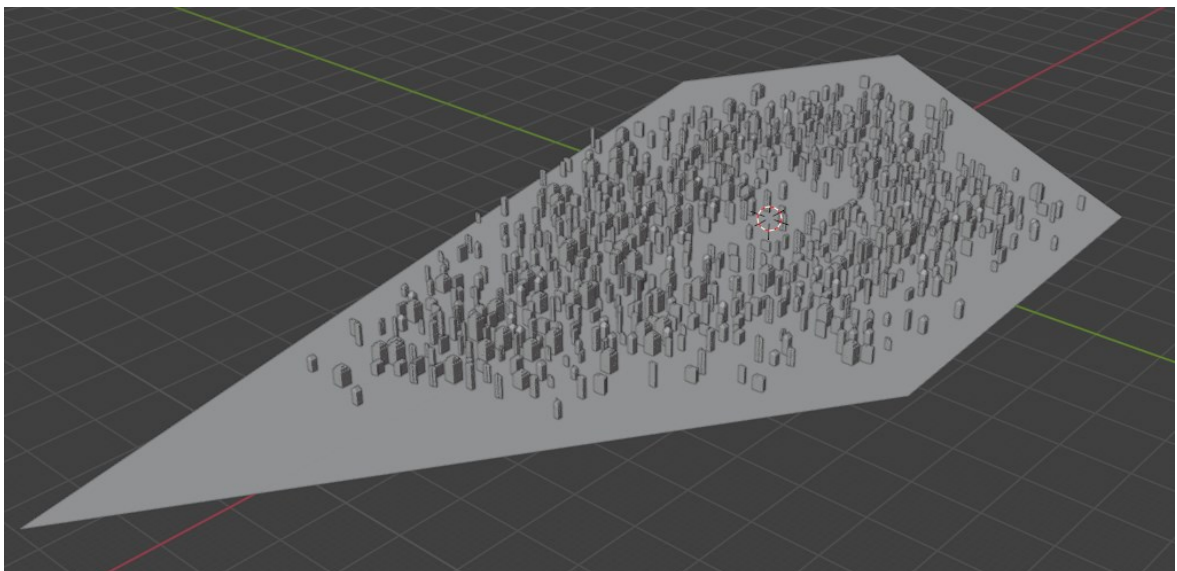
Čtvrtá a poslední částí generovaného města je okolí. První se stejně jako v předchozích částech města opět zapojí výstup teď už posledního uzlu Scale Elements. Pro úpravu velikosti plochy, se vytvoří nový vstup uzlu Group Input a je pojmenován Scale of Outskirt. Zapojení uzlů pro vygenerování budov se nemění a budou použity stejné uzly až na uzel Collection Info. V tomto uzlu se vybere kolekce OutSkirtBuildings, ale nastavení zůstane stejné.

V této části se budovy opět generují po celé ploše. Tímto způsobem by ale město ztratilo reálný aspekt. Proto byla přidána kombinace několika uzlů, která implementuje funkci pro generování budov s postupně se zmenšující hustotou od středu města směrem ven.

Jako první je přidán uzel Delete Geometry, pomocí kterého budou mazány rozmístěné budovy po celé ploše, aby byl dosažen tento efekt. Uzel je přidán mezi zapojení z uzlu

Distribute points on Faces a Instance on Points. Jelikož jsou smazány body, na kterých budou budovy umístěné, je přepnuto na smazání Point a All v uzlu Delete Geometry. Do vstupu Selection je zapojen nový uzel Random Value, který má nastavený atribut na Boolean. Následně je přidán uzel Map Range, pomocí kterého funguje celá myšlenka mazání od středu směrem ven. Nejprve jsou potřebné body, od kterých se bude mazat. Proto je použit uzel Geometry Proximity, který vypočítává vzdálenost od připojeného objektu. Geometry Proximity je přepnut na vypočítávání od Edges a do vstupu Target je připojen výstup uzlu Scale Elements z části distant Suburb. Tenhle výstup je připojen z části Distant Suburb, protože navazuje na část Outskirt. Dále je důležité zapojit uzel Position do vstupu Source Position uzlu Geometry Proximity, jelikož udává polohu připojeného geometrického objektu.

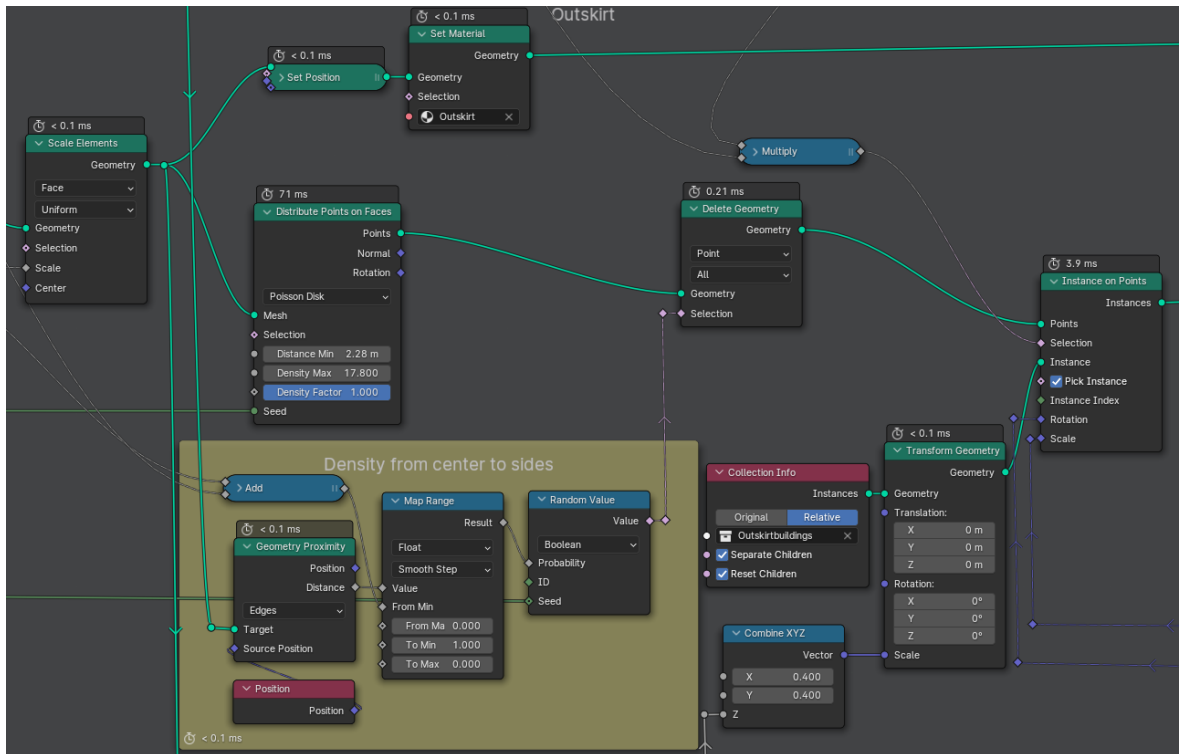
Nyní je nutné nastavit uzel Map Range. Nejprve je výpočet přepnut z Linear přechodu na Smoother Step, kde název napovídá, že přechody budou hladší. Pokud je nastavena hodnota From Min menší a hodnota From Max větší, budou se body mazat od křivky směrem ven. V tomhle případě je potřeba, aby se body mazaly přesně naopak a tím byl dosažen efekt zmenšující se hustoty budov od středu města směrem ven. Proto je hodnota From Min nastavena na větší než hodnota From Max.



Obrázek 33 Ukázka funkce generování budov od středu směrem ven

K tomu, aby se vždy přizpůsobila hodnota hustoty tvaru města, je zapojen výstup Scale of Outskirt do uzlu Add jako první hodnota. Tento uzel slouží k tomu, aby si uživatel mohl změnit hustotu dle svých představ. Proto pomocí nově přidaného vstupu uzlu Group Input s názvem Density of buildings from Center, který je zapojen do druhého vstupu uzlu Add,

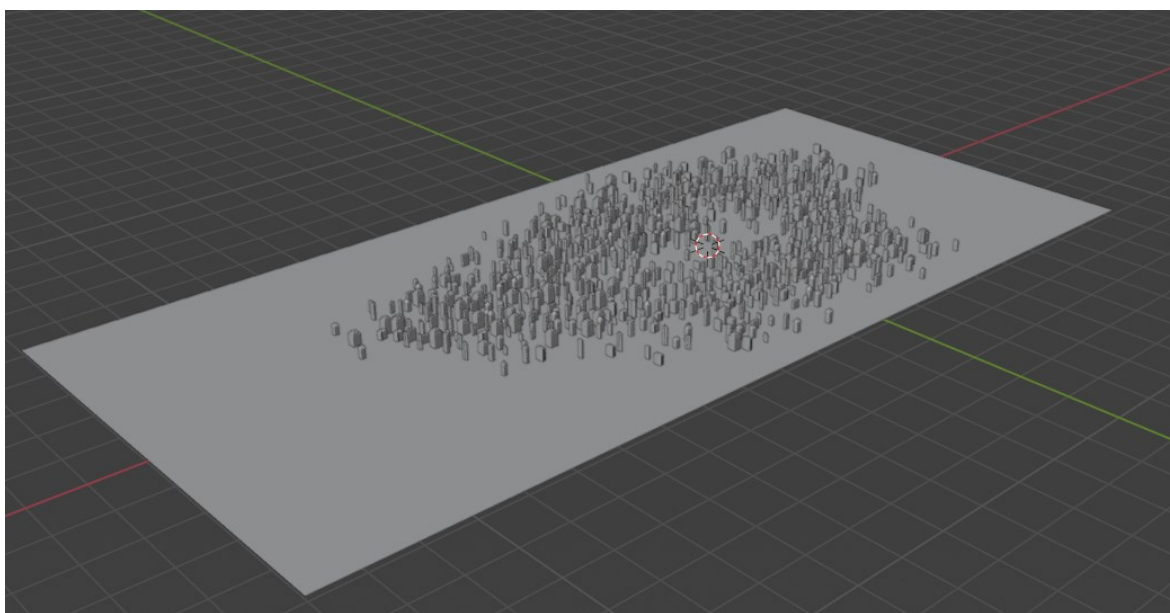
jí může ovlivňovat. Výstup uzlu Add je zapojen do vstupu Value uzlu Map Range. Nyní se tedy bude generovat pokaždé stejná hustota budov od centra směrem ven podle vygenerovaného tvaru města, dokud není hodnota Density of buildings from center změněna.



Obrázek 34 Zapojení funkce generující budovy od centra směrem ven

4.4.1 Ohraničení města

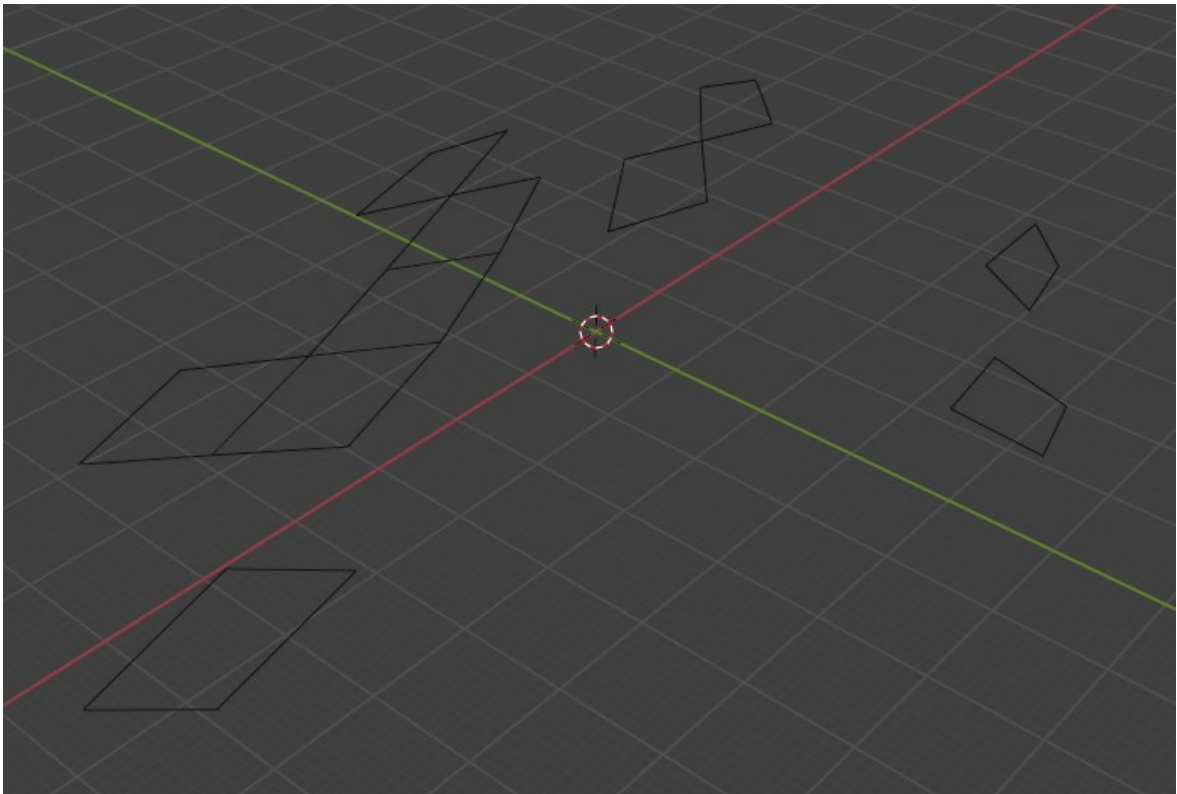
Ted se město pokaždé vygeneruje jako náhodný tvar. V předešlém odstavci byla přidána funkce, pro vytvoření efektu mizících budov od centra směrem ven. Aby celkově město působilo na pohled lépe, tak je přidána nová funkce, která obalí náhodně vytvořený tvar města nově vygenerovaným objektem. Je toho dosaženo uzlem Bounding Box. Tento uzel vytvoří nový objekt Box Mesh, který obklopuje připojený Geometry vstup s minimálním požadovaným objemem. Zjednodušeně řečeno objekt, který připojíme do uzlu Bounding Box, tak se nám vloží do perfektně sedící krabice. Plocha města je objekt Plane, což je vlastně rovný povrch definovaný svými vrcholy v rovině XY, takže nemá žádnou stanovenou výšku. Proto bude použit uzel Bounding Box, který nám náhodný tvar plochy města zarámuje novou rovinou. Poslední přidanou částí byla část Outskirt, takže bude použit výstup Scale Elements z této části. Ten je zapojen do vstupu uzlu Bounding Box. Výstup je zapojen do uzlu Set Material, kde je vybrán materiál Outskirt.



Obrázek 35 Ukázka uzlu Bounding Box

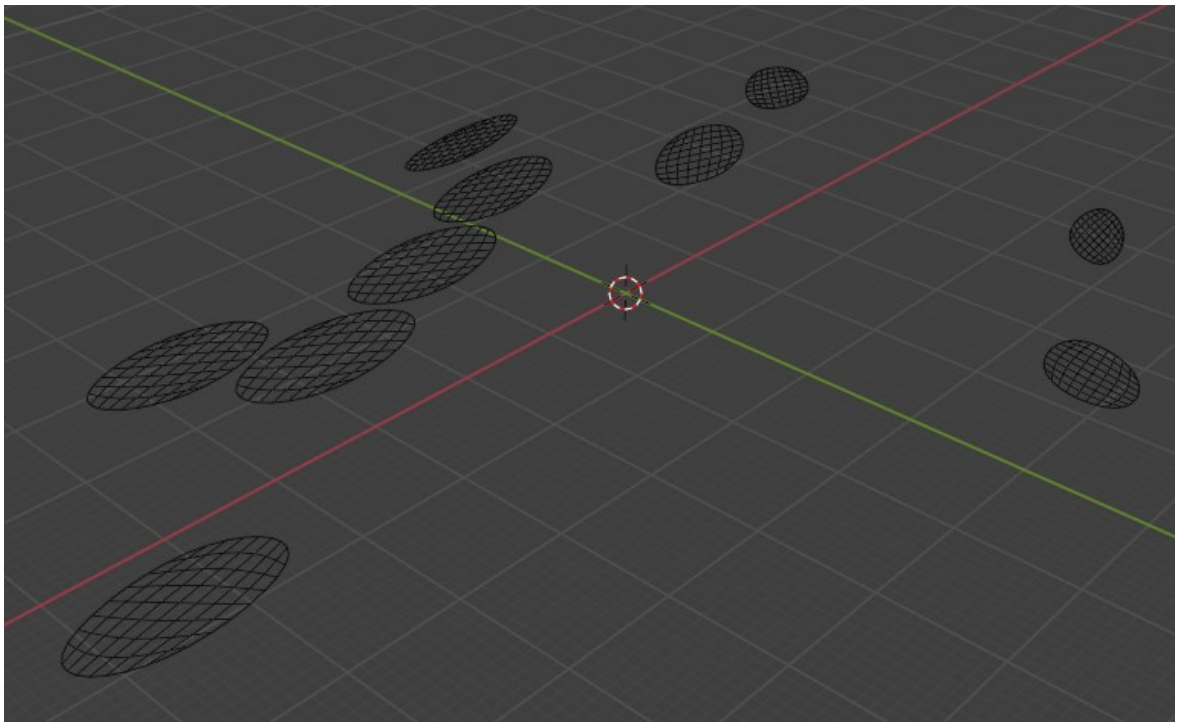
4.5 Parky

Generování parků bude pouze na území všech tří částí města. Díky tomu, že všechny části města vycházejí z jednoho tvaru, který je pouze násoben pomocí vstupu Scale uzlu Scale Elements, stačí použít tu největší část Outskirt. Tato část plochy města je rozdělena pomocí uzlu Subdivision Surface. Výstup Scale Elements části Outskirt je připojen do vstupu Mesh uzlu Subdivision Surface. Parametr Level je nastaven na 3, což poskytuje ideální poměr ploch, se kterými lze dále pracovat. Výstup je následně připojen do nového uzlu Delete Geometry, ve kterém je nastaveno smazání na Face a All. Do vstupu Selection je připojena kombinace dvou uzlů. Prvním je Random Value, který je přepnutý na atribut Boolean, jehož výstup je zapojen do uzlu Map Range, který je v režimu Smoother Step s nezměněnými základními hodnotami.

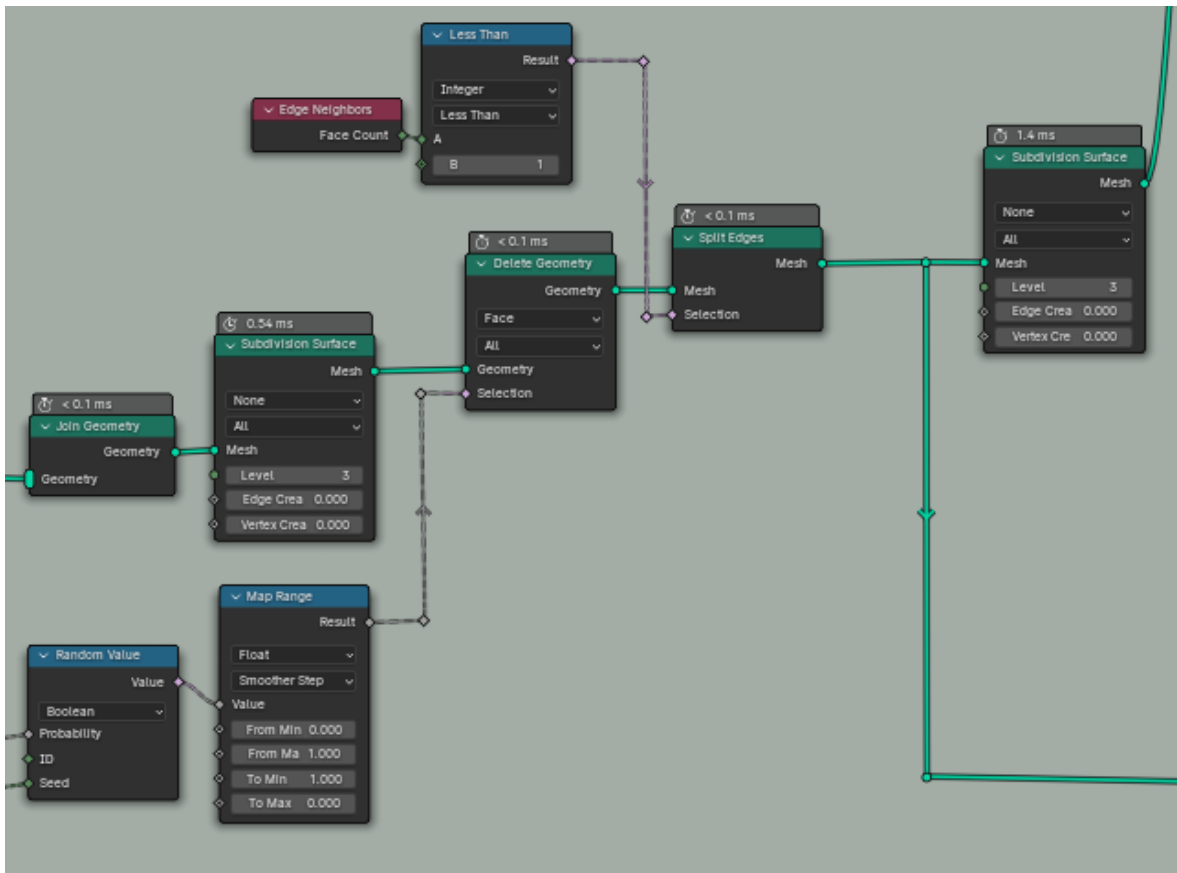


Obrázek 36 Pozůstalé plochy po náhodném odstranění

Nyní se většina ploch smazala a pár náhodně rozmístěných zůstalo. Tyhle plochy budou využity pro generování parků, ale některé jsou spojeny a mají společné hrany. Pro jejich rozpojení je výstup uzlu Delete Geometry připojen do uzlu Split Edges. Tento uzel slouží k rozdělení a duplikaci hran, čímž se přeruší společná část a získají se samostatné plochy. Do vstupu Selection je připojen uzel Compare s přepnutým typem vstupu na Integer a operací Less Than. Přidán je také uzel Edge Neighbors, pomocí kterého je zjištěno, kolik ploch má společnou hranu. Výstup Face Count je připojen do vstupu A. Vstup B je poté nastaven na 3. Všechny plochy parků teď mají ostré hrany, proto za pomoci nově přidaného dalšího uzlu Subdivision Surface se zakulatí. Parametr Level je opět nastaven na 3 a do vstupu Mesh je připojen výstup Split Edges.

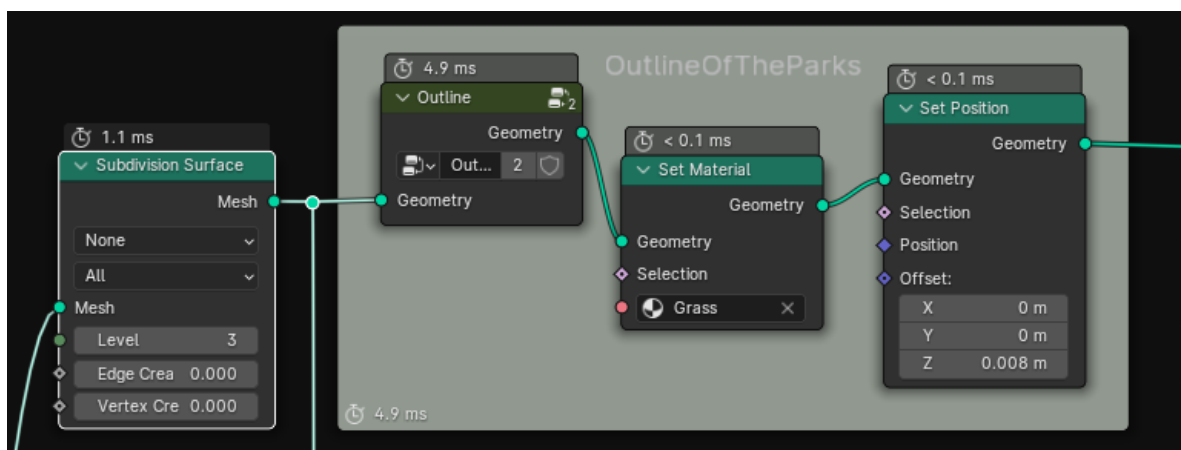


Obrázek 37 Rozpojení společných hran a zakulatění ploch parků



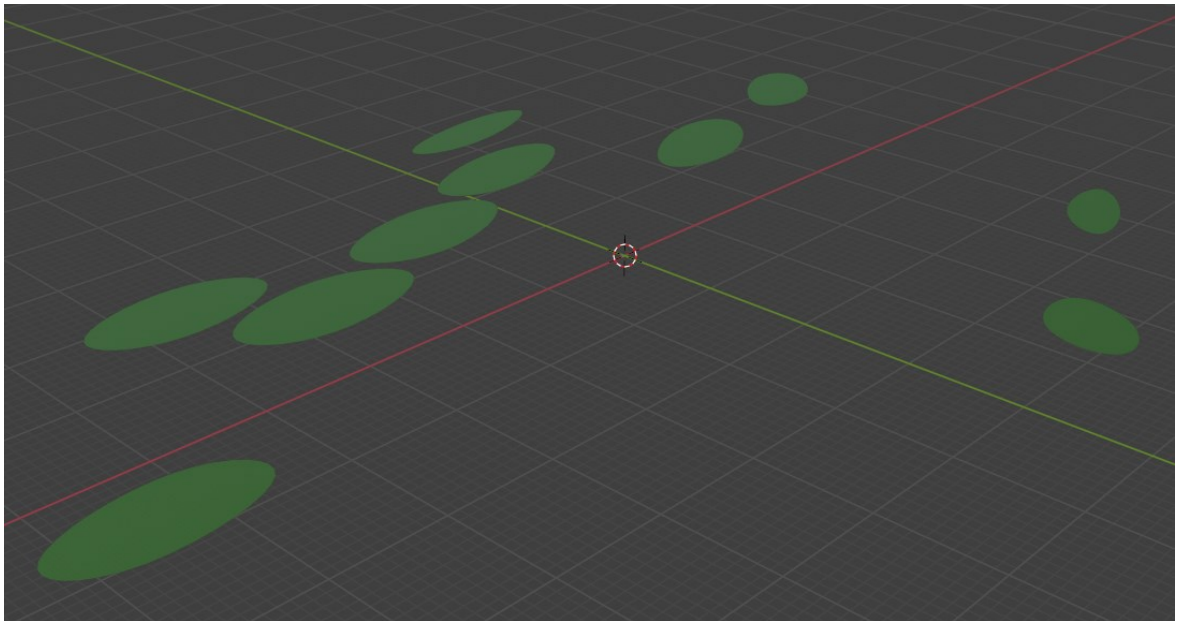
Obrázek 38 Rozpojení společných hran a zakulatění ploch parků

Pro lepší manipulaci se plochy parků zbaví veškerých hran a ploch uvnitř a získá se pouze obrys tvaru. Není potřeba vkládat znovu uzly pro tuhle funkci, jelikož v kapitole 4.1. Plocha města bylo vytvoření seskupení uzlů do skupiny s názvem Outline, která má přesně tuhle funkci. Pomocí klávesové zkratky Shift+A a názvu Outline je tahle skupina uzlů vložena. Výstup posledně použitého uzlu Subdivision Surface je připojen do vstupu naší skupiny Outline. Výstup Outline je připojen do uzlu Set Material, kde je vybrán materiál



Obrázek 39 Zapojení Geometry Nodes pro skupinu uzlů Outline

trávy Grass. Ještě je zapojen výstup uzlu Set Material do uzlu Set Position, ve kterém je nastaven Offset Z na 0.008 m, aby nedocházelo ke kolizi ploch části okolí města a parků.



Obrázek 40 Ukázka vygenerovaných ploch parků s texturou

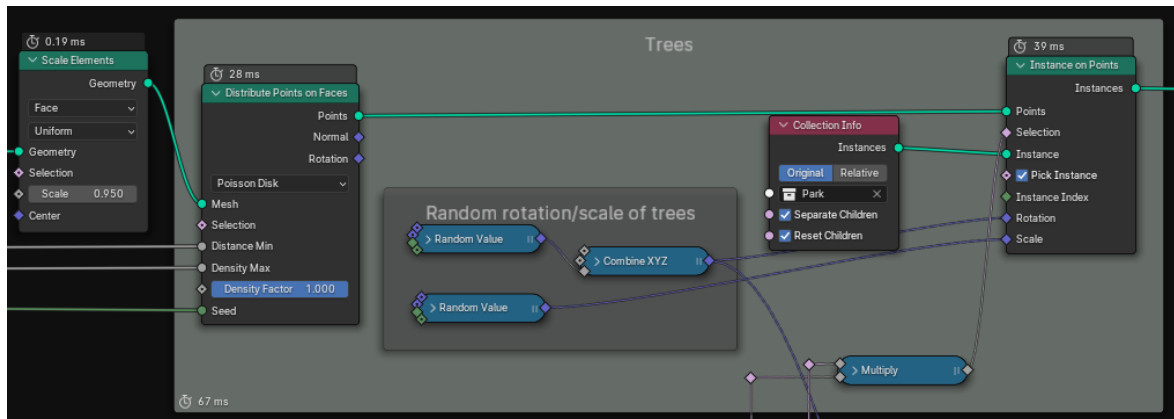
4.5.1 Stromy

Generování stromů bude fungovat stejným způsobem jako generování budov ve městě. Nejprve je výstup uzlu Subdivision Surface zapojen do uzlu Scale Elements, aby se zabránilo zasahování stromů do budov. Hodnota Scale je změněna na 0.950, čímž se zmenší plocha parků, ve kterých budou stromy generovány. Výstup Scale Elements je následně zapojen do uzlu Distribute Points on Faces, kde je metoda změněna na Poisson Disk. Pro lepší modifikaci parků jsou vstupy Distance Min, Density Max a Seed zapojeny stejně jako u částí měst, jako nové vstupy uzlu Group Input, aby si je uživatel mohl měnit. Nové vstupy jsou pojmenovány jako Distance of trees, Density of trees a Seed. Výstup je zapojen do uzlu Instance on Points, ve kterém je zaškrtnuto Pick Instance.

Následně je přidán nový uzel Collection Info, ve kterém je vybrána kolekce Park a zaškrtnuty možnosti Original, Separate Children a Reset Children. Do kolekce Park jsou vloženy modely stromů, které jsou zdarma dostupné a stažené ze stránky Sketchfab.

V uzlu Instance on Points je do vstupu Rotation zapojená náhodná rotace za pomoci uzlu Random Value s atributem Vector. V sekci Max je poslední hodnota nastavena na 180 a výstup uzlu zapojen do vstupu Z uzlu Combine XYZ. Tím je dosažena náhodná rotace v rozmezí 0 až 180 stupňů. Dále je aplikována náhodná transformace za pomoci totožného

uzlu Random Value se stejným atributem. V sekci Min jsou první dvě hodnoty nastaveny na 0.4 a poslední hodnota na 0.2. V sekci Max jsou hodnoty nastaveny na 0.6 a poslední na 0.5. Výstup tohoto uzlu je pak přímo zapojen do vstupu Scale a náhodná transformace v nastaveném rozmezí je aplikována.



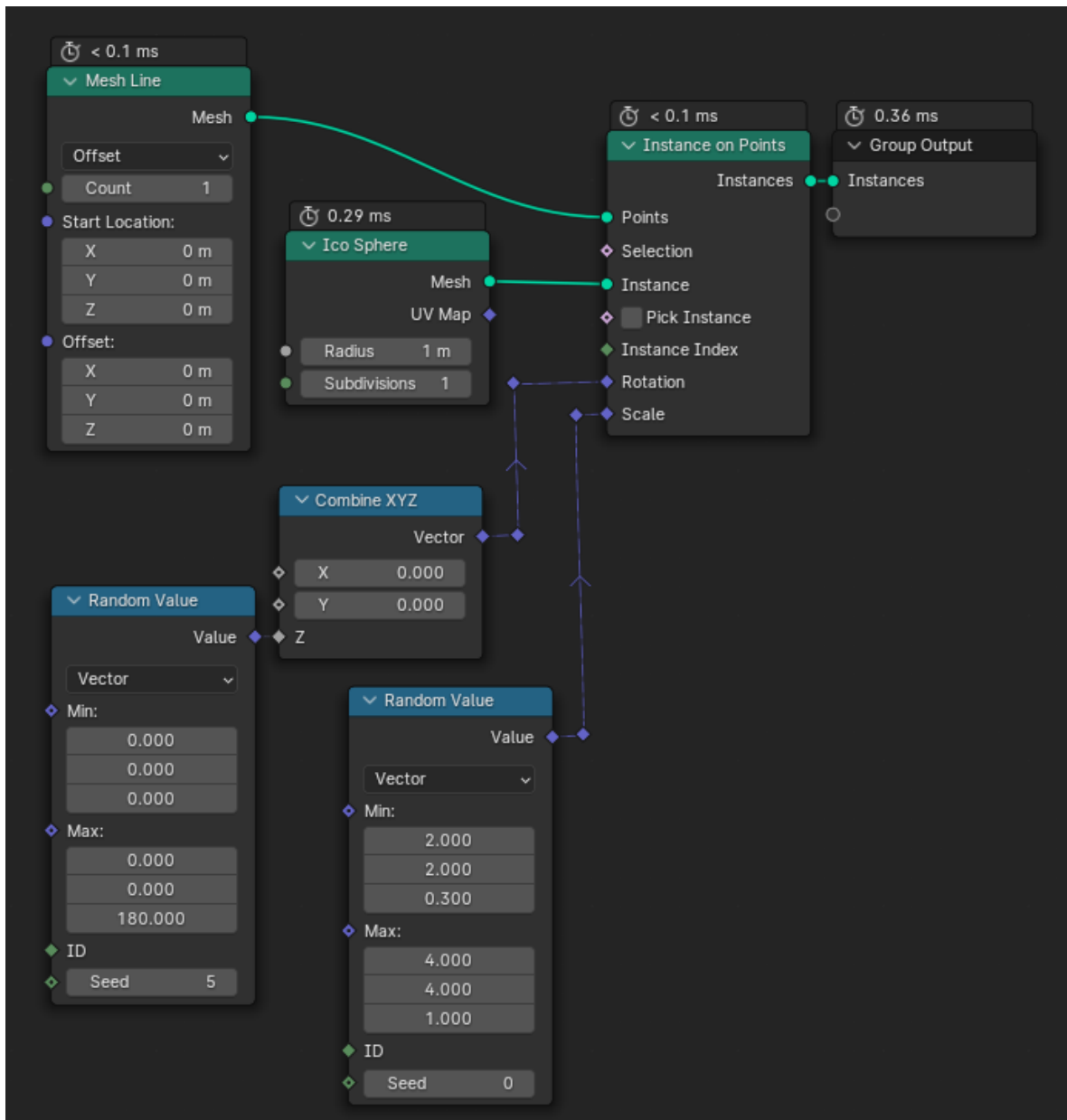
Obrázek 41 Zapojení Geometry Nodes pro generování stromů

4.5.2 Kamenná cesta

Generování kamenné cesty je opět vytvořeno stejně jako u stromů, s několika změnami. Nejprve je výstup uzlu Split Edges připojen do uzlu Subdivision Surface, kde je parametr Level nastaven pouze na 1. Při generování parků se použil level 3, aby byly parky co nejvíce zaobleny. Pro kamennou cestu je použit pouze level 1, čímž vytvoříme pouze jedno rozdělení plochy, tím vzniknou uvnitř plochy 4 hrany, na které budou později umístěny kameny. Výstup z uzlu Subdivision Surface je připojen do uzlu Distribute Points on Faces, kde je metoda změněna z Random na Poisson Disk, aby bylo možné nastavit minimální vzdálenost mezi kameny. Hodnoty Distance Min je nastavena na 0.12m a Density Max na 50, čímž je dosaženo efektu kamenné cesty. Poté je výstup připojen do uzlu Instance on Points. Místo manuálního vytváření kamenů nebo stahování zdarma dostupných modelů je vytvořen jednoduchý generátor kamenů.

Pro generátor kamenů je použit uzel Mesh Line, kde je počet vrcholů nastaven na 1. Výstup je zapojen do uzlu Instance on Points. Do vstupu Instance je připojen uzel Ico Sphere, kde je nastaven Radius na 1 m a Subdivisions na 1. Pro náhodnou rotaci a transformaci kamenů je zapojeno seskupení uzlů, které již bylo vytvořeno při generování stromů. První seskupení uzlů pro náhodnou rotaci na ose Z není třeba měnit nic, takže jsou vloženy stejné propojené uzly Random Value a Combine XYZ do vstupu Rotation. Pro náhodnou transformaci je znovu použit uzel Random Value, kde v sekci Min jsou

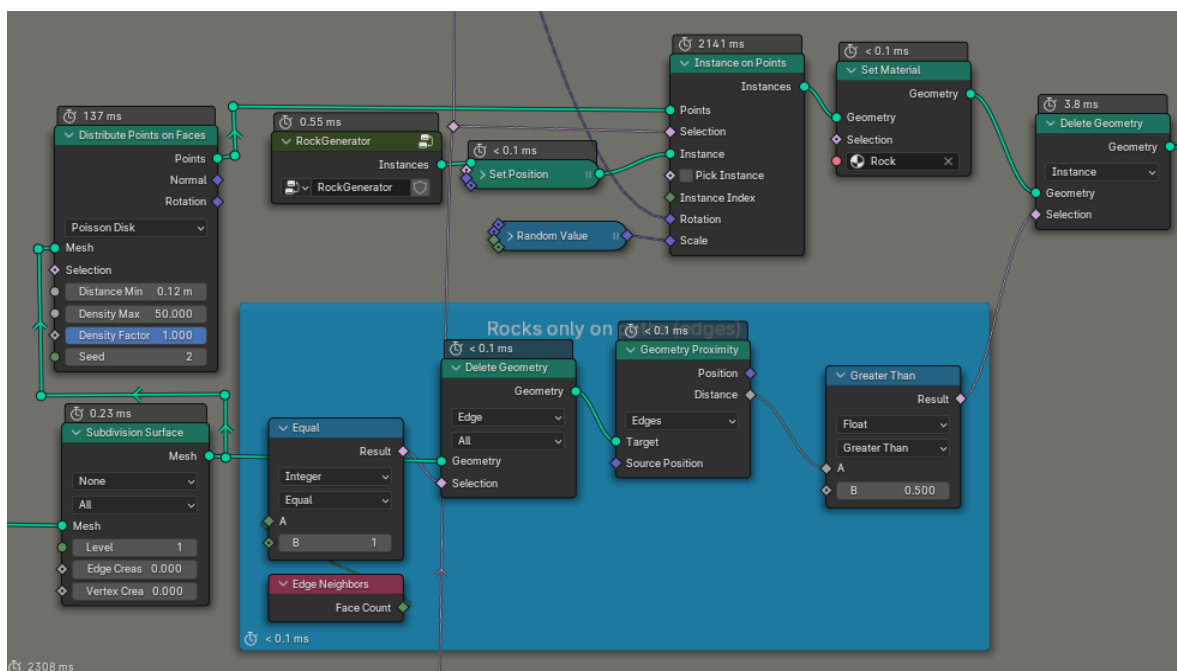
nastaveny hodnoty 2 pro osy X a Y a 0.3 pro osu Z. V sekci Max jsou hodnoty nastaveny na 4 pro osy X a Y a 1 pro osu Z. Výstup tohoto uzlu je připojen do vstupu Scale, čímž je aplikována náhodná transformace v nastaveném rozmezí. Pro lepší přehlednost tohle seskupení uzlů je vloženo do skupiny za pomoci klávesové zkratky Ctrl+G a pojmenováno Rock Generator.



Obrázek 42 Zapojení Geometry Nodes pro generování kamene

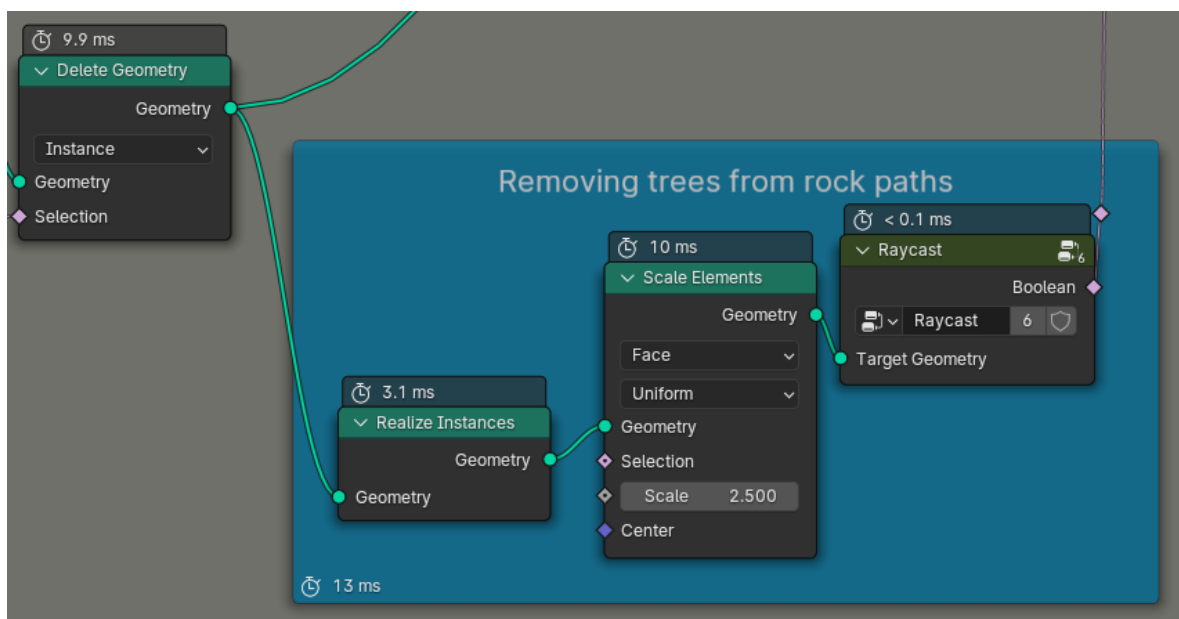
Výstup skupiny Rock Generator je zapojen do uzlu Set Position, kde je v sekci Offset hodnota Z nastavena na 0.4 m, aby byly kameny většinu své výšky pod plochou parků. Výstup uzlu Set Position je následně připojen do vstupu Instance uzlu Instance on Points. Do vstupů Rotation a Scale jsou opět zapojeny stejné uzly jako v předchozím odstavci. Jediná změna nastává u Scale, kde jsou všechny hodnoty v sekci Min nastaveny na 0.03 a v sekci Max na 0.05. Výstup uzlu Instance on Points je následně zapojen do uzlu Set Material s vybraným materiálem Rock.

Pro vygenerování kamenů pouze na hranách uvnitř ploch parků musí být smazány hrany, které tvoří obrys. Ty jsou smazány tak, že do uzlu Delete Geometry je připojen výstup uzlu Subdivision Surface, který byl zapojen v prvním odstavci této kapitoly. Smazání je nastaveno na Edge a All. Do vstupu Selection je připojen uzel Compare s typem Integer a operací Equal. Do vstupní hodnoty A je připojen uzel Edge Neighbors a hodnota B je nastavena na 1. Následně je výstup Delete Geometry připojen do uzlu Geometry Proximity, pomocí kterého je vypočítána vzdálenost od hran, tím pádem je vybráno Edges. Výstup Distance je připojen do uzlu Compare s typem Float a operací Greater Than. Nyní pomocí nového uzlu Delete Geometry budou kameny smazány podle vzdálenosti, která bude zvolena v předešlém uzlu Compare v hodnotě B. Do vstupu Delete Geometry, kde je vybráno Instance, je připojen výstup uzlu Set Material kamene a do Selection je připojen výstup Compare uzlu. Čím vyšší hodnota B, tím širší bude cesta.

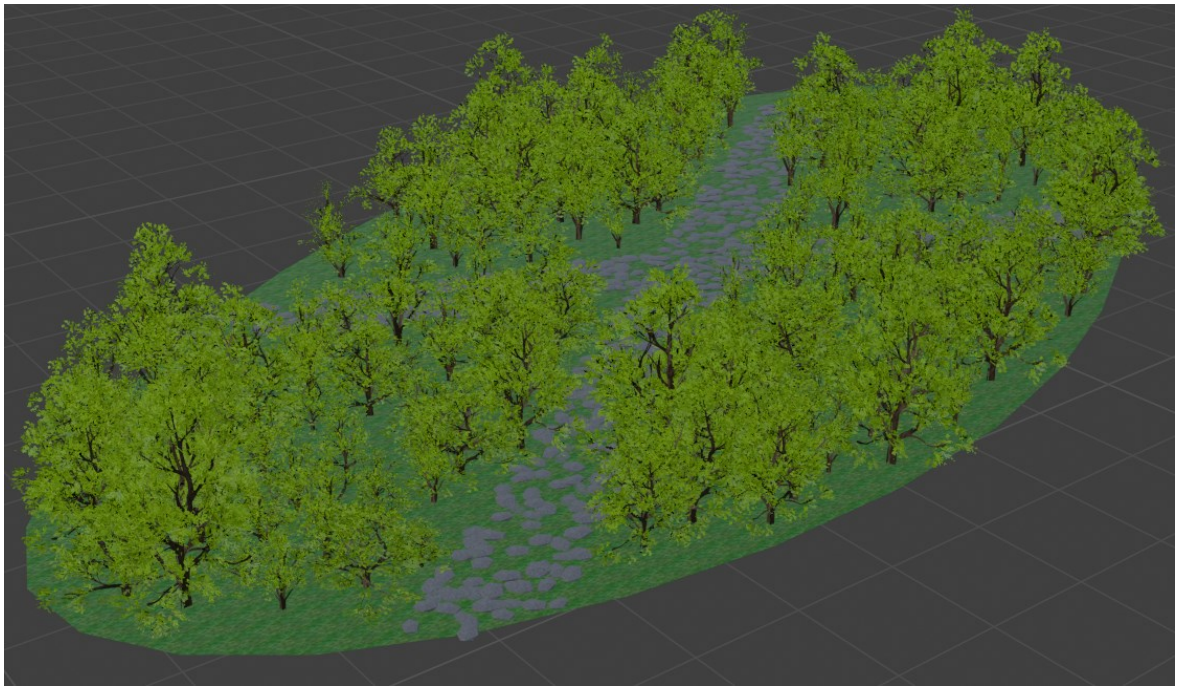


Obrázek 43 Zapojení Geometry Nodes kamenná cesta

Cesta je již funkční, ale mezi kameny jsou malé mezery, ve kterých se generují stromy. Tyto mezery jsou odstraněny tak, že výstup předešlého uzlu Delete Geometry je připojen do uzlu Realize Instances. Tento uzel vytvoří z připojených instancí (Instances) skutečná geometrická data. Důvodem použití instancí je možnost odkazovat se na více objektů, aniž by se skutečná data duplikovala, čímž se šetří výkon. V tomhle případě nyní je potřeba převést instance kamenů na skutečná data, protože bude použit uzel Raycast, do kterého nelze připojit instance. Výstup uzlu Realize Instances je připojen do uzlu Scale Elements, kde je ve vstupu Scale nastavena hodnota 2.5, díky které jsou všechny kameny zvětšeny násobkem 2.5. Zvětšené kameny ale nejsou viditelné, jsou totiž použity pouze jako vstup do uzlu Raycast. Při generování všech částí města byla již vytvořena a použita skupina uzlů s názvem Raycast. Stačí tedy vložit tuto skupinu a připojit výstup Scale Elements do vstupu. Aby se na tomto vytvořeném výběru negenerovaly stromy, musí být výstup skupiny Raycast připojen do vstupu Selection uzlu Instance on Points, který je zapojen v části generování stromů.



Obrázek 44 Zapojení Geometry Nodes k odstranění stromů z kamenných cest



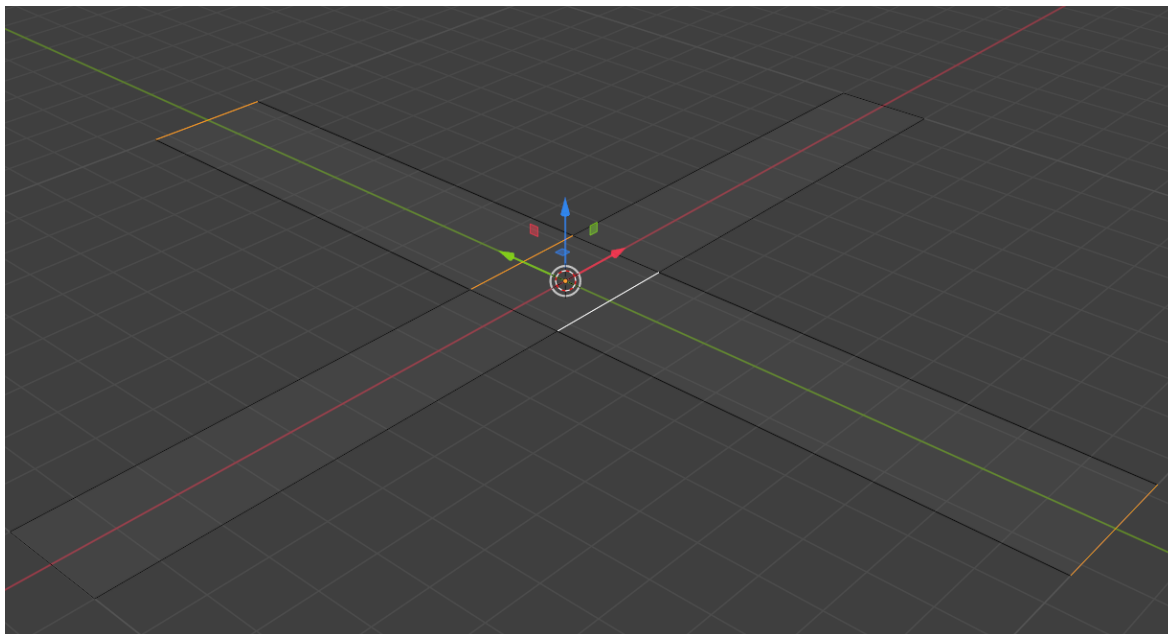
Obrázek 45 Ukázka vygenerovaného parku

4.6 Silnice

Silnice je vytvořena manuálně pomocí základních nástrojů a funkcí Blenderu, které jsou popsány v kapitole 1.3 Modelování. První je nový objekt Plane vložen pomocí klávesové zkratky Shift+A a název změněn na Road. Další úpravy objektu Road jsou provedeny v editačním režimu, na který je přepnut pomocí klávesy Tab. Pomocí klávesy 7 na numerické klávesnici je přepnuto na pohled z vrchu. Následně je vybrán režim výběru na Edge a jsou označeny vrchní a dolní hrany. Poté je stisknuta klávesa E, která aplikuje nástroj Extrude. Pomocí klávesy Y je označeno, že chceme použít nástroj pouze na ose Y. Ještě je stisknuta klávesa S, pomocí které se zapne nástroj Scale a je zadána hodnota 10. Tyto kroky jsou zopakovány i na postranních hranách a vytvoří se kříž. Objekt se z duplikuje a schová se klávesou H.

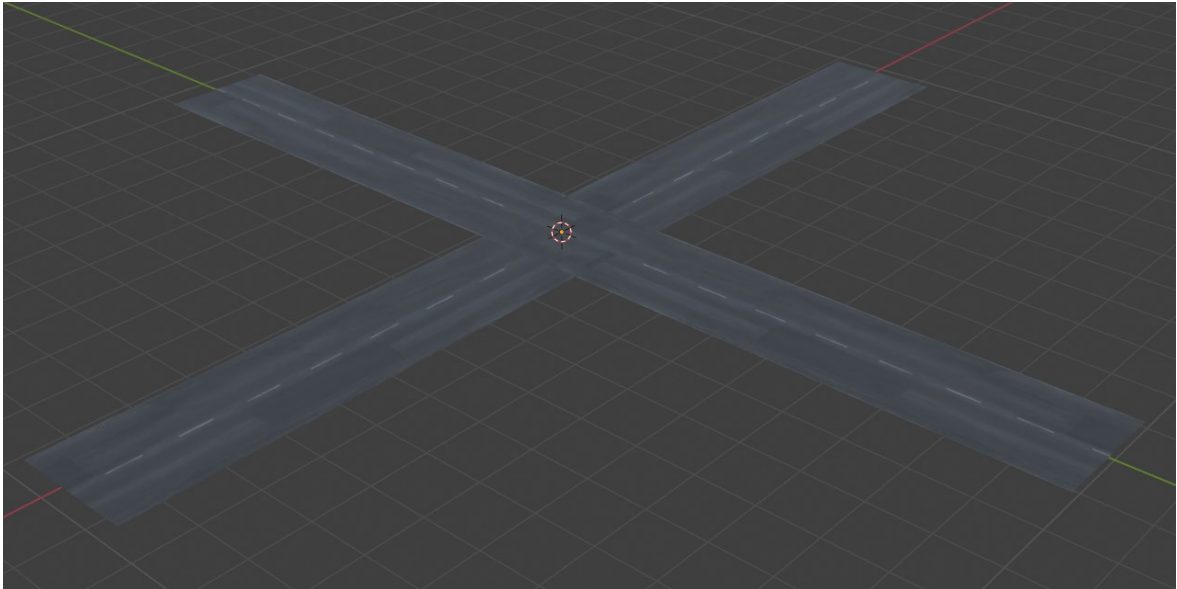
Nyní je aplikován nástroj Bevel klávesovou zkratkou Ctrl+Shift+B na čtyři prostřední vrcholy. Za pomoci pohybu nahoru kolečka na myši je nastaveno Bevel Segments na 5. Poté jsou odstraněny nově vytvořené plochy uvnitř silnice za použití funkce Dissolve Faces. Ta je aplikována kliknutím klávesy X a z nabídky je vybráno Dissolve Faces. Takhle připravený objekt bude dále použit při modelování chodníku.

Z duplikovaná část silnice, která byla schována na začátku se pomocí ikony oka vedle jména znovu zobrazí. Hrany ležící na ose Y se označí a pomocí klávesy S se zvětší o 1,3 na ose X a hrany ležící na ose X se také označí a zvětší se o 1,3 na ose Y. Tímto je docíleno zvětšení šířky silnice



Obrázek 46 Zvětšení šířky hrana na ose Y

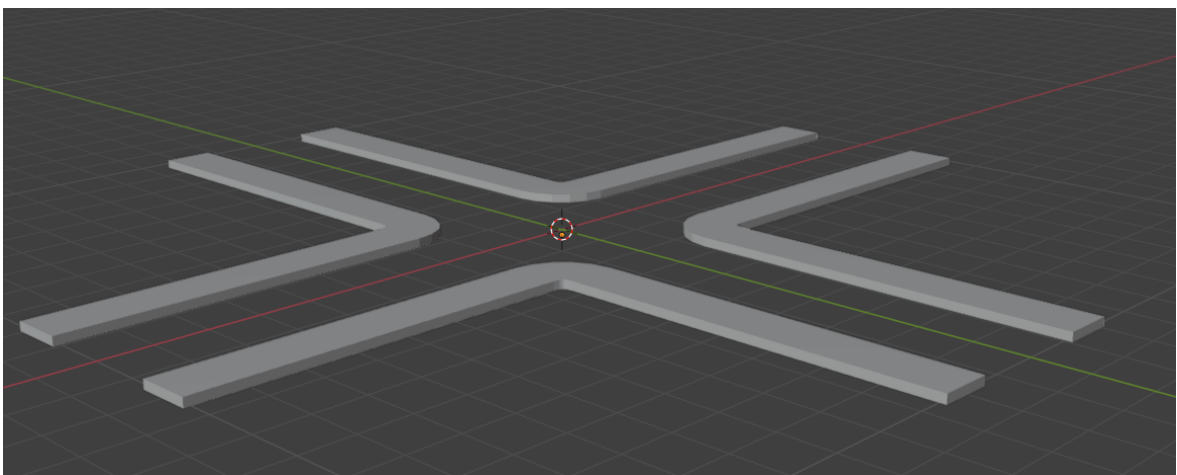
K aplikaci materiálu je vybrána sekce Material v panelu nástrojů, která se nachází úplně dole. Postranní plochy silnice jsou označeny a je aplikován materiál Roads pomocí tlačítka Assign. Prostřední ploše je přiřazen materiál CenterOfRoads. Materiál Roads je zdarma dostupný obrázek a je přiřazen jednotlivým plochám v pracovním prostoru UV Editing za pomoci UV Mappingu. Je použit UV Mapping, protože pomocí něj je jednoduše nanášena 2D textura na 3D model a to díky shodě mezi 2D souřadnicemi textury (U a V) a 3D povrchem modelu. Obrázek CenterOfRoads je totožný jako Roads až na to, že v něm byla odstraněna přerušovaná podélná čára. Nejprve v pracovním prostoru UV Editing v pravé části jsou označeny plochy silnice a je aplikován Unwrap, který lze vybrat v nabídce po kliknutí pravého tlačítka myši na objekt. V levé části jsou následně rozmístěny UV souřadnice ploch a upraveny tak, aby pasovaly na vytvořený model Road. Další části silnice jako chodník a přechod jsou po jejich vytvoření spojeny s objektem Road pomocí klávesové zkratky Ctrl+J.



Obrázek 47 Dokončený UV Mapping silnice

4.6.1 Chodník

Pro vytvoření chodníku se přepne metoda vykreslování na Wireframe klávesou Z. Teď se bude pracovat s objektem, který byl vytvořen ve druhém odstavci minulé kapitoly. V editačním režimu jsou označeny hrany, kde bude chodník a jsou z duplikovány klávesovou zkratkou Shift+D. Z duplikované hrany jsou odděleny pro lepší přehlednost klávesou P, kde je vybrán způsob oddělení Selection. Původní objekt se již nebude používat a může se tak smazat. Nástrojem Extrude je po ose Z vytvořená výška chodníku. Nyní je přidána šířka chodníku pomocí nástroje Extrude Faces Along Normal, který je vybrán z nabídky pomocí klávesové zkratky Alt+E. Šířka je nastavena pohybem myši.

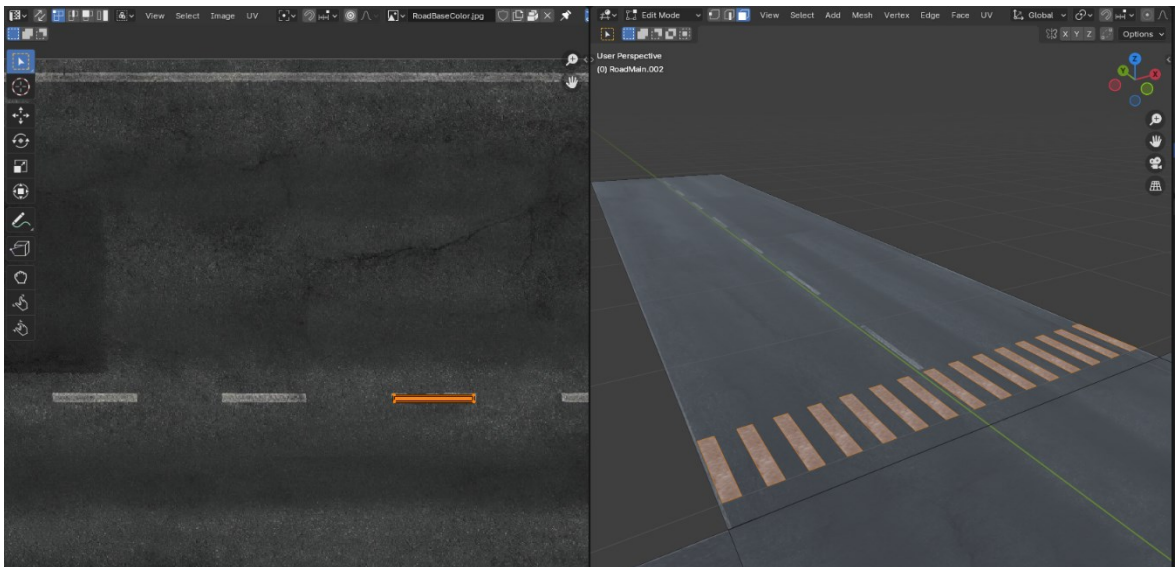


Obrázek 48 Dokončený model chodníku

Stejným způsobem jako u silnice je přiřazen materiál s názvem Sidewalk. Materiál Sidewalk je zdarma dostupný a stažený ze stránky blenderkit.com.

4.6.2 Přejechod

Přejechod bude vytvořen na jedné ze čtyř částí ploch silnice. Pomocí nástroje Loop Cut, který je aplikován klávesovou zkratkou Ctrl+R, se vytvoří plocha jednoho pruhu přechodu. Loop Cut se použije dvakrát. První na získání šířky celého přechodu a druhý pro plochu jednoho pruhu. Nyní nám stačí tento jeden pruh z duplikovat se stejnou délkou mezi pruhy až k okraji druhé strany chodníku. Před duplikováním ještě tomuto pruhu je přiřazen materiál Roads, jelikož je použita podélná přerušovaná čára jako barva pro přechod. Ta je aplikována v pracovním prostoru UV Editing s využitím UV Mappingu, který byl už použit a vysvětlen v kapitole 4.6 Silnice. V editačním režimu jsou všechny plochy přechodu označeny a v levé části jsou souřadnice přesunuty na jednu z částí přerušované čáry silnice.



Obrázek 49 Dokončený přechod s UV Mappingem

4.6.3 Semafory a lampy

Model semaforu je zdarma dostupný a stažen ze stránky cgtrader.com. Semafor za pomoci základních nástrojů byl upraven a taky z něj byl vytvořen model pouliční lampy. Pro semafory a lampy byly vytvořené jednoduché materiály MetalBlack, MetalYellow, Red, Yellow, Green a Lamp. Následně jednotlivým názvem odpovídajícím částem byly všechny tyto materiály přiřazeny. Semafory i lampy byly z duplikovány pomocí klávesové zkratky Shift+D a následně manuálně rozmístěny. Jelikož model silnice Road tvoří křižovatku se čtyřmi strany, tak na každé straně jsou rozmístěny 2 semafory a 6 pouličních lamp.

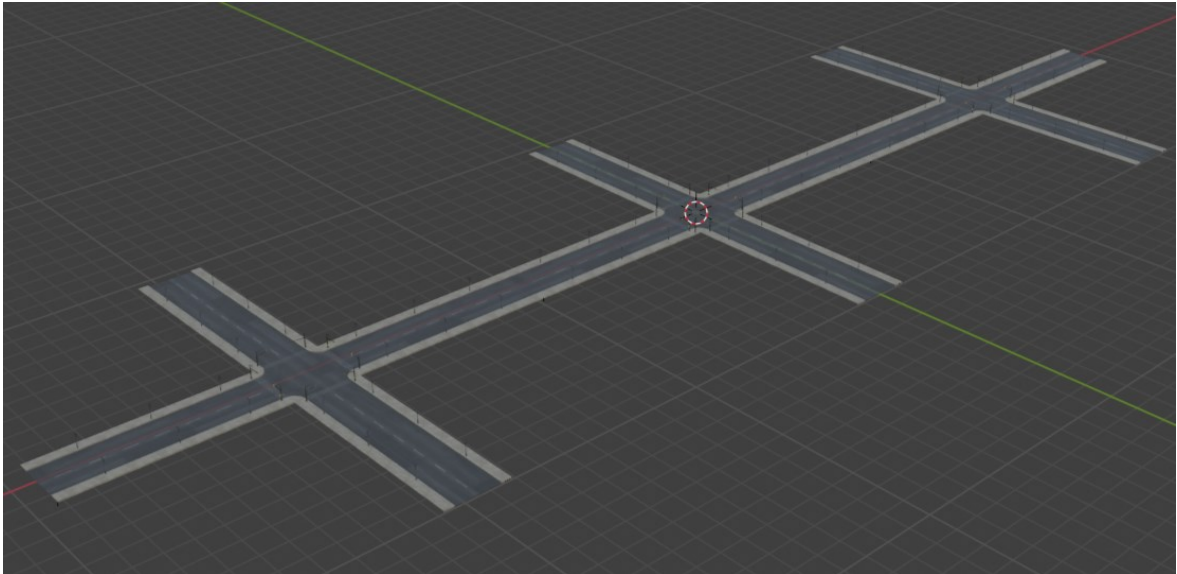


Obrázek 50 Dokončený model silnice

4.6.4 Duplikování silnic

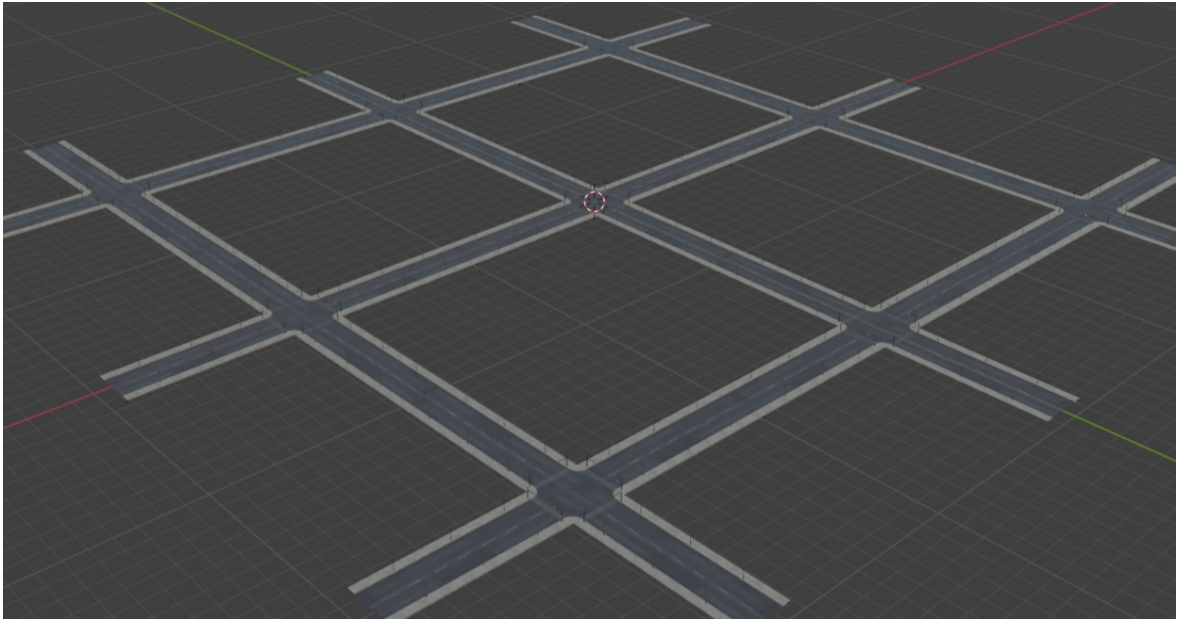
Dokončená silnice je vložena do City Generatoru pomocí uzlu Object Info. Uzel je nastaven jako Relative a As Instance, protože výstup je zapojen do uzlu Duplicate Elements, ve kterém je aktivováno duplikování Instance. Výstup Duplicate Index je připojen do prvního vstupu uzlu Math, kde je funkce nastavena na Multiply. Druhá hodnota je nastavena na 19.487. Výstup se připojí do vstupu X uzlu CombineXYZ, aby se silnice mohla duplikovat pouze na ose X. Pro posunutí z duplikované silnice je vložen nový uzel Set Position. Do tohoto uzlu je zapojen výstup Geometry z uzlu Duplicate Elements a výstup uzlu CombineXYZ do vstupu Offset. Pokud se hodnota Amount v uzlu Duplicate Elements změní na 2, tak se vytvoří kopie silnice posunutá na ose X o hodnotu 19.487. Pro duplikaci i na zápornou hodnotu osy X jsou předešlé zapojení uzlů

zkopírovány a pouze hodnoty se změň. Hodnota v novém uzlu Math s funkcí Multiply je změněna na -19.487. Kladný i záporný výstupy uzlu CombineXYZ se spojí novým uzlem Vector Math s operací Add. Výstup se spojí do vstupu Offset uzlu Set Position. Výstupy obou uzlů Duplicate Elements jsou spojeny do nového uzlu Join Geometry, který sloučí tyto výstupy, a poté je zapojen do vstupu Geometry uzlu Set Position. Tento krok se provedl, protože do vstupu Geometry lze zapojit pouze jeden výstup Geometry.

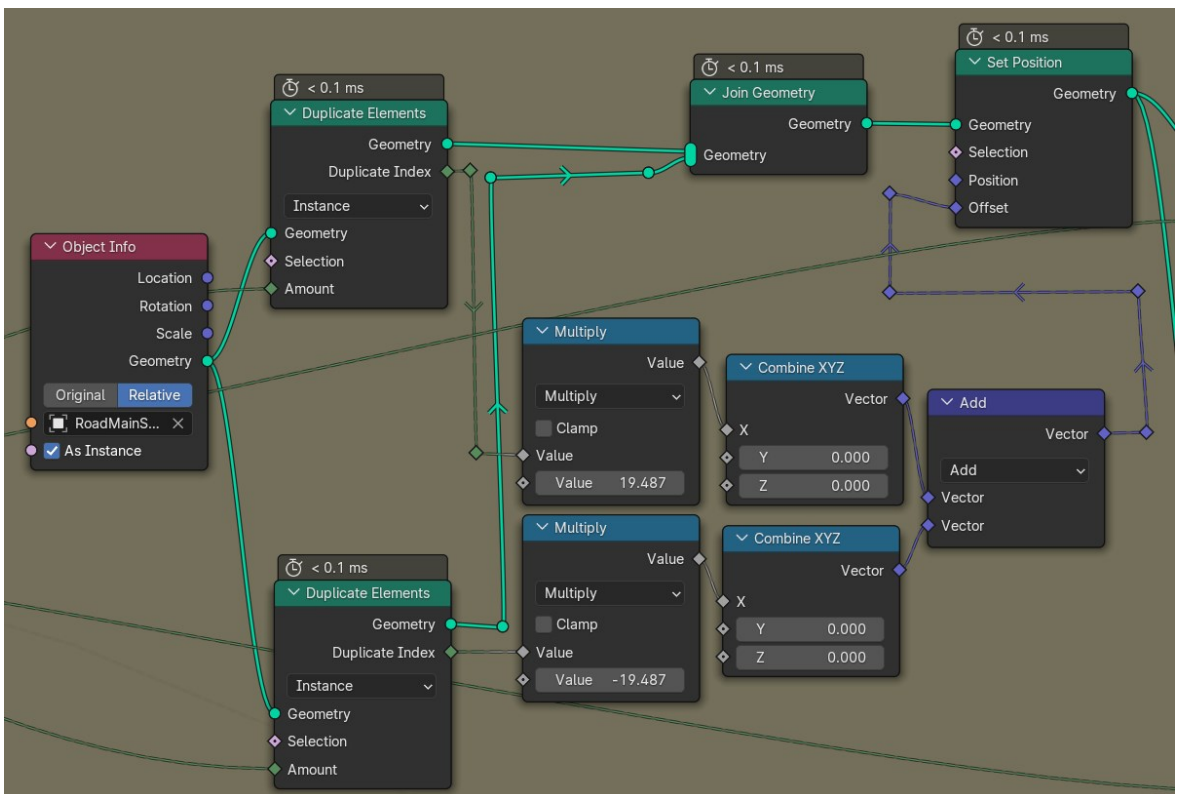


Obrázek 51 Duplikace silnic na ose X

Pro duplikaci silnic i na ose Y se použije stejné zapojení uzlů jako při duplikaci na ose X až na pár změn. Místo uzlu Object Info, který byl připojen jako vstup uzlů Duplicate Elements, jsou nové uzly Duplicate Elements propojeny s výstupem minulého uzlu Set Position. Při násobení hodnot Offsetu zůstanou hodnoty stejné, pouze se změň zapojení do uzlů CombineXYZ. Namísto zapojení do vstupu X jsou výstupy uzlu Multiply připojeny do vstupu Y, čímž jsou cesty duplikovány na ose Y.



Obrázek 53 Duplikace silnic do všech stran



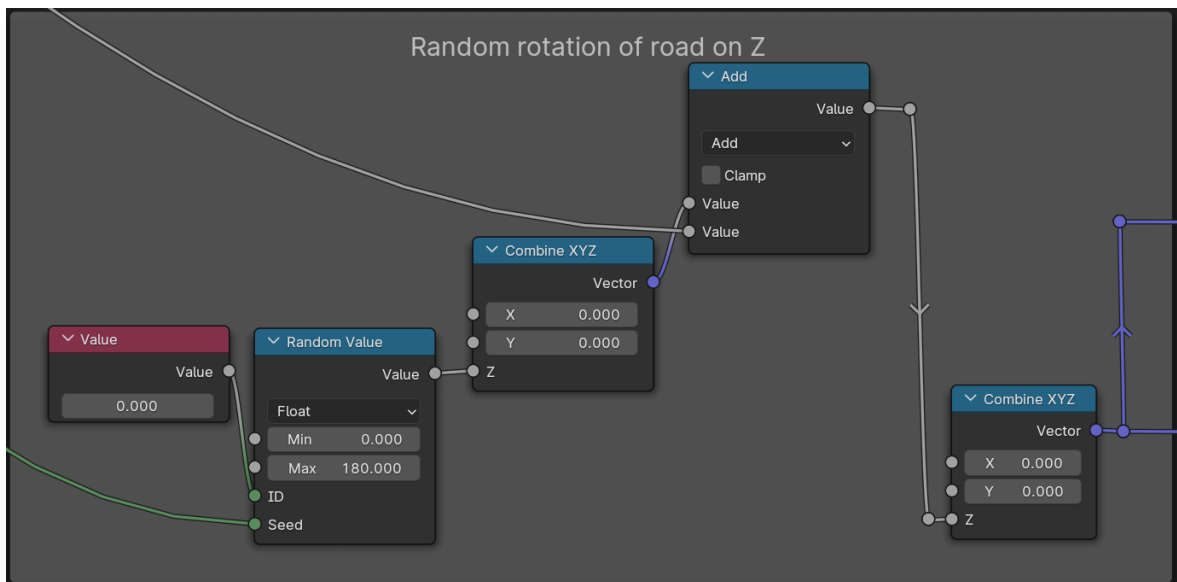
Obrázek 52 Zapojení Geometry Nodes pro duplikaci na ose X

Nyní se přidá nový vstup uzlu Group Input s názvem AmmountOfGridRoads, který je spojen se všemi vstupy Ammount uzlů Duplicate Elements, aby si uživatel mohl zadat počet duplikací.

4.6.5 Rotace silnice

Silnice se momentálně generují pokaždé stejně. Pro přidání vizuálního prvku procedurálního generování bude silnici přidána náhodná rotace a zároveň bude poskytnut uživateli nový vstup, pomocí kterého si může nastavit vlastní rotaci.

Výstup posledního uzlu Set Position se připojí do uzlu Transform Geometry. Přidá se nový vstup uzlu Group Input s názvem RotationOfGridRoads, který je připojen do vstupu Z uzlu CombineXYZ, aby se rotace aplikovala pouze na ose Z. Následně je výstup připojen do vstupu Rotation uzlu Transform Geometry. Nyní si uživatel může nastavit vlastní rotaci, ale bude přidána ještě náhodná rotace. Náhodnou rotaci přidá kombinace uzlů Random Value, CombineXYZ a Value. V uzlu Random Value je typ přepnut na Float a v sekci Min se hodnota nastaví na 0 a hodnota Max na 180. Výstup se připojí do vstupu Z uzlu CombineXYZ. Jelikož do vstupu Rotation lze zapojit pouze jeden výstup, tak obě rotace jsou spojeny. Spojení se provede pomocí uzlu Math s funkcí Add, kdy do prvního vstupu se zapojí vstup RotationOfGridRoads a do druhého výstup posledně přidaného uzlu CombineXYZ. Přidá se nový vstup uzlu Group Input s názvem Seed a ten je připojen do vstupu Seed uzlu Random Value. Posledním krokem je zapojení uzlu Value s hodnotou 0 do vstupu ID, bez kterého náhodná rotace nefunguje.

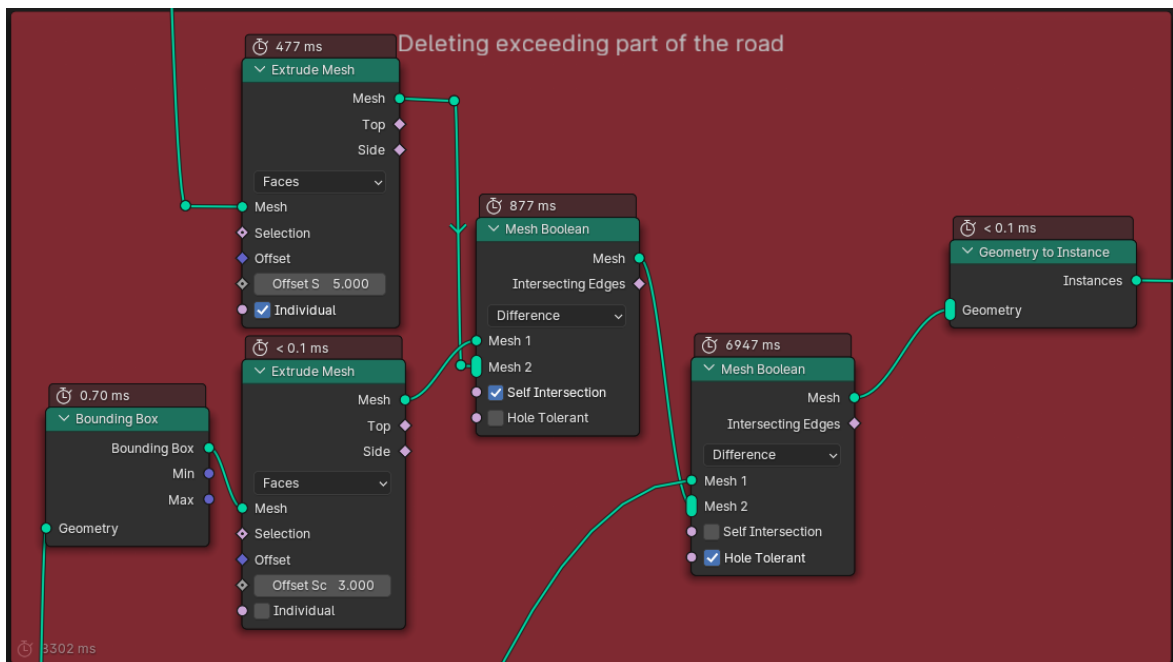


Obrázek 54 Zapojení Geometry Nodes pro náhodnou rotaci silnic na ose Z

4.6.6 Přesahující část

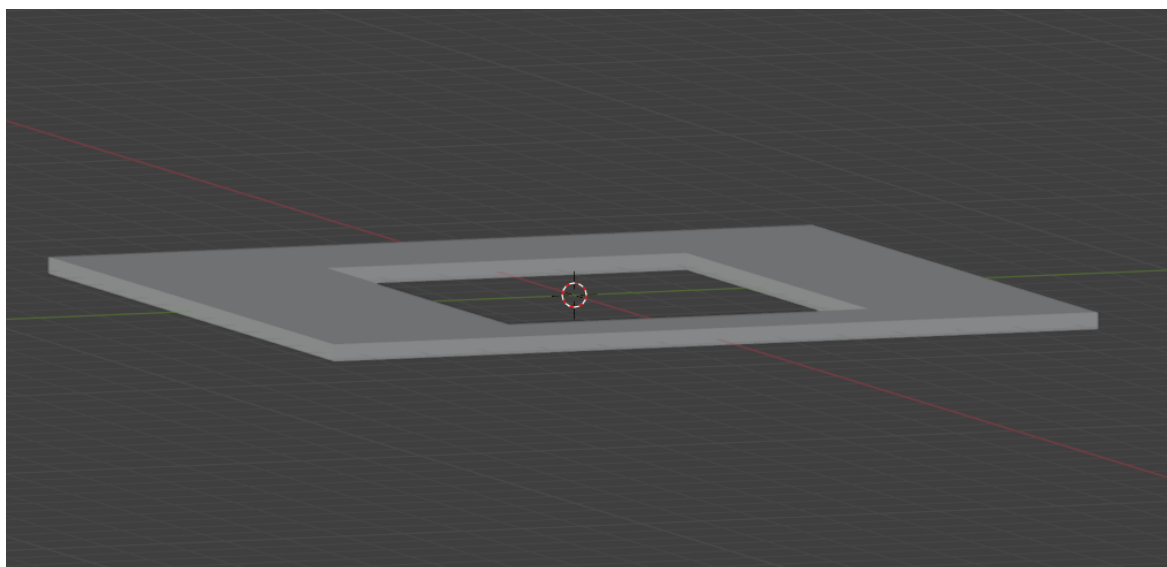
Pokud je zadán velký počet duplikací, aby se cesty vygenerovaly až k okraji ohraničení města, které bylo vytvořeno v kapitole 4.4.1 Ohraničení města, tak některé části silnice budou přesahovat tohle ohraničení. K odstranění těchto přesahujících částí silnic bude použit uzel Mesh Boolean. Tento uzel je schopen řezat, odečítat a spojovat části Geometry vstupů objektů podle jejich zapojení. Obsahuje tedy 2 vstupy Mesh 1 a Mesh 2. Mesh Boolean bude použit pro vytvoření objektu, který ohraničí přesahující části a následně vše uvnitř smaže.

První je výstup uzlu Bounding Box z části ohraničení města zapojen do vstupu Extrude Mesh. Pomocí tohoto uzlu bude rovině přidána tloušťka na ose Z. Extrude Mesh se přepne na Faces a hodnota Offset Scale je nastavena na 5. Poté by se zapojil výstup z uzlu Transform Geometry, který byl přidán v kapitole 4.6.5 do uzlu Bounding Box. Jelikož ale později budou části silnic jako jednotlivé hrany, body a plochy přesahující plochu smazány, tak se musí mezi spojení přidat uzel Realize Instances. Následně výstup Bounding Box je zapojen do Extrude Mesh, který je opět přepnut na Faces, ale hodnota Offset Scale musí být nastavena na menší než v předešlé části, proto bude nastavena na 3. Menší musí být, jelikož část pokrývající město bude odstraněna.



Obrázek 55 Zapojení Geometry Nodes odstraňující přesahující část silnic

Těmito kroky se tedy vytvořily dva nové objekty. První objekt pokrývá celou část města. Druhý objekt pokrývá momentálně část města, ale zároveň i přesahující část silnic. Za použití uzlu Mesh Boolean bude teď druhý objekt, který pokrývá pouze část města odstraněn od objektu pokrývající obě části a zůstane tak pouze přesahující část. Mesh Boolean je nastaven jako Difference a musí být zaškrtnuté pole Self Intersection, protože objekty se protínají. Pokud by nebylo zaškrtnuto výpočet by nebyl správný. Do vstupu Mesh 1 je zapojen výstup Extrude Mesh prvního objektu, který pokrývá jak část města, tak i přesahujících cest. Do vstupu Mesh 2 je zapojen výstup Extrude Mesh druhého objektu. Tím je získán objekt, pomocí kterého budou smazány části silnice přesahující plochu města.



Obrázek 56 Vytvořený objekt prvního uzlu Mesh Boolean

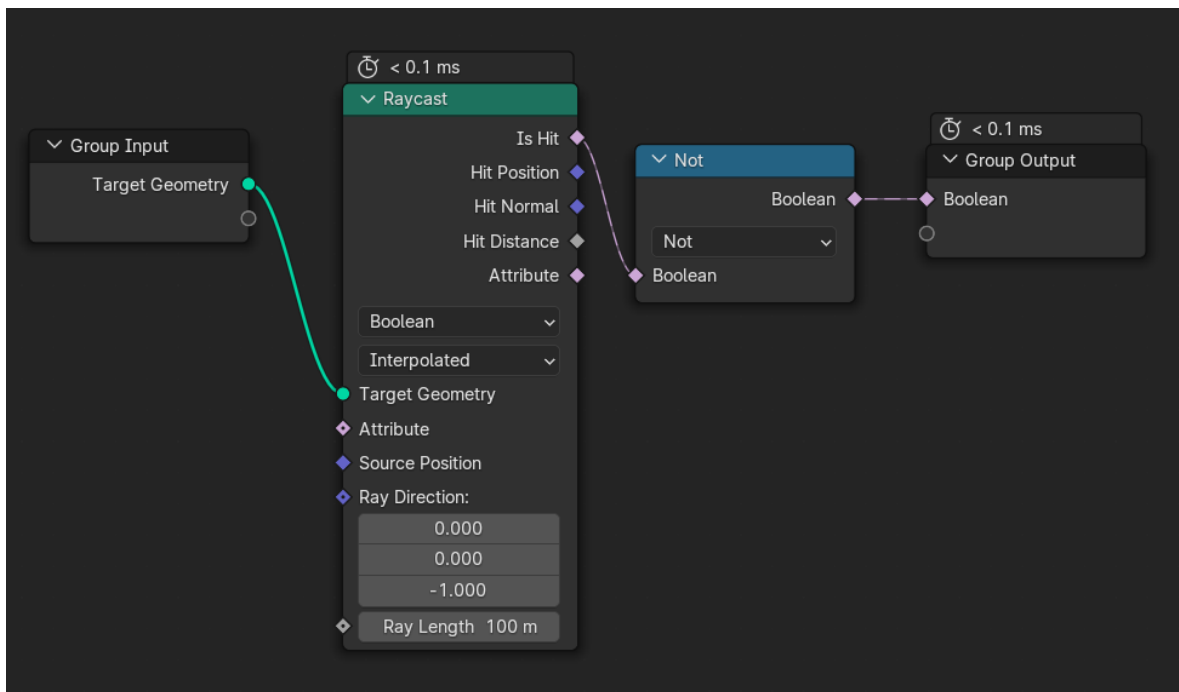
Ke smazání přesahující části silnice bude použit znovu uzel Mesh Boolean. Pouze se pole Self Intersection odškrtně a zaškrtně se Hole Tolerant. Hole Tolerant zlepšuje výstup z Boolean pro geometrii Non-manifold objektů, ale také vyžaduje více výpočetního času. Doporučuje se použít tuhle funkci pouze pokud je zapojen Non-manifold Mesh objekt. Jelikož součástí silnic jsou plochy, které nemají žádnou tloušťku, tak spadají právě pod Non-manifold a musí být zaškrtnuto Hole Tolerant. Do vstupu Mesh 1 se připojí výstup použitého uzlu Mesh Boolean a do vstupu Mesh 2 se zapojí výstup uzlu Realize Instances, který byl již přidán mezi spojení Transform Geometry a Bounding Box. Celé tohle zapojení je ukázáno na obrázku 55.

Nyní je odstranění přesahujících částí silnic funkční, ale zmizely textury nastavené pomocí UV Mappingu. Ty byly aplikovány, když silnice byly Instances, proto se převedou zpět

z geometrie na Instances díky uzlu Geometry to Instance. Výstup uzlu Mesh Boolean se zapojí do Geometry to Instance a textury jsou znovu funkční.

4.7 Raycast

Veškeré generování, co bylo vytvořené se teď protíná, jelikož pro generaci byl použit uzel Distribute Points on Faces. Tento uzel, jak již bylo zmíněno, rozmístí po připojení vstupním objektu body a tyto body poté obsadí objekty, které jsou připojené ve vstupu Instance uzlu Instance on Points. Proto pro odstranění rozmístěných bodů, které přesahují svou plochu a tím zasahují tam, kde by neměly bude použit uzel Raycast. Uzel Raycast protíná paprsky z jednoho objektu zdrojové geometrie s jiným cílovým Mesh objektem. Tyto paprsky jsou v podstatě imaginární čáry, které vycházejí z bodu na zdrojové geometrii v určeném směru. Pokud se tyto paprsky protínají s cílovým Mesh objektem, tak uzel vypočítá různé vlastnosti těchto průsečíků. Raycast tedy poskytuje cenné údaje pro různé účely, jako je například vykreslování, detekce kolizí nebo procedurální generování obsahu. V tomhle případě bude použit právě k detekci kolizí a následně k jejich odstranění.



Obrázek 57 Zapojení Geometry Nodes skupiny uzlů Raycast

Díky tomu, že všechny generované části jsou za pomoci uzlů Distribute Points on Faces a Instance on Points, tak bude vytvořena jedna skupina uzlů. Tahle skupina se použije všude pouze bude změněno připojení vstupu Geometry. Hlavním uzlem této skupiny bude Raycast. Po vložení se nemusí měnit žádné hodnoty, zůstane v základním nastavení. Výstup Is Hit se zapojí do nového uzlu Boolean Math. Tento uzel provádí základní logice operace se vstupy, jako například And, Or nebo Not. Bude vybrána operace Not, která funguje tak, že vstupní hodnotu zneguje. Zjednodušeně řečeno invertuje zapojený vstup. Toho bude využito tak, že do vstupu Target Geometry se zapojí objekt, pokud se v tomto objektu budou nacházet body, které jsou vygenerované na jiném objektu, tak tyto body nebudou použity a budou jakoby smazány.

4.7.1 Silnice

Nově vytvořená skupina uzlů Raycast bude pro silnice použita u všech částí města, jelikož cesty zasahují až k okraji města. První se musí zapojit výstup uzlu Transform Geometry, do kterého byl v kapitole 4.6.4 Duplikování silnic zapojen výstup z uzlu Set Position. Aby uzel Raycast mohl správně fungovat, musíme převést výstup Transform Geometry z Instances na Geometry. To je provedeno díky uzlu Realize Instances. Nyní se zapojí výstup Realize Instances do vstupu skupiny uzlů Raycast. Tímto spojením je nachystán výstup, který bude použit v dalších částech.

4.7.2 Parky

Další součástí generátoru města, která se nachází ve všech částech města, jsou parky. Při generování parků byl použit dvakrát stejný uzel Subdivision Surface. Bude použit výstup druhého uzlu Subdivision Surface, pomocí kterého se odstranily ostré hrany v kapitole 4.5 Parky. Výstup Mesh je zapojen do vstupu znovu přidané skupiny uzlů Raycast. V dalších částech budou tyto dva Raycasty použity a propojeny.

4.7.3 Centrum

Pro část centrum města budou použity skupiny uzlů Raycast jak silnic, tak i parků. K aplikování obou skupin se nejdříve musí spojit, aby byly použity obě zároveň. Spojení se provede pomocí uzlu Math. Funkce uzlu se přepne na Multiply, kde do prvního vstupu Value se připojí výstup Raycast parků a do druhého vstupu Value se připojí výstup Raycast silnic. Po změně funkce uzlu Math se uzel bude jmenovat stejně jako vybraná funkce. Budou se vybírat body, které jsou vygenerovány na ploše centra města zasahující do ploch

parků a silnic. Proto se zapojí výstup uzlu Multiply do vstupu Selection uzlu Instance on Points, který je součástí generující centrum města.

4.7.4 Předměstí

Díky tomu, že veškeré čtyři části města jsou stejného tvaru a pouze se zvětšují díky uzlu Scale Elements, tak bude pokaždé zapojen předešlý uzel Multiply. Tím pádem uzel Multiply z předešlé kapitoly se zapojí do nově vytvořeného uzlu Math, kde se vybere opět funkce Multiply. Teď je jeden volný vstup, do kterého bude zapojena nově přidaná skupina uzlů Raycast. Do skupiny Raycast musí být zapojen výstup prvního uzlu Scale Elements patřící části centrum města. Teď se zapojí výstup Multiply do vstupu Selection uzlu Instance on Points, ale do již druhého, a to patřícího části předměstí města. Nyní kdyby se zvětšovala část centrum města pomocí vstupu Scale of Center, tak při každém zvětšení zasahujícího do plochy předměstí města, by se vygenerované budovy předměstí smazaly. Naopak pokud by se část centrum města zmenšovala, tak při každém zmenšení plochy centru města, by byla dostupná větší plocha předměstí, na které se budou generovat budovy.

4.7.5 Vzdálené předměstí

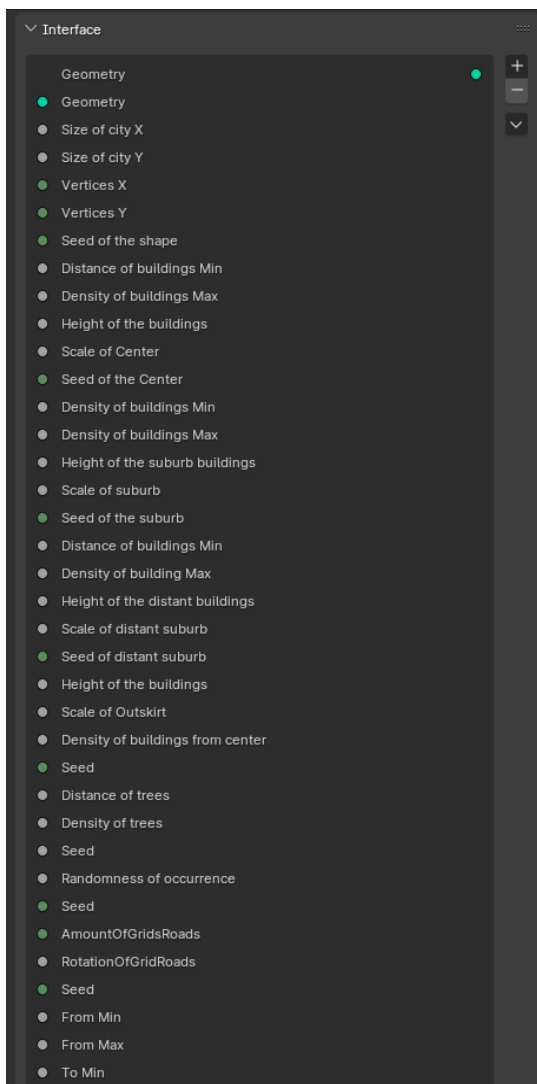
Jak již bylo vysvětleno v předešlé kapitole, tak se přidá pouze jedna nová skupina uzlů Raycast, do které bude zapojen výstup druhého uzlu Scale Elements, který zvětšuje část předměstí. Následně se výstup předešlého uzlu Multiply zapojí do nového stejného uzlu. Do volného vstupu se zapojí výstup skupiny uzlů Raycast, který byl vytvořen v této kapitole. Poté zbývá zapojit výstup Multiply do vstupu Selection uzlu Instance on Points patřícího části vzdáleného předměstí.

4.7.6 Okolí

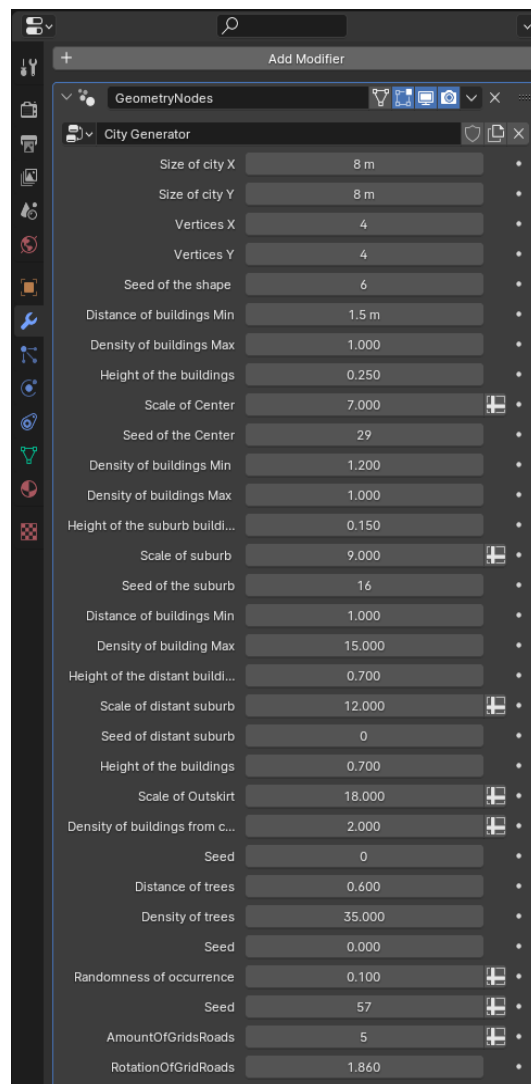
V téhle části města už nebude přidána nová skupina uzlů Raycast, ale stále je nutné ji přidat předešlé části, a to vzdáleného předměstí. Pomocí stejných kroků jako v minulé kapitole, se vytvořil nový uzel Multiply, ve kterém je zapojen předešlý výstup Multiply a výstup skupiny uzlů Raycast, ve kterém do vstupu byl zapojen výstup třetího uzlu Scale Elements.

4.8 Uživatelské rozhraní

V každé kapitole praktické části generátoru města byly vytvořeny nové vstupy uzlu Group Input a následně propojeny příslušným vstupům uzlů, díky kterým bude uživatel schopen měnit a upravovat parametry města. Veškeré vstupy, jejich propojení a názvy jsou již vytvořené v předešlých kapitolách, ale nejsou moc přehledné.



Obrázek 58 Přidané vstupy v praktické části uzlu Group Input



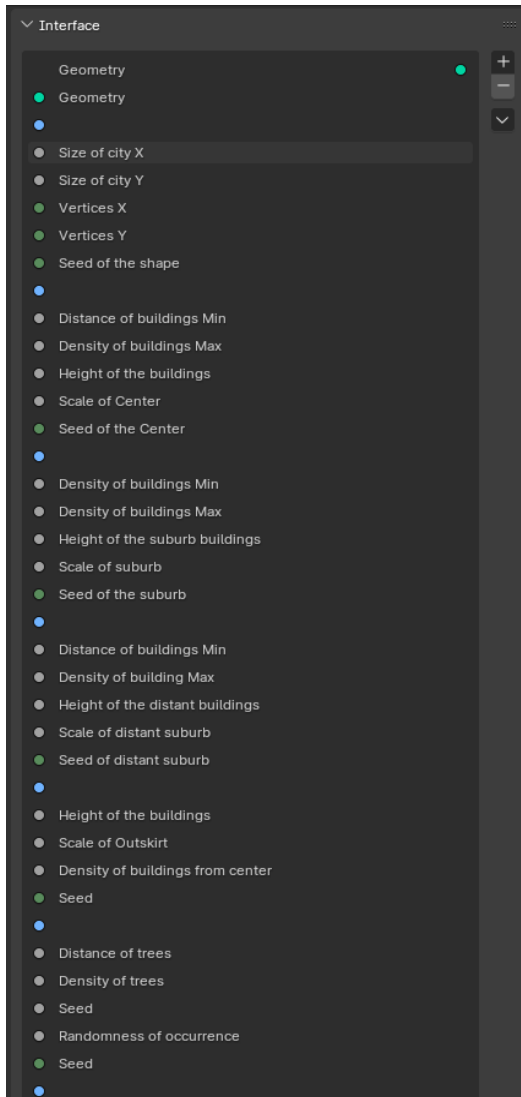
Obrázek 59 Zobrazení přidaných vstupů uzlu Group Input v panelů vlastností

Proto aby byly veškeré přidané vstupy přehlednější, tak se přidají prázdné nové vstupy typu String. Budou vždy vloženy před první vstup, který byl vytvořen v příslušné kapitole. Například pro úpravu parametrů náhodně vygenerovaného tvaru plochy města byly přidány čtyři vstupy, a to Size of city X, Size of city Y, Vertices X a Vertices Y. Tím pádem první nový prázdný vstup typu String je vložen před vstup Size of city X. Druhý prázdný vstup

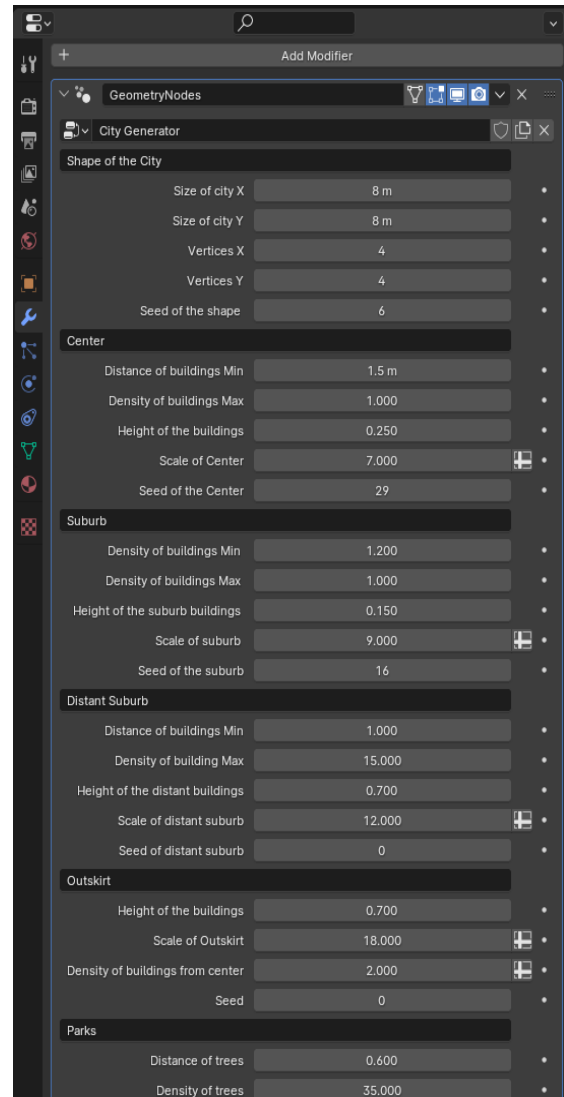
bude umístěn pod poslední vstup kapitoly Plocha města a tím je Vertices Y. Stejným způsobem budou vytvořeny i další prázdné vstupy String až po poslední kapitolu praktické části.

Nyní pro ještě větší přehlednost v uživatelském rozhraní se vloží text jednotlivých částí. V panelu vlastností v sekci modifikátory jsou prázdné černé vstupní pole. Do těchto vstupů bude vložen text odpovídající částem, které tyto parametry budou ovlivňovat. Takže pro příklad, který byl popsán v minulém odstavci, bude do prvního vstupu vepsán text Shape of the City. Do druhého prázdného vstupu se napíše Center. Opět stejným způsobem se přejmenují i všechny ostatní prázdné vstupy, aby odpovídaly jednotlivým kapitolám v praktické části.

Uživatelské rozhraní je tedy realizováno pomocí uzlu Group Input. Díky přidaným vstupům uzlu Group Input má uživatel možnost snadno ovládat a upravovat parametry generovaných částí města. Veškeré vstupy jsou pojmenovány a přehledně popsány podle specifických parametrů, které ovlivňují vzhled jednotlivých částí města. Uživatel může měnit parametry buď pohybem myši, nebo si sám přepíše určitou hodnotu parametru, kterou chce změnit. Změny se provedou okamžitě při změně hodnoty parametru. Díky tomu je zajištěna snadná a intuitivní manipulace s generátorem města a možnost rychlého přizpůsobení výsledného obsahu podle individuálních potřeb a představ uživatele



Obrázek 60 Nově přidané prázdné vstupy String uzlu Group Input



Obrázek 61 Přejmenované prázdné vstupy uzlu Group Input v panelu vlastností

ZÁVĚR

Cílem této bakalářské práce bylo navrhnout a implementovat funkční modul v programu Blender umožňující procedurální generování 3D modelů měst dle zadaných parametrů uživatelem. K vytvoření modulu bylo nutné se prvně seznámit s prostředím Blenderu.

V teoretické části práce bylo tedy vysvětleno prostředí programu Blender. Byly popsány základní rozhraní, techniky modelování a nástroje, které Blender poskytuje. Dále se v této části popsala existující řešení generování měst v Blenderu. Další část se už přímo věnovala nástroji Geometry Nodes. Na začátku bylo vysvětleno, co je to procedurální generování, jak funguje a kde všude je možné ho použít. Dále byly uvedeny přímo odvětví a příklady, kde se nejvíce procedurální generování používá a proč ho použít nebo nepoužít. Dále se už tato část podrobněji věnovala nástroji Geometry Nodes, pomocí kterého byl modul zrealizován. Byly popsány základní principy Geometry Nodes, jeho funkčnosti, možnosti a jak lze využít jednotlivé uzly. Zmíněny a obecně popsány byly veškeré typy uzlů, které jsou dostupné v Blenderu verzi 4.0. Vysvětlení a pochopení těchto popsanych nástrojů a funkcí poskytla důležitý základ pro praktickou aplikaci v následující části.

Praktická část práce se pak zabývala konkrétní implementací procedurálního generování 3D modelů měst pomocí nástroj Geometry Nodes. Popisovala jednotlivé kroky, jak se vytvořila náhodně generovaná plocha, která byla důležité pro všechny čtyři části města. Popsáno bylo použití všech uzlů, jak byly mezi sebou propojeny a proč. Dále zmiňované čtyři generované části měst byly stejně tak do hloubky popsány. Součástí měst také bylo generování parků a silnic. V parcích se dále generovaly stromy a kamenná cesta. Poslední větší částí byla kapitola o uzlu Raycast. Pomocí tohoto uzlu byly vyřešeny veškeré kolize od vygenerovaných budov až po kamennou cestu. Jako poslední se tahle část věnovala uživatelskému rozhraní, které umožňuje uživatelům snadno měnit parametry generovaného města a přizpůsobit ho jejich potřebám a preferencím.

Výsledkem práce je modul, vytvořený za pomoci seskupení Geometry Nodes, který umožňuje uživatelům rychle a snadno procedurálně generovat rozsáhlé 3D modely měst. Uživatelé mají možnost upravovat tyto modely a přizpůsobovat je svým preferencím pomocí uživatelského rozhraní, které bylo taktéž řešeno pomocí Geometry Nodes.

Implementace tohoto modulu byla časově náročná, zejména kvůli složitějším výpočtům a náročnosti programu, které jsou součástí generování. Limitace výkonu počítače, na kterém byl tenhle modul vytvořen výrazně omezila efektivitu práce, což se projevilo i na

dokončeném funkčním modulu. Z tohoto důvodu by části, jako například generování silnic, mohly být řešeny efektivněji a více procedurálně. Avšak, pokud je uživatel pokročilejší v Blenderu s nástrojem Geometry Nodes, má možnost vložit do tohoto modulu jiné řešení generování silnic.

SEZNAM POUŽITÉ LITERATURY

- [1] *Introduction*. Online. Blender 4.0 Manual. 2024, Last updated on 01/15/2024. Dostupné z: https://docs.blender.org/manual/en/latest/getting_started/about/introduction.html. [cit. 2024-03-14].
- [2] CHILLINGWORTH, Alec. *The pros & cons of creating 3D content with Blender software*. Online. In: Epidemic Sound. Mar 30, 2023. Dostupné z: <https://www.epidemicsound.com/blog/blender-software/>. [cit. 2024-03-14].
- [3] WOOD, Larry. *10 Reasons to Use Blender*. Online. All3DP. Published Sep 27, 2022. Dostupné z: <https://all3dp.com/2/reasons-to-use-blender-advantages/>. [cit. 2024-03-14].
- [4] *Blender's History*. Online. Blender Manual. 2024, Last updated on 01/15/2024. Dostupné z: https://docs.blender.org/manual/en/latest/getting_started/about/history.html. [cit. 2024-03-14].
- [5] *Window System Introduction*. Online. Blender Manual. 2024, Last updated on 01/15/2024. Dostupné z: https://docs.blender.org/manual/en/latest/interface/window_system/introduction.html. [cit. 2024-03-17].
- [6] M. RODAS, Lorena. *How To Use Properties Panel in Blender*. Online. In: Skillademia. 2022, Updated on: August 24, 2022. Dostupné z: <https://www.skillademia.com/3d/blender/how-to-use-properties-panel-in-blender/>. [cit. 2024-03-17].

- [7] M. RODAS, Lorena. *How To Use Animation Timeline in Blender*. Online. In: Skillademia. 2022, Updated on: August 24, 2022. Dostupné z: <https://www.skillademia.com/3d/blender/how-to-use-animation-timeline-in-blender/>. [cit. 2024-03-17].
- [8] *Workspaces*. Online. Blender Manual. 2024, Last updated on 01/15/2024. Dostupné z: https://docs.blender.org/manual/en/latest/interface/window_system/workspaces.html. [cit. 2024-03-17].
- [9] STOCKING, Brandon. *Using Workspaces in Blender 3D*. Online. Brandon's Drawings. 2023, Updated on June 4, 2023. Dostupné z: <https://brandonsdrawings.com/how-to-use-workspaces-in-blender-3d/>. [cit. 2024-03-17].
- [10] *3D Modeling in Blender: What Beginners Need To Know*. Online. Upwork. 2023. Dostupné z: <https://www.upwork.com/resources/3d-modeling-in-blender>. [cit. 2024-03-17].
- [11] *Edit Mode Basics – Blender Knowledgebase*. Online. Katbits. 2024. Dostupné z: <https://www.katsbits.com/codex/edit-mode-basics/>. [cit. 2024-03-17].
- [12] SULLINS, Samuel. *Blender Basics Part 2: Edit Mode*. Online. In: Medium. 2022. Dostupné z: <https://medium.com/@samuelsullins/blender-basics-part-2-edit-mode-706770da9810#:~:text=Open%20Blender%20again%2C%20and%20select,%3A%20Vertex%2C%20Edge%20and%20Face>. [cit. 2024-03-17].
- [13] *Modeling in Blender for budding 3D Artists*. Online. Garagefarm. 2023. Dostupné z: <https://garagefarm.net/blog/modeling-in-blender-for-budding-3d-artists>. [cit. 2024-03-17].

- [14] *Structure*. Online. Blender Manual. 2024, Last updated on 01/15/2024. Dostupné z: <https://docs.blender.org/manual/en/latest/modeling/meshes/structure.html>. [cit. 2024-03-17].
- [15] SELIN, Erik. *Blender curve object: Bezier, Nurbs, paths, Modifiers and profiles*. Online. In: Artisticrender. 2021, Last update: June 12, 2021. Dostupné z: <https://artisticrender.com/blender-curve-object-bezier-nurbs-paths-modifiers-and-profiles/>. [cit. 2024-03-17].
- [16] *API Overview*. Online. Blender Manual. 2024. Dostupné z: https://docs.blender.org/api/current/info_overview.html. [cit. 2024-03-17].
- [17] *Documentation*. Online. Github. 2021. Dostupné z: <https://github.com/vvoovv/blosm/wiki/Documentation>. [cit. 2024-03-17].
- [18] *Google Maps in Blender - BLOSM Addon* [@Markom3D]. Online. Dostupné z: https://www.youtube.com/watch?v=PCLHosNWojM&t=74s&ab_channel=Markom3D. [cit. 2024-03-17].
- [19] LUCKE, Jacques. *Object Scatter*. Online. Blender Addons. 2024. Dostupné z: <https://blender-addons.org/object-scatter-addon/>. [cit. 2024-03-17].
- [20] COUTURIER, Arnaud. *SceneCity: 3D city generator addon for Blender 3.4+*. Online. Cgchan. 2022. Dostupné z: <https://www.cgchan.com>. [cit. 2024-03-18].
- [21] COUTURIER, Arnaud. *SceneCity 1.9 doc - Fundamentals*. Online. Cgchan. 2022. Dostupné z: <https://scenecitydoc.cgchan.com/fundamentals>. [cit. 2024-03-18].

- [22] SELIN, Erik. *Blender geometry nodes fundamentals guide*. Online. In: Artisticrender. 2022, Last update: July 23, 2022. Dostupné z: <https://artisticrender.com/blender-geometry-nodes-fundamentals-guide/>. [cit. 2024-03-17].
- [23] ALI, Aftab. *Blender: Geometry Nodes – Simply Explained*. Online. All3DP. 2022. Dostupné z: <https://all3dp.com/2/blender-geometry-nodes-simply-explained/>. [cit. 2024-03-17].
- [24] *Blender Geometry Nodes – Tutorial for Beginners*. Online. Polyfable. 2023, Last Updated on December 13, 2023. Dostupné z: <https://polyfable.com/tutorials/blender-geometry-nodes-tutorial-for-beginners/>. [cit. 2024-03-18].
- [25] *Procedural Generation*. Online. Saturncloud. 2024. Dostupné z: <https://saturncloud.io/glossary/procedural-generation/>. [cit. 2024-03-18].
- [26] VAN BRUMMELEN, Jessica a CHEN, Bryan. *Procedural Generation*. Online. In: Mit. 2019. Dostupné z: https://www.mit.edu/~jessicav/6.S198/Blog_Post/ProceduralGeneration.html#:~:text=What%20is%20Procedural%20Generation%3F,or%20non-player%20character%20dialogue. [cit. 2024-03-18].
- [27] IKEDA, Aoi. *Unraveling the Mysteries of Procedural Generation in Gaming*. Online. In: Tokengamer. 2023. Dostupné z: <https://tokengamer.io/unraveling-the-mysteries-of-procedural-generation-in-gaming/>. [cit. 2024-03-18].
- [28] RA, Francisco. *Procedural Generation: Pros and Cons*. Online. In: Gamedev. 2016. Dostupné z: <https://www.gamedev.net/articles/programming/graphics/procedural-generation-pros-and-cons-r4362/>. [cit. 2024-03-18].

- [29] *Fields*. Online. Blender Manual. 2024, Last updated on 01/15/2024. Dostupné z: https://docs.blender.org/manual/en/latest/modeling/geometry_nodes/fields.html. [cit. 2024-03-18].
- [30] BAILY, Joe. *What Are Fields In Geometry Nodes And How Do They Affect Object Creation?* Online. Blenderbasecamp. 2022. Dostupné z: <https://blenderbasecamp.com/what-are-fields-in-geometry-nodes-and-how-do-they-affect-object-creation/>. [cit. 2024-03-18].
- [31] SATO, Hideki. *[Blender] Enhance your Modeling Skills with Geometry Nodes (Basic)*. Online. In: Styly. 2022. Dostupné z: <https://styly.cc/tips/expanding-modeling-using-geometry-node-basic/>. [cit. 2024-03-18].
- [32] *Node Parts*. Online. Blender Manual. 2024, Last updated on 01/15/2024. Dostupné z: <https://docs.blender.org/manual/en/latest/interface/controls/nodes/parts.html>. [cit. 2024-03-18].
- [33] *Attributes*. Online. Blender Manual. 2024, Last updated on 01/15/2024. Dostupné z: https://docs.blender.org/manual/en/latest/modeling/geometry_nodes/attributes_reference.html. [cit. 2024-03-18].
- [34] BAILY, Joe. *What Are The Different Types Of Attributes In Geometry Nodes?* Online. In: Blenderbasecamp. 2022. Dostupné z: <https://blenderbasecamp.com/what-are-the-different-types-of-attributes-in-geometry-nodes/>. [cit. 2024-03-18].
- [35] *Node Types*. Online. Blender Manual. 2024, Last updated on 01/15/2024. Dostupné z: https://docs.blender.org/manual/en/latest/modeling/geometry_nodes/index.html#node-types. [cit. 2024-03-18].

- [36] *How to Use the New Simulation Nodes in Blender 3.6 LTS* [@Blender Studio]. Online. Dostupné z: https://www.youtube.com/watch?v=RjLiFTNHnl&ab_channel=BlenderStudio. [cit. 2024-03-18].
- [37] *Eight ways to organize nodes in BLENDER!* [@BrandonsDrawings]. Online. 2023. Dostupné z: https://www.youtube.com/watch?v=MS7q6L9gzYE&ab_channel=Brandon%27sDrawings. [cit. 2024-03-18].
- [38] MISHRA, Kanhaiya. *What is the image texture node in Blender 3D, and how is it used?* Online. In: Quora. 2023. Dostupné z: <https://www.quora.com/What-is-the-image-texture-node-in-Blender-3D-and-how-is-it-used/answer/Kanhaiya-Mishra-144>. [cit. 2024-05-06].
- [39] *Image Texture Node*. Online. Blender Manual. 2024, Last updated on 04/23/2024. Dostupné z: https://docs.blender.org/manual/en/latest/modeling/geometry_nodes/texture/image.html. [cit. 2024-05-06].
- [40] *Hair Nodes*. Online. Blender Manual. 2024, Last updated on 04/23/2024. Dostupné z: https://docs.blender.org/manual/en/latest/modeling/geometry_nodes/hair/index.html. [cit. 2024-05-06].

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

2D Two dimensional (dvojrozměrný)

3D Three dimensional (trojrozměrný)

Ctrl Control

GB Gigabyte

GPL General public license

RAM Random-access memory

SEZNAM OBRÁZKŮ

Obrázek 1 Úvodní obrazovka	12
Obrázek 2 Výchozí rozložení, první část (modrá), druhá část (zelená), třetí část (červená) [5].....	13
Obrázek 3 Pracovní prostory	14
Obrázek 4 Mesh objekt a jeho struktura	15
Obrázek 5 Blossm menu a jeho použití.....	18
Obrázek 6 Blossm ukázka importované části terénu z oficiální dokumentace	19
Obrázek 7 Blossm ukázka importované části města z oficiální dokumentace.....	19
Obrázek 8 Object Scatter menu	20
Obrázek 9 Výstup z Addonu Object Scatter.....	21
Obrázek 10 Příklad použití Addonu Object Scatter.....	21
Obrázek 11 SceneCity editor pro práci s grafy [21]	22
Obrázek 12 SceneCity seznam přednastavených grafů měst.....	23
Obrázek 13 Strom uzlů [23].....	24
Obrázek 14 Příklad zapojení Fields	26
Obrázek 15 Blender Node menu.....	28
Obrázek 16 Blender Spreadsheet editor obsahující atributy.....	29
Obrázek 17 Blender simulační zóna	31
Obrázek 18 Blender ukázka použití Texture Nodes [39]	32
Obrázek 19 Empire State Building	34
Obrázek 20 Rozmístěné body na ploše Grid	36
Obrázek 21 Spojené body vygenerované na ploše Grid	37
Obrázek 22 Seskupení uzlů Outline.....	37
Obrázek 23 Generování náhodného tvaru města	38
Obrázek 24 Vygenerované body na ploše centra města	39
Obrázek 25 Rozmístěné budovy na vygenerovaných bodech v centru města.....	40
Obrázek 26 Zapojení Geometry Nodes pro vygenerování centrum města.....	41
Obrázek 27 Náhodná transformace budov.....	42
Obrázek 28 Vstup Height of the buildings	43
Obrázek 29 Náhodná rotace budov na ose Z	44
Obrázek 30 Funkce XYZscaler.....	45
Obrázek 31 Ukázka uzlu Group Input pro centrum a předměstí	46
Obrázek 32 Ukázka uzlu Group Input pro vzdálené předměstí.....	47
Obrázek 33 Ukázka funkce generování budov od středu směrem ven.....	48

Obrázek 34 Zapojení funkce generující budovy od centra směrem ven.....	49
Obrázek 35 Ukázka uzlu Bounding Box	50
Obrázek 36 Pozůstalé plochy po náhodném odstranění	51
Obrázek 37 Rozpojení společných hran a zakulatění ploch parků	52
Obrázek 38 Rozpojení společných hran a zakulatění ploch parků	53
Obrázek 39 Zapojení Geometry Nodes pro skupinu uzlů Outline.....	53
Obrázek 40 Ukázka vygenerovaných ploch parků s texturou	54
Obrázek 41 Zapojení Geometry Nodes pro generování stromů	55
Obrázek 42 Zapojení Geometry Nodes pro generování kamene	56
Obrázek 43 Zapojení Geometry Nodes kamenná cesta	57
Obrázek 44 Zapojení Geometry Nodes k odstranění stromů z kamenných cest	58
Obrázek 45 Ukázka vygenerovaného parku	59
Obrázek 46 Zvětšení šířky hrana na ose Y	60
Obrázek 47 Dokončený UV Mapping silnice.....	61
Obrázek 48 Dokončený model chodníku.....	61
Obrázek 49 Dokončený přechod s UV Mappingem.....	62
Obrázek 50 Dokončený model silnice	63
Obrázek 51 Duplikace silnic na ose X.....	64
Obrázek 52 Zapojení Geometry Nodes pro duplikaci na ose X	65
Obrázek 53 Duplikace silnic do všech stran	65
Obrázek 54 Zapojení Geometry Nodes pro náhodnou rotaci silnic na ose Z	66
Obrázek 55 Zapojení Geometry Nodes odstraňující přesahující část silnic	67
Obrázek 56 Vytvořený objekt prvního uzlu Mesh Boolean	68
Obrázek 57 Zapojení Geometry Nodes skupiny uzlů Raycast	69
Obrázek 58 Přidané vstupy v praktické části uzlu Group Input	72
Obrázek 59 Zobrazení přidaných vstupů uzlu Group Input v panelů vlastností	72
Obrázek 60 Nově přidané prázdné vstupy String uzlu Group Input.....	74
Obrázek 61 Přejmenované prázdné vstupy uzlu Group Input v panelu vlastností	74

SEZNAM PŘÍLOH

Příloha P I: Struktura CD

Příloha P II: Vyrenderované různé pohledy na město

Příloha P III: Ukázky vygenerovaných měst

PŘÍLOHA P I: STRUKTURA CD

Struktura CD obsahu:

- Text bakalářské práce v digitální formě – soubory .docx a .pdf
- Adresář CityGenerator – obsahuje .blend soubor
- Adresář Renders – obsahuje veškeré rendery a obrázky různých příkladů vygenerovaných měst

PŘÍLOHA P II: VYRENDEROVANÉ RŮZNÉ POHLEDY NA MĚSTO









PŘÍLOHA P III: UKÁZKY VYGENEROVANÝCH MĚST

